UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Understanding isoform expression and alternative splicing biology through single-cell RNAseq

Author: Ángeles Arzalluz-Luque

Supervisors: Dr. Ana Conesa Cegarra
Dr. Sonia Tarazona Campos

January 2024

Abstract

In the world of transcriptomics, the emergence of single-cell RNA sequencing (scRNA-seq) ignited a revolution in our understanding of cellular diversity, unraveling novel mechanisms in tissue heterogeneity, development and disease. However, when this thesis began, using scRNA-seq to understand Alternative Splicing (AS) was a challenging frontier due the inherent limitations of the technology. In spite of this research gap, pertinent questions persisted regarding cell-level AS patterns, particularly concerning the recapitulation of isoform diversity observed in bulk RNA-seq data at the cellular level and the roles played by cell and cell type-specific isoforms.

The work conducted in the present thesis aims to harness the potential of scRNA-seq for alternative isoform analysis, outlining technical and analytical challenges and designing computational methods to overcome them. To achieve this, we established a roadmap with three main aims. First, we set requirements for studying isoforms using scRNA-seq and conducted an extensive review of existing research, interrogating whether these requirements were met. Combining this acquired knowledge with several computational simulations allowed us to delineate the strengths and pitfalls of available data generation methods and computational tools. During the second research stage, this insight was used to design a suitable data processing pipeline, in which we jointly employed bulk long-read and short-read scRNA-seq sequenced from full-length cDNAs to ensure adequate isoform reconstruction as well as sensitive cell-level isoform quantification. Additionally, we refined available transcriptome curation strategies, introducing them as innovative

modules in the transcriptome quality control software SQANTI3. Lastly, we harnessed single-cell isoform expression data and the rich biological diversity inherent in scRNA-seq, encompassing various cell types, in the design of a novel isoform co-expression analysis method. Percentile correlations effectively mitigated single-cell noise, unveiling clusters of co-expressed isoforms and exposing a layer of regulation in cellular identity that operated independently of gene expression. We additionally introduced co-Differential Isoform Usage (coDIU) analysis, enhancing our ability to interpret isoform cluster networks. This endeavour, combined with the computational annotation of functional sites and domains in the long read-defined isoform models, unearthed a distinctive functional signature in coDIU genes. This research effort materialized in the release of acorde, an R package that encapsulates all analyses functionalities developed throughout this thesis, providing a reproducible means for the scientific community to further explore the depths of alternative isoform biology within single-cell transcriptomics.

This thesis describes a complex journey aimed at unlocking the potential of scRNA-seq data for investigating AS and isoforms: from a landscape marked by the scarcity of tools and guidelines, towards the development of novel analysis solutions and the acquisition of valuable biological insight. In a swiftly evolving field, our methodological contributions constitute a significant leap forward in the application of scRNA-seq to the study of alternative isoform expression, providing innovative resources for delving deeper into the intricacies of post-transcriptional regulation and cellular function through the lens of single-cell transcriptomics.

Resumen

La introducción de la secuenciación de ARN a nivel de célula única (scRNA-seq) en el ámbito de la transcriptómica ha redefinido nuestro entendimiento de la diversidad celular, arrojando luz sobre los mecanismos subyacentes a la heterogeneidad tisular y su papel en procesos dinámicos como el desarrollo y la progresión de enfermedades. No obstante, al inicio de esta tesis, las limitaciones inherentes a esta tecnología obstaculizaban su aplicación en el estudio de procesos celulares complejos, entre ellos el *splicing* alternativo del ARN. A pesar de ello, los patrones de *splicing* a nivel de célula única planteaban incógnitas que esta tecnología tenía el potencial de resolver: ¿es posible observar, a nivel celular, la misma diversidad de isoformas que se detecta mediante RNA-seq a nivel de tejido (*bulk RNA-seq*)? ¿Qué función desempeñan las isoformas alternativas en la constitución de la identidad celular?

El objetivo de esta tesis es desbloquear el potencial del scRNA-seq para el análisis de isoformas alternativas, abordando sus dificultades técnicas y analíticas mediante el desarrollo de nuevas metodologías de análisis computacional. Para lograrlo, se trazó una hoja de ruta con tres objetivos principales. En primer lugar, se establecieron cuatro requisitos indispensables para el estudio de las isoformas mediante scRNA-seq, llevando a cabo una revisión exhaustiva de la literatura existente para evaluar su cumplimiento. Tras completar este marco con diversas simulaciones computacionales, se identificaron las debilidades y fortalezas de los métodos de scRNA-seq y de las herramientas computacionales disponibles. Durante la segunda etapa de

la investigación, los conocimientos adquiridos mediante este trabajo teórico se utilizaron para diseñar un protocolo óptimo de procesamiento de datos de scRNA-seq. En concreto, se integraron datos de lecturas largas a nivel de tejido (*bulk long reads*) con datos de scRNA-seq para garantizar una identificación adecuada de las isoformas así como su cuantificación precisa a nivel celular. Este proceso de integración de datos permitió, además, ampliar las estrategias computacionales disponibles para la reconsrucción de transcriptomas a partir de lecturas largas, así como la distribución de estos avances a la comunidad bioinformática mediante su implementación en SQANTI3, un software de referencia en transcriptómica. Por último, los datos procesados se utilizaron para desarrollar un nuevo método de análisis de co-expresión de isoformas que tiene como objetivo desentrañar las redes de regulación del *splicing* alternativo implicadas en la constitución de la identidad celular.

Dada la elevada variabilidad existente en los datos de scRNA-seq, este método se fundamenta en la utilización de una estrategia de correlación basada en percentiles que atenúa el ruido técnico, posibilitando así la identificación de grupos de isoformas co-expresadas. Una vez configurada la red de co-expresión, se introdujo una nueva estrategia de análisis para la detección de patrones de co-utilización de isoformas que suceden de forma independiente a la expresión a nivel de gen, denominada *co-Differential Isoform Usage* (coDIU). Este enfoque facilita la identificación y caracterización de una capa de regulación de la identidad celular atribuible únicamente a mecanismos post-transcripcionales, incluyendo el *splicing* alternativo. Finalmente, para una interpretación biológica más profunda, se aplicó una estrategia de anotación computacional de motivos y dominios funcionales en las isoformas definidas con lecturas largas, con el propósito de revelar las propiedades biológicas de las isoformas involucradas en la red de co-expresión. El resultado de estas investigaciones culmina en el lanzamiento de acorde, un paquete de R que encapsula las diferentes metodologías desarrolladas en esta tesis, potenciando la reproducibilidad de sus resultados y proporcionando una nueva herramienta para explorar la biología de las isoformas alternativas a nivel de célula única.

En resumen, esta tesis describe una serie de esfuerzos destinados a desbloquear el potencial de los datos de scRNA-seq para avanzar en la comprensión del *splicing* alternativo y las isoformas. Desde un contexto marcado por la escasez de herramientas y conocimiento previo, se han desarrollado soluciones de análisis innovadoras que permiten la generación de nuevas hipótesis biológicas. En un campo en constante evolución, consideramos que los métodos presentados representan un avance significativo en la aplicación de scRNA-seq al estudio de las isoformas alternativas, proporcionando recursos innovadores para profundizar en las complejidades de la regulación post-transcripcional y la función celular a través de la transcriptómica a nivel de célula única.

Resum

La introducció de la seqüenciació d'ARN a escala de cèl·lula única (scRNA-seq) en l'àmbit de la transcriptòmica ha redefinit el nostre enteniment de la diversitat cel·lular, projectant llum sobre els mecanismes subjacents a l'heterogeneïtat tissular i el seu paper en processos dinàmics com el desenvolupament i la progressió de malalties. No obstant això, a l'inici d'aquesta tesi, les limitacions inherents a aquesta tecnologia obstaculitzaven la seua aplicació en l'estudi de processos cel·lulars complexos, entre ells l'*splicing* alternatiu de l'ARN. Malgrat això, els patrons d'*splicing* a escala de cèl·lula única plantejaven incògnites que aquesta tecnologia tenia el potencial de resoldre: es pot observar, a escala cel·lular, la mateixa diversitat d'isoformes que es detecta mitjançant RNA-seq a escala de teixit (*bulk RNA-seq*)? Quina funció tenen les isoformes alternatives en la constitució de la identitat cel·lular?

L'objectiu d'aquesta tesi és desbloquejar el potencial de scRNA-seq per a l'anàlisi d'isoformes alternatives, abordant les dificultats tècniques i analítiques mitjançant el desenvolupament de noves metodologies d'anàlisi computacional. Per a això, es va tramar una ruta amb tres objectius principals. En primer lloc, es van establir quatre requisits indispensables per a l'estudi de les isoformes mitjançant scRNA-seq, realitzant una revisió exhaustiva de la literatura existent per a avaluar-ne el compliment. Després de completar aquest marc amb diverses simulacions computacionals, es van identificar les debilitats i forces dels mètodes de scRNA-seq i les eines computacionals disponibles. Durant la segona etapa de la investigació, els coneixements

adquirits mitjançant aquest treball teòric es van utilitzar per a dissenyar un protocol òptim de processament de dades de scRNA-seq. Concretament, es van integrar dades de lectures llargues a escala de teixit (*bulk long reads*) amb dades de scRNA-seq per a garantir una identificació adequada de les isoformes així com la seua quantificació precisa a escala cel·lular. Aquest procés d'integració de dades va permetre, a més, ampliar les estratègies computacionals disponibles per a la reconstrucció de transcriptomes a partir de lectures llargues, així com la distribució d'aquests avenços a la comunitat bioinformàtica mitjançant la seua implementació en SQANTI3, un programari de referència en transcriptòmica. Finalment, les dades processades es van fer servir per a desenvolupar un nou mètode d'anàlisi de coexpressió d'isoformes que té com a objectiu desxifrar les xarxes de regulació de l'*splicing* alternatiu implicades en la constitució de la identitat cel·lular.

Davant l'elevada variabilitat tècnica existent en les dades de scRNA-seq, aquest mètode es fonamenta en la utilització d'una estratègia de correlació basada en percentils que minimitza el soroll tècnic, possibilitant així la identificació de grups d'isoformes coexpressades. Un cop configurada la xarxa de coexpressió, es va introduir una nova estratègia d'anàlisi per a la detecció de patrons de co-utilització d'isoformes que succeeixen de forma independent a l'expressió del seu gen, anomenada *co-Differential Isoform Usage* (coDIU). Aquest enfocament facilita la identificació i caracterització d'una capa de regulació de la identitat cel·lular atribuïble únicament a mecanismes post-transcripcionals, incloent-hi l'*splicing* alternatiu. Finalment, per a una interpretació biològica més profunda, es va aplicar una estratègia d'anotació computacional de motius i dominis funcionals en les isoformes definides amb lectures llargues, amb la finalitat de revelar les propietats biològiques de les isoformes involucrades en la xarxa de coexpressió. El resultat d'aquestes investigacions culmina en el llançament d'acorde, un paquet de R que encapsula les diferents metodologies desenvolupades en aquesta tesi, potenciant la reproductibilitat dels seus resultats i proporcionant una nova eina per a explorar la biologia de les isoformes alternatives a escala de cèl·lula única.

En resum, aquesta tesi descriu una sèrie d'esforços destinats a desbloquejar el potencial de les dades de scRNA-seq per a avançar en la comprensió de l'*splicing* alternatiu i les isoformes. Des d'un context marcat per l'escassetat d'eines i coneixement previ, s'han desenvolupat solucions d'anàlisi innovadores que permeten la generació de noves hipòtesis biològiques. En un camp en constant evolució, considerem que els mètodes presentats representen un avanç significatiu en l'aplicació de scRNA-seq a l'estudi de les isoformes alternatives, proporcionant recursos innovadors per a aprofundir en les complexitats de la regulació post-transcripcional i la funció cel·lular a través de la transcriptòmica a escala de cèl·lula única.

Contents

Chapter 1

Introduction

1.1 The era of transcriptomics: from bulk to cell-level resolution

Transcriptomics, the comprehensive study of the set of genes that are active in an organism, is pivotal in deciphering the intricate language of gene expression. The identification of transcripts that are expressed in a given biological sample, i.e. the transcriptome, provides a dynamic snapshot of an organism's gene expression landscape and offers a deeper perspective into the functional complexity of living systems. Since the mid-2000s, high-throughput sequencing technologies have had a profound impact on the way we conduct transcriptome research, enabling access to the entire span of transcripts in a biological sample thanks to bulk RNA sequencing (bulk RNA-seq) [1, 2]. In this process, the RNA content of an entire tissue or cell population is measured as an ensemble average, generating sample-level expression estimates by counting detected transcript molecules. RNA-seq applications range from classic evaluations of differential transcript or gene expression between samples [1] to more diverse problems such as the characterization of gene expression dynamics [3], gene boundaries [4, 5], translation efficiency [6], RNA–protein interactions [7, 8] and large-scale characterization of alternative splicing [9, 10], to name a few.

Despite offering valuable insights into the global expression patterns of a sample, bulk RNA-seq fell short in capturing the nuances that drive cellular behavior. The need for a more detailed, high-resolution view of the transcriptome, and the conviction that the average behaviour captured by bulk RNA-seq could obscure sample heterogeneity and cell-level expression patterns, catalyzed the emergence of single-cell RNA sequencing (scRNA-seq) [11]. By enabling the profiling of individual cells, the technology quickly unlocked the capacity to scrutinize the distinct transcriptional profiles of rare cell types and complex tissues [12–16], gain novel insight into cell differentiation mechanisms [17–21] or tackle tumor heterogeneity [22, 23], unveiling unprecedented biological diversity and complex gene regulation patterns that would not be observable at the bulk level.

1.1.1 Single-cell RNA-seq technologies

Transcriptome sequencing is based on an elaborate library preparation process which, although slightly different depending on the method of choice, consists in several common steps [2]. First, RNA is extracted from individual samples and converted to cDNA by primming of the polyA tail and subsequent reverse-transcription. Upon cDNA synthesis, barcodes and adapter sequences are attached to each molecule. When required, this is followed by linear PCR amplification to increase the available amount of cDNA. Finally, generated cDNA can be sequenced using different technologies and equipment. Even though bulk and cell-level sequencing largely rely on the same principles, handling small amounts of starting material creates unique challenges and requires the adaptation of RNA-seq library preparation to scRNA-seq.

First, as opposed to bulk protocols, in which the entire sample is used for RNA extraction, cells need to be isolated prior to this process, which can be done using different techniques, including Fluorescence-Activated Sorting (FACS), Laser-Capture Microdissection (LCM) and microfluidics platforms (see [24] for a detailed review). The method used for cell isolation depends on the type of sample and research objectives, and may affect the generated data [24]. For instance, methods such as FACS and microfluidics allow for larger throughput, that is, for the sequencing of a larger number of cells (i.e. samples), at more efficient costs. The inclusion of more samples increases the cellular diversity captured in the experiment and has a positive effect on statistical power, although this property also depends on library preparation and sequencing methodologies, as the number of reads generated per cell plays an important part in gene detection [25, 26]. Method selection can further impact data composition, depending on whether or not they enable for the enrichment of specific cells, as is the case of FACS or LCM, which allow fluorescence-based and visual inspection of cellular properties of interest, respectively. Finally, each method offers different cost-efficiency depending on reagent volume, cell capture efficiency and time requirements, with microfluidics methods yielding substantial advantages in this direction [27]. Finally, methods such as FACS may generate sample damage or exacerbate RNA degradation as cells go through the instrument, and the application of microfluidics may require previous enzymatic treatment to generate a cell suspension, which can negatively affect cell viability [28].

In addition to cell isolation, a high number of PCR amplification rounds (often $>$30 and up to 40 cycles) is required to generate sufficient cDNA for sequencing [27], which in turn increases technical variability [29] and creates strong PCR bias [30]. Finally, appropriate barcodes need to be incorporated into the cDNA molecule upon reverse transcription in order to be able to identify the cell of origin and conduct cell multiplexing. As a result of this complex scenario, a myriad of protocols were quickly developed in the early days of the technology to tackle single-cell data generation [25, 28]. These can be classified into three main groups of methods, according to the combination of library preparation and sequencing technologies that they employ: end-tagging or Unique Molecular Identifier-based (UMI-based) methods [14, 31, 32], full-length or SMART-based methods [33, 34], and long-read or Single Molecule Sequencing (SMS) technologies [35, 36]. In figure 1.1, we present an overview of these three strategies, with a special focus on illustrating their performance regarding gene and isoform detection.

End-tagging methods (figure 1.1, left panel) focus on reading the 3' or 5' end of transcripts, and use UMIs for the identification of individual transcript molecules and the elimination of non-linear amplification bias [31]. UMIs are short (6-8bp), random oligonucleotides that are incorporated upon cDNA synthesis, and before PCR amplification. After PCR, cDNA molecules that originated from the same transcript will therefore have the same UMI, which allows for molecular counting after PCR, collapsing reads with matching UMIs and mapping sites. In most protocols (e.g. inDrop [32], Drop-seq [14]), the UMI is attached at the 3' end, which precludes usage of 5' end reads for transcript characterization. Importantly, these methods were the first to leverage the usage of microfluidics platforms for library preparation [14, 32]. This quickly enhanced their throughput and decreased their costs [38], leading to their adoption as the main scRNA-seq method in the field. However, the usage of minimal volumes in droplet-based technologies makes them

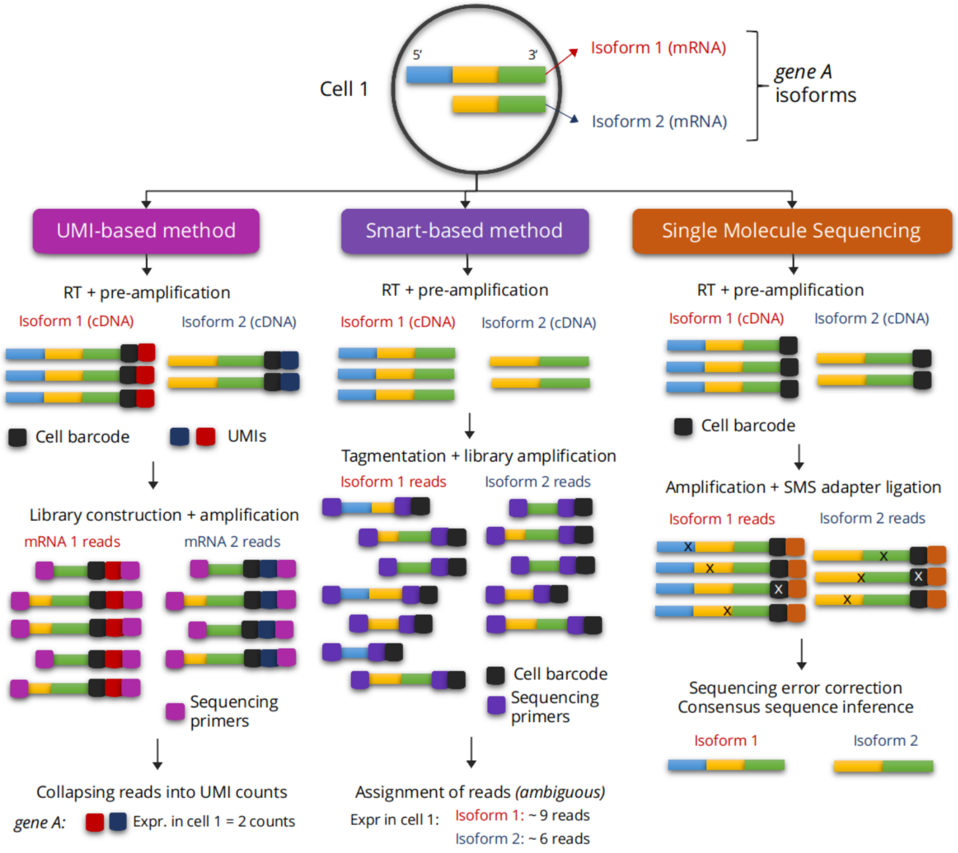

Cell 1
5'
3'
Isoform 1 (mRNA)
Isoform 2 (mRNA)
gene A isoforms
UMI-based method
Smart-based method
Single Molecule Sequencing
RT + pre-amplification
Isoform 1 (cDNA)
Isoform 2 (cDNA)
Cell barcode
UMIs
Library construction + amplification
mRNA 1 reads
mRNA 2 reads
Sequencing primers
Collapsing reads into UMI counts
gene A: Expr. in cell 1 = 2 counts
Tagmentation + library amplification
Isoform 1 reads
Isoform 2 reads
Assignment of reads (ambiguous)
Expr in cell 1: Isoform 1: ~ 9 reads
Isoform 2: ~ 6 reads
Amplification + SMS adapter ligation
X
Sequencing error correction
Consensus sequence inference
Isoform 1
Isoform 2

Figure 1.1: Single-cell RNA-seq methods. The combination of library preparation and sequencing technologies yields three distinct methods. UMI-based methods are limited to the sequencing of the 3' (or 5' end), which enables the usage of UMIs in addition to early cell barcoding. SMART-based methods are incompatible with UMIs and generate short reads spanning the full transcript, with barcodes being added during tagmentation. Single-molecule sequencing methods generate a single read per transcript molecule but suffer from a high prevalence of sequencing errors. From Arzalluz-Luque, A. and Conesa, A. [37].

prone to missing transcripts upon mRNA capture, causing an unprecedented level of zero expression values across cells, which are commonly referred to as dropouts [39].

SMART-based methods (figure 1.1, middle panel), among which the Smart-seq2 [34]protocol constitutes the most widely used approach, generate full-length coverage of transcripts by adding an enhanced reverse transcription step that ensures capture of the entire transcript upon cDNA synthesis. The system, known as SMART [40], uses the Moloney murine leukemia virus (MMLV) reverse transcriptase to leave a 5' oligonucleotide overhang after the enzyme has reached the end of the first strand, which is then used for template-switching, i.e. priming and synthesis of the second strand of the cDNA. To generate multiple, correctly labeled short reads from each cDNA molecule, library preparation includes a process called tagmentation [33]. In this process, the Tn5 a transposase enzyme is used to fragment cDNA and simultaneously incorporate sequencing adapters. While this strategy is key to the full-length transcript coverage provided by SMART-based methods, which constitutes one of its main strengths, it also generates incompatibilities with UMIs due to these tags being absent from mid-transcript and 5' reads. Due to their inability to effectively remove PCR duplicates, full-length methods have been reported to create length biases in gene detection, causing transcripts from longer genes to be overrepresented in sequencing libraries [41].

Both groups of methods described above rely on short-read sequencing using the Illumina platform. While short-read single-cell libraries are well-suited for many gene-level applications, they have limitations when it comes to capturing comprehensive information about alternative splicing, isoforms and other sources of transcript variation [42–44]. To address these challenges and enable a more holistic characterization of single-cell transcriptomes, researchers have turned to Single Molecule Sequencing (SMS) or long-read RNA-sequencing (lrRNA-seq) technologies (figure 1.1, right panel), which offer distinct advantages due to their ability to detect the entire transcript sequence in a single read. These lrRNA-seq methods require full-length cDNA libraries and are often integrated with SMART-based approaches for long-read single-cell RNA sequencing [45]. In contrast to the tagmentation step present in the Smart-seq2 Illumina protocol, lrRNA-seq methods include specialized library preparation adaptations that facilitate the sequencing of entire cDNA molecules. Notably, two technologies, Single-Molecule Real-Time (SMRT) Sequencing by Pacific Biosciences (PacBio) [46, 47] and Oxford Nanopore Technologies (ONT) [48, 49], emerged as prominent choices in the field, and have become the leading methods for lrRNA-seq.

Both PacBio and ONT technologies, however, are prone to sequencing errors [48, 50, 51] and thus employ distinct strategies to sequence full transcripts and mitigate inaccuracies, each offering unique advantages. PacBio employs a circular consensus sequencing (CCS) approach, where a cDNA molecule is circularized and sequenced multiple times. The resulting reads, consisting in multiple sequencing passes, are subsequently concatenated and collapsed into a consensus sequence. Even though the accuracy of the final transcript sequence may vary depending on the number of passes the cDNA molecule undergoes during sequencing, remarkable results have been obtained for transcriptomics applications [35, 52]. In contrast, ONT jointly reports the sequences of the forward and reverse strands. Initially, a harpin adapter was used to create two-dimensional (2D) reads, linking forward and reverse strands [49], whereas later, time-based inference was introduced for the same purpose [45]. Although less effective than PacBio in terms of increasing sequencing accuracy [48, 50, 51], these innovations maintain higher throughput and cost efficiency [44, 49], making ONT a competitive choice for lrRNA-seq applications.

By combining these long-read sequencing methods with cell barcoding and multiplexing strategies, lrRNA-seq has been readily applied to single-cell RNA sequencing [53, 54]. Nevertheless, in order to be able to correctly assign reads to individual cells, error correction steps have steadily been integrated into long-read scRNA-seq analysis pipelines to rectify demultiplexing errors induced by sequencing inaccuracies [55–57].

The transition from bulk RNA-seq to scRNA-seq represents a significant leap in our capacity to decode the complexity of gene expression. However, the technology is inherently more complex and computationally demanding, and the adoption of scRNA-seq is not without challenges. Specifically, the need for careful consideration of experimental design, data processing and quality control cannot be overstated [27, 58, 59]. Technical variability in single-cell data is often higher than in bulk datasets, and issues related to dropouts, technical biases, confounding factors and data sparsity must be effectively addressed [28, 29, 60, 61]. Furthermore, the high dimensionality of single-cell datasets, although powerful regarding the large diversity of biological signals encoded in them, can present difficulties in data analysis and interpretation [62]. After data generation, the choice of appropriate analytical tools and techniques is therefore crucial in deriving meaningful biological insights [63].

1.2 Post-transcriptional regulation of gene expression: alternative splicing and mRNA isoform expression

The central dogma of molecular biology, classically limited to a one-gene-one-protein paradigm, has undergone a profound transformation over the past decades. The concept of Alternative Splicing (AS), a molecular mechanism that allows transcribed pre-mRNAs to be processed in different ways and give rise to multiple mRNA isoforms, perfectly exemplifies this conceptual shift. These isoforms, characterized by differences in their exon composition, can encode proteins with diverse structures and functions, each tailored to specific cellular contexts, significantly expanding the functional potential of a single gene [64, 65]. As a result, AS provides eukaryotic cells with a powerful mechanism for the expansion and fine-tuning of the catalog of functionalities encoded by the genome.

AS is a pervasive and dynamic phenomenon among eukaryotes, ranging from yeast to humans, whose prevalence increases with organism complexity [66–68]. Specifically, after gene activation and mRNA transcription in the nucleus, introns are removed from newly synthesized pre-mRNA molecules and exons are joined to form mature mRNAs [69]. This is achieved by a tightly regulated process, involving the identification of different types of cis-regulatory elements by RNA-binding proteins, namely the constituents of the spliceosome [70], as well as auxiliary proteins that act as enhancers or silencers to the splicing process [71]. These regulatory sequences are situated within the boundaries of exons and introns, known as donor and acceptor splice sites. Specifically, the donor site is located at the 5' end of the intron, and is typically defined by the presence of the "GU" sequence immediately at the beginning of the intron. The acceptor site, on the other hand, is situated at the 3' end of the intron, closest to the transcript end, and generally contains the sequence "AU". The splicing machinery binds the "GU" and "AU" sequences, which are highly conserved in eukaryotes and are known as *canonical* splice sites. After the removal of the intron, the two flanking exons become adjacent, and the splice site becomes a splice junction (SJ).

During transcription, mRNA structural diversity can be generated by the selection of alternative transcription start (TSS) and termination sites (TTS) by the gene expression regulatory machinery. At the post-transcriptional level, AS and alternative polyadenylation (APA) have been identified as the main mechanisms that generate structural diversity during mRNA maturation. AS can be categorized into several well-defined patterns, including exon skipping, mutually exclusive exons, intron retention, alternative donor (5') and acceptor (3') splice sites and alternative first and last exons (figure 1.2) [69]. In exon skipping, specific exons within a pre-mRNA transcript are either included or excluded from the mature mRNA. Mutually exclusive exons are a particular type of exon skipping, in which two or more exons present a pattern

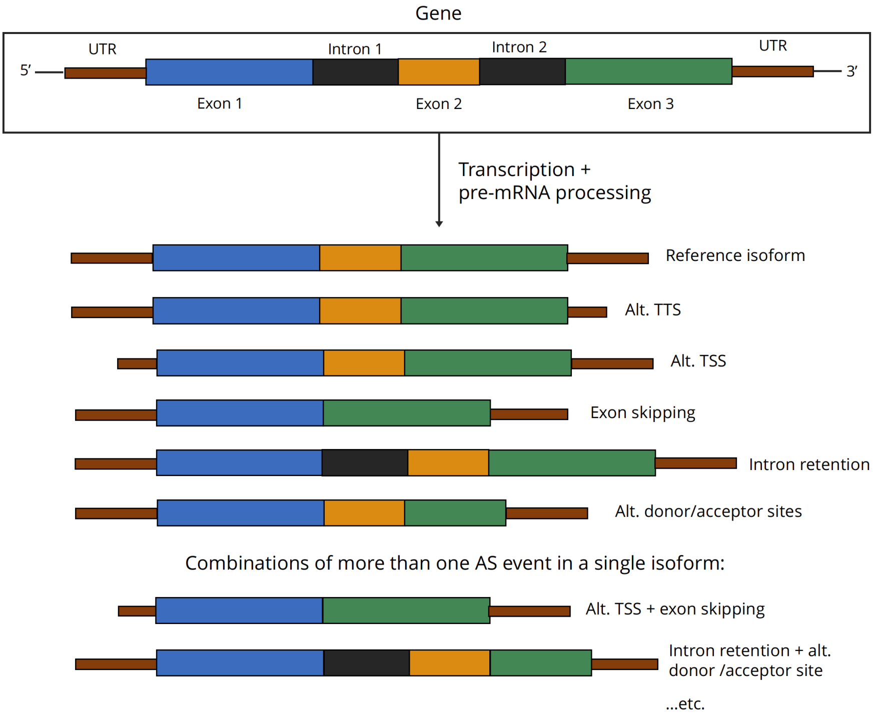


Figure 1.2: Sources of transcript variation that yield alternative isoforms and their position along the transcript. When compared with a reference isoform (for convenience, that including all exons, no introns and the complete UTRs), alternative TSS (transcription start sites) and TTS (transcription termination sites) are generated during the transcription process by shortening of the UTRs. Processing of the pre-mRNA eliminates or retains introns and exons, adding variability to the isoforms that can be generated from the gene. In addition, more than one event can simultaneously be present in the same isoform, and consequently, isoform diversity will increase with the number of possible combinations of AS events. From Arzalluz-Luque, A. and Conesa, A. [37]

by which only one of them can be included in the final transcript. Contrarily to exon skipping, intron retention consists in the inclusion of one or more introns in the mature mRNA, often leading to the production of non-coding isoform variants. AS at 5' and 3' splice sites involves the selection of splice sites situated upstream or downstream of the expected site, resulting in the inclusion or exclusion of specific regions of exons or introns. Finally, AS can also impact the selection of the first or last exon in a transcript, modulating the 5' or 3' untranslated regions (UTRs) and influencing the stability, localization and translation efficiency of mRNAs. APA, on the other hand, occurs by the identification of polyadenylation signals upstream and downstream of the 3' end of the mRNA, generating the cleavage of the molecule and the incorporation of a polyA tail [72]. Multiple polyA sites per gene can be identified for the majority of eukaryotic genes which, together with AS, generates additional post-transcriptional diversity and influences mRNA fate, translation and coding potential [73].

In addition to the alternative usage of individual splicing events, eukaryotic transcriptomes exhibit a remarkable capacity to combine multiple events, giving rise to complex splicing patterns (figure 1.2) [74, 75]. This phenomenon of combinatorial control of AS can enrich the regulatory and functional complexity of eukaryotic organisms, unlocking a wide variety of fine-tuned transcriptional responses from a relatively modest number of genes [76]. Comprehensive genome-wide studies reveal that a substantial portion of multi-exon human genes, ranging from 90% to 95%, undergo some degree of alternative splicing, resulting in tissue-specific isoform expression patterns

[9, 10]. These patterns have been shown to present even further complexity at the single-cell and cell type levels, with splicing events and alternative isoforms presenting distinct roles in establishing cell identity and function in a context-dependent manner [77, 78].

1.2.1 Computational approaches for the study of AS and isoform transcriptomics

The advent of high-throughput RNA-seq has revolutionized our ability to dissect the complexities of AS and alternative isoform expression in eukaryotic transcriptomes. However, short-read sequencing data precludes the unambiguous assignment of reads to isoforms due to multiple alternative isoforms presenting high structural similarity, that is, similar inclusion and exclusion patterns for several AS events [79]. Mapping algorithms have been adapted for this purpose, namely by including AS-aware mechanisms that are able to detect reads mapping across splice junctions [80]. Most quantification tools use this information to generate event-level metrics, such as Percentage Spliced In (PSI), which assesses the proportion of reads supporting the inclusion of a given exon or junction, supplying an estimation of the percentage of event-including isoforms among the pool of a gene's alternative variants [81–83]. Quantifying individual events can be insufficient when complex AS patterns are present and may damage interpretability due to the lack of correspondence between event patterns and the final structure of mature mRNAs, precluding the characterization of full-length isoforms and the investigation of their different functional roles. To circumvent read fragmentation, authors have additionally employed Percent-isoform (Pi) values, which captures the proportion of within-gene spliced reads that span a given isoform, and is applicable to both short and long-read data [84]. This metric, however, only allows relative measurements of AS changes, thus being limited to within-gene comparisons.

Isoform-level quantification therefore requires tools that can estimate the number of reads belonging to isoforms with overlapping exons, producing a single expression value per transcript. Tools like RSEM [85] employ the Expectation-Maximization (EM) algorithm to iteratively estimate isoform abundances from genome alignments, accounting for ambiguities in read assignment by computing the probability that a read is generated from a particular isoform. Alternatively, pseudoalignment methods avoid the need for genome alignment by rapidly assigning reads to transcript sequences based on k-mer matches, subsequently estimating isoform-level counts [86–88]. These methods facilitate rapid and accurate quantification without requiring time-consuming computation and extensive resources.

More recently, transcriptomics approaches based on lrRNA-seq emerged to eliminate the need for short-read transcriptome assembly [52, 89, 90]. Long-reads can readily capture AS complexities, enabling the precise delineation of event patterns, mutually exclusive exons, and other isoform-specific features, even when inclusion coordination occurs among distant events [91]. Moreover, lrRNA-seq data enables the quantification of isoform expression levels directly from the sequencing data. These strategies, however, require the identification of expressed isoforms from the data, a process that has proved difficult in a context where sequencing errors and library preparation artifacts are hard to distinguish from novel and/or non-canonical splice variants [50]. Correct long-read data processing therefore constitutes a non-trivial task and, while the technology presents unprecedented advantages regarding full-length sequencing, it also comes with unique computational challenges for the transcriptomics field.

The research undertaken in this thesis resides at the intersection of single-cell research and alternative splicing investigation. On one hand, our goal is to gain fresh insights into post-transcriptional regulation mechanisms. To do so, we require an experimental context that can accommodate multiple conditions and a substantial number of observations. This particular context is offered by scRNA-seq data, which offers a wealth of biological diversity through its simultaneous capture of gene expression profiles across diverse cells and cell types. However, as shown in this introduction, the analytical intricacies associated with this type of data were non-trivial, and methodological innovations were still required to enable new, diverse forms of computational analysis. As a result, this project was devised as a global method development effort that, considering the technological and analytical context at the time, could unlock the potential of single-cell datasets for isoform-level analysis. In the following chapter, we will provide context for this motivation and outline specific aims and sub-aims that will will serve as our guiding framework for achieving this overarching objective.

Chapter 2

Motivation, aims and contributions

2.1 Motivation

The present thesis has been carried out at the Genomics of Gene Expression laboratory, currently based in the Institute for Integrative Systems Biology (I2SysBio, UV-CSIC) in Valencia, which has extensive experience in the computational analysis of alternative splicing and isoform expression using both short and long-read RNAseq data. At the time, the group had undertaken the development of several methods that enabled Functional Iso-Transcriptomics (FIT) analysis -that is, the exploration of the potential biological role of transcriptome-scale changes in alternative isoform expression- and their application to bulk samples from a mouse neural development model. As these projects progressed, the limitations of bulk data for capturing the complexity of the biological system under study became evident. The research project presented in this thesis was born of the need to extend the methods and insights that arised from this previous work to the cell-level perspective; and was funded by the BIO2015-71658 and BES-2016-076994 grants awarded by the Spanish Ministry of Science and Innovation.

At that time, however, the technologies for the generation of single-cell RNAseq data were still in its early days and therefore suffered from abundant limitations, both regarding technical and analytical issues. As a result, barely any published studies had attempted to explore the alternative isoform landscape using single-cell data and, when doing so, these analyses had been done merely as proof-of-concept. The scenario in which this thesis started was therefore a challenging one. First, the extent to which isoform expression could be studied using single-cell data had not yet been thoroughly evaluated and virtually no dedicated computational methods existed to enable these analyses: on the contrary, most research efforts had tackled the quantitative analysis of individual splicing events, rather than full-length, alternative transcript isoforms.

In order to fill these gaps, we set out to better understand the potential of the single-cell RNAseq technology in order to find innovative ways to leverage this type of data for the study of alternative isoform expression.

2.2 Aims

The general goal of the present thesis project is therefore to unlock isoform-level analyses in single-cell data, developing novel computational methods that use cell-level transcriptomics data to gain additional insight that can help understand the mechanics of alternative splicing. To achieve this, we established three main objectives, which will be described below.

1. **Evaluate the state-of-the-art to understand the potential of single-cell RNAseq data for the study of isoforms.**
   First, a thorough evaluation of possible alternatives for single-cell isoform studies will be performed in order to successfully tackle the central aim of this thesis. Specifically, we set the following secondary objectives:
   (a) Understand the experimental strategies available to produce single-cell RNAseq data and evaluate their suitability for isoform-level analysis.

(b) Use simulated data to evaluate the technical limitations of different single-cell data generation methods for isoform detection and quantification.

(c) Understand the computational strategies used for single-cell, isoform-level analyses and assess their suitability.

(d) Considering the state-of-the-art, select the optimal data types and computational methods and produce a set of general recommendations to tackle isoform analysis.

2. **Design an optimal single-cell RNAseq data processing pipeline to unlock downstream isoform-level analyses.**
   Using the insight gained from the previous evaluation (Aims 1.a and 1.b), the next goal is to define a methodology to process single-cell reads into a transcript-level expression matrix that is ready to be used for more complex analyses of alternative isoform biology. This task is divided into two secondary goals:

   (a) Define a combination of datasets and computational tools that results in reliable transcript isoform expression estimates using single-cell RNA-seq data.

   (b) Determine a strategy to detect significant expression changes of isoforms across multiple cell-types.

3. **Develop a novel analysis method to leverage single-cell RNA-seq data to gain valuable insight into isoform biology.**
   Considering the gaps in single-cell computational method development

delineated in Aim 1.c, we will devise an innovative single-cell isoform analysis strategy. The method will use cell and cell type-level isoform expression estimates to derive interesting clues regarding the functional importance of alternative splicing biology.

2.3 Contributions made during the thesis

2.3.1 Publications

The completion of this thesis has resulted in the production of two first-author and one first co-author manuscripts, all of which have been published or are awaiting publication in high-impact computational biology and bioinformatics journals:

- Arzalluz-Luque, Á., Conesa, A. Single-cell RNA-seq for the study of isoforms: how is that possible? *Genome Biology* **19**, 110 (2018).
- Arzalluz-Luque, Á., Salguero, P., Tarazona, S., Conesa, A. *acorde* unravels functionally interpretable networks of isoform co-usage from single cell data. *Nature Communications* **13**, 1828 (2022).
- Pardo-Palacios, F., Arzalluz-Luque, Á., Kondratova, L., Salguero, P., Mestre-Tomás, J., Amorín, R., Estevan, E., Liu, T., Nanni, A., McIntyre, L.M., Tseng, E., Conesa, A. SQANTI3: curation of long-read transcriptomes for accurate identification of known and novel isoforms. *Accepted for publication in Nature Methods*.

The contents of the present thesis have mostly been extracted and adapted from these three publications. However, during this PhD project, I have also become involved in multiple collaborative projects, which have resulted in the publication of the following papers:

- de la Fuente, L., Arzalluz-Luque, Á., Tardáguila, M., del Risco, H., Martí, C., Tarazona, S., Salguero, P., Scott, R., Lerma, A., Alastrue-Agudo, A., Bonilla, P., Newman, J.R.B., Kosugi, S., McIntyre, L.M., Moreno-Manzano, V., Conesa, A. tappAS: a comprehensive computational framework for the analysis of the functional impact of differential splicing. *Genome Biology* **21**, 119 (2020).

- Arzalluz-Luque, Á., Cabrera, J.L., Skottman, H., Benguria, A., Bolinches-Amorós, A., Cuenca, N., Lupo, V., Dopazo, A., Tarazona, S., Delás, B., Carballo, M., Pascual, B., Hernan, I., Erceg, S., Lukovic, D. Mutant PRPF8 causes widespread splicing changes in spliceosome components in retinitis pigmentosa patient iPSC-derived RPE cells. *Frontiers in Neuroscience* **15**, (2021).

- Tarazona, S., Arzalluz-Luque, Á., Conesa, A. Undisclosed, unmet and neglected challenges in multi-omics studies. *Nature Computational Science* **1**, 395–402 (2021).

2.3.2 Conferences

During the thesis, I have additionally presented my work at several conferences in the form of posters and short talks:

- Florida Genetics Symposium. Gainesville, Florida, United States. October 2017.

"Single-cell RNAseq for the study of isoforms: an analysis of current limitations". **Poster**.

- Bioinformatics@Valencia. Valencia, Spain. July 2018.
  "Single-cell RNAseq for the study of isoforms: how is that possible?". **Poster**.

- V Meeting of PhD Students of the Polytechnic University of Valencia. Valencia, Spain. July 2018.
  "Single-cell RNAseq for the study of isoforms: how is that possible?". **Poster**.

- 2nd International Caparica Conference in Splicing (Splicing 2018). Caparica, Portugal. July 2018.
  "Single-cell RNAseq for the study of isoforms: how is that possible?". **Short talk**.

- XIV Symposium on Bioinformatics (JBI 2018). Granada, Spain. October 2018.
  "Single-cell RNAseq for the study of isoforms: how is that possible?". **Poster**.

- ISMB/ECCB 2019 (27th Conference on Intelligent Systems for Molecular Biology and 18th European Conference on Computational Biology).
  "Measuring isoform co-expression in single-cell RNAseq successfully decodes splicing coordination as a key determinant of neural cell type identity". **Short talk**.

- Advances in Computational Biology (AdvCompBio). Barcelona, Spain. November 2019.
  "Measuring isoform co-expression in single-cell RNAseq data to decode splicing coordination". **Poster**.

- 20th Genome Informatics. Virtual (based in Cambridge, United Kingdom). September 2020.
  "Measuring isoform co-expression in single-cell RNA-seq to decode splicing coordination". **Short talk**.

- Earlham Institute Single Cell Symposium. Virtual (based in Norwich, United Kingdom). September 2020.
  "Measuring isoform co-expression in single-cell RNA-Seq data successfully decodes splicing coordination as a key determinant for neural cell type identity". **Short talk**.

- CSHL Genome Informatics 2021. Virtual (based in Cold Spring Harbor, New York, United States). November 2021.
  "Combining long-reads and single-cell RNA-Seq to infer isoform co-expression networks reveals splicing regulation as a key determinant in cell type identity". **Short talk**.

- ECCB 2022 (21st European Conference on Computational Biology). Sitges, Spain. September 2022.
  "acorde unravels functionally interpretable networks of isoform co-usage from single cell data". **Poster**.

2.3.3 Software development

The scientific advances made during this thesis have resulted in the development of acorde, an R package for the analysis of isoform co-usage networks in single-cell RNA-seq data, and the novel Filter and Rescue modules in the SQANTI3 software:

- Arzalluz-Luque, A. acorde: unraveling functionally interpretable networks of isoform co-usage from single cell data. *GitHub*.
  URL: https://github.com/ConesaLab/acorde
  DOI: https://doi.org/10.5281/zenodo.6341636 (2022)

- Pardo-Palacios, F., Tseng, E., Arzalluz-Luque, Á., Kondratova, L., Amorín, R., Liu, T. SQANTI3: a tool for the curation of long-read transcriptomes. *GitHub*.
  URL: https://github.com/ConesaLab/SQANTI3

In addition, I have made contributions to the maintenance documentation and development of the tappAS and MOSim software tools:

- de la Fuente, L., Tardaguila, M., del Risco, H., Salguero, P., Arzalluz-Luque, Á., Tarazona, S., Conesa, A. tappAS: a user-friendly application to analyze the functional implications of alternative splicing. *GitHub*.
  URL: https://github.com/ConesaLab/tappAS
  DOI: doi:https://doi.org/10.5281/zenodo.3751009 (2020)

- Monzó, C., Martínez-Mira, C., Febbo, A., Arzalluz-Luque, Á., Conesa, A., Tarazona, S. MOSim: bulk and single-cell multiomics data simula-

tor in R. *GitHub*.
URL: https://github.com/ConesaLab/MOSim

2.3.4 Undergraduate and Master's thesis supervision

I have participated as an experimental tutor in the supervision of the following students:

- Eva Estevan Morió: "Evaluation and improvement of quality control methods for long read-defined transcriptomes".
  Bachelor's Degree (BSc) in Biotechnology
  Universitat Politècnica de València
  2021/22

- Arianna Febo: "scMOSim: a method for the simulation of single-cell multimodal data".
  Master's Degree (MSc) in Bioinformatics for computational genomics
  University of Milan & Polytechnic University of Milan
  2021/22

2.3.5 Teaching

University courses

- Course: *In silico* studies in biomedicine.
  Teaching unit: Multiomics integration methods (4 hours).
  Master's Degree (MSc) in Bioinformatics

Universitat de València
2021/22 2022/23 2023/24

- Course: Multiomics integrative analysis.
  Teaching unit: Single-cell multiomics and data integration (4 hours).
  Master's Degree (MSc) in Omics Data Analysis and Systems Biology
  Universidad de Sevilla & Universidad Internacional de Andalucía
  2021/22 2022/23 2023/24

Private courses

- Course: "Transcriptomics in biomedicine: expression analysis and databases" (9 teaching hours). Universidad Católica de Valencia (2019).
- Course: "Single-cell data analysis" (5 teaching hours). INTERCEPT-MDS, European Union Innovative Training Network, BioBam Bioinformatics, Valencia (2023).

Conference workshops and tutorials

- Conference: Intelligent Systems for Molecular Biology 2020 (ISMB2020), virtual (based in Montreal, Canada).
  Tutorial: "Full-Length RNA-Seq Analysis using PacBio long reads: from reads to functional interpretation" (4 hours).
- Conference: 19th European Conference on Computational Biology (ECCB2020), virtual (based in Sitges, Spain).

Tutorial: "Full-Length RNA-Seq Analysis using PacBio long reads: from reads to functional interpretation" (3 hours).

Chapter 3

Evaluation of the limitations of single-cell RNA-seq technologies for the study of isoforms

Chapter adapted from Arzalluz-Luque, Á., Conesa, A. Single-cell RNA-seq for the study of isoforms: how is that possible? *Genome Biology* **19**, 110 (2018).

3.1 Introduction

Next-Generation Sequencing (NGS) technologies, particularly RNA-sequencing (RNA-Seq), have enabled the study of Alternative Splicing (AS) on a large scale. Early publications in the field of transcriptomics showed high levels of tissue-specific and developmentally regulated AS events [9, 10, 92–94], which was interpreted as an extra layer of phenotypic complexity. Since then, RNA-Seq has allowed the characterization of an increasing number of AS events with well-established roles in biological processes (thoroughly reviewed in [64, 65, 76, 92, 95–97]). All in all, this has set the notion of alternative splicing as a complex, tightly regulated, functionally-relevant process, although still poorly understood on a global scale.

In contrast to the high abundance of bulk-level AS studies, single-cell studies profiling isoform variability were scarce at the time when this thesis started (table 3.1). Transcriptome-level analyses of isoforms had mostly been performed as a part of single-cell RNA-Seq (scRNA-Seq) gene expression publications [98, 99] or in bulk studies of isoform diversity [100], but merely as a proof-of-concept. Usually, the aim of these studies was not to address single-cell isoform diversity, but to test the performance of the experimental protocols or computational tools in this scenario. In such a limited frame, the former studies accomplished identification of only a small number of above-noise splicing differences among single cells and lacked in-depth evaluation of results. In addition, for some years, only methods developed for RNAseq, mainly *mixture of isoforms* (MISO) [81], were used in single-cell isoform research [98, 101], and it was not until recently that computational strategies tailored to the particularities of single-cell RNAseq began to appear [102–104]. Notably, the use of short-read sequencing and the unavailability of tools for comprehensive isoform structure analysis have limited most research to solely quantification of exon inclusion levels [98, 99, 101, 102, 104] or targeting specific regions of the transcripts —that is, alternative polyadenylation (polyA) sites [105] or transcription start sites (TSS) [106]. At the same time, long-read sequencing technologies had started being applied to single cells and succeeded in characterizing full-isoform structures, but only for a limited number of cells and genes [53, 54].

Contrarily to what might be suggested by this gap in the literature, daring to go beyond the bulk is essential to answer some of the questions concerning the expression patterns of alternative isoforms. In spite of their limitations, some of the above-cited scRNA-Seq studies found unexpected levels of heterogeneity in isoform expression mechanisms in single cells [98, 101, 102, 107], raising the question of whether these patterns constituted an additional layer of biological complexity or were solely a result of the stochastic functioning of the AS machinery. There was no doubt at the time that performing more comprehensive single-cell isoform studies was key to better understand post-transcriptional regulation; however, the question of whether -and to what extent- taking isoform transcriptomics to the single-cell level was feasible was far from being answered.

Considering all this, and before tackling the main aim of this thesis (described in section 2.2), it was mandatory for us to make an assessment of how these two elements -isoform-level transcriptomics and single-cell data-could be pieced together. In order to achieve this, we performed four evaluations, the results of which will be described in the next sections of this chapter. First, we observed the properties of isoform expression to define the ideal conditions under which single-cell isoform studies would need to be performed, given the technologies that were available at the time (section 3.2). Next, we investigated how current studies fulfilled these conditions, measuring whether and how they were affected by the technical constraints of single-cell RNAseq data (section 3.3). To understand the field's gaps regarding method development, we also performed a revision of extant computational methods for scRNA-seq isoform analysis (section 3.4). Finally, we put a simulation strategy into place to better understand the kind of limitations that we would run into when using available experimental protocols and datasets (section 3.5). All together, this insight was essential to design the research strategies that resulted in the successful development of this project, as will be laid out in section 3.6 of this chapter.

3.2 Ideal conditions for single-cell isoform studies

In order to perform high-quality isoform expression analyses, transcriptomics datasets are required to meet a series of requirements. These are linked to the properties of alternative isoforms, both regarding their structural complexity and the nature of changes in their expression level. In the case of scRNA-seq, the challenges associated to isoform studies add to the constraints and biases intrinsic to this type of data, making the issue even more complex to address. Considering these potential synergies and the state-of-the-art at the time when this project was initiated, we defined a series of conditions that would be important for single-cell RNAseq isoform studies.

3.2.1 Full-length transcript capture

Isoform diversity is determined by the number of exons, introns, transcription start (TSS) and termination sites (TTS) and alternative donor/acceptor sites that are contained in a gene, but more importantly, by the different combinations of them that are expressed as transcripts (see Chapter 1). Hence, each event is likely to be present in several different isoforms. Partial sequencing of the transcript will naturally overlook a fraction of the events, and may make it impossible to distinguish some of a gene's isoforms. Such is the case for Illumina library preparation protocols that include UMIs, in which usable reads come only from the 3' (or 5') end of transcripts, where the UMI is attached. These methods therefore prevent isoform discrimination when differences are not comprised in this part of the sequence. As a result, a trade-off is established, by which isoform studies would require to give up on UMI usage in favour of strategies that provide full-length transcript information but are likely to suffer from technical biases, such as Smart-seq and Smart-seq2.

Even if conducted using full-length protocols, short read-based isoform studies still suffer from limitations due to the structural complexity of isoforms, since the combination of events generally spans distances larger than the base pair span covered by a single Illumina read. As a result, quantification tools often struggle to accurately assign short reads to the different isoforms, limiting quantification to the exon/event level. Long read-technologies, which permit sequencing of the full transcript in one read, could therefore constitute an attractive alternative to facilitate transcript identification and quantification in single-cell isoform studies.

3.2.2 Sequencing depth

The low amount of starting material in single-cells hinders capture efficiency and causes the appearance of drop-outs, that is, the identification of a gene as unexpressed due to missing of transcripts during reverse transcription [29, 98]. This is more likely to affect genes that are lowly expressed [108], for which zero-expression values cannot be distinguished from absence of expression. Therefore, mRNA capture efficiency sets a limit to the number of transcripts that can be detected in single-cell RNAseq. For transcripts expressed above the detection limit, high sequencing depth is the key to maximize sensitivity, that is, the probability to capture a particular transcript in the cell [25]. At the time when this study was carried out, consensus in the single-cell RNA-seq field was that saturation was achieved at approximately 1 million Illumina reads per cell [58, 109].

However, the properties of isoform expression require slightly different considerations concerning capture efficiency and sequencing depth. First, isoforms within a gene are very differently expressed, typically showing, for a particular cell type, a major (i.e. highly expressed) isoform and several alternative isoforms with lower expression values [10, 110]. This makes alternative isoforms more sensitive to drop-outs, and thus the level of isoform diversity per gene can easily be underestimated. In addition, when the study presented in this chapter was initiated, the saturation limit of single-cell RNAseq had been evaluated regarding library complexity at the gene level. Given the nature of alternative isoform expression, reaching saturation at the isoform level could potentially require more than 1 million reads per cell. Regarding comparisons across conditions/cell types, lower depth might suffice when aiming to detect major isoform switches -that is, changes in the most highly expressed isoform-, as long as sequencing is deep enough to capture the most highly expressed isoform. However, isoform expression changes often comprise subtle modifications in a gene's isoform expression ratio, which would require higher sensitivity.

3.2.3 Cell throughput

Sequencing a high number of samples -or, in the case of scRNA-seq, cells-increases the statistical power of a transcriptomic analysis, that is, the ability to characterize expression patterns across samples with high confidence [25]. Sequencing large sets of cells therefore has the potential to yield significant advances in our understanding of isoform expression at the single-cell level.

When the research presented in this chapter was published, the development of microfluidics and droplet-based sequencing systems had radically scaled UMI-based library preparation protocols, enabling parallel processing of thousands of cells [14, 32]. A particularly relevant milestone was the commercial implementation of the inDrop protocol by 10X Genomics, which achieved a throughput of up to 250,000 cells in a single experiment [111]. Even though throughput has continued to increase ever since, these methods are not only restricted to coverage of the 3' end, but also hindered by their low sensitivity, which results in detection of fewer genes -and isoforms- per cell. Conversely, SMART-based alternativeswarrant high sensitivity, with detection rates of up to 20,000 genes per cell as reported by Zieghain et al. in their 2017 comparative analysis [25]. However, these protocols have cost and throughput limitations, since they require manual preparation of libraries in microwell plates that, in addition, operate with larger reagent volumes. To reduce labour, the Smart strategy was implemented on the Fluidigm C1 instrument for parallelization and automation of the library preparation process, although this option was still limited in terms of cost per cell. As a result, short-read, isoform studies published at the time were far from reaching the throughput achieved by UMI-based alternatives [38]. Throughput limitations in the context of these studies are further discussed in section 3.3.3.

When evaluating this aspect for long reads, even though publications were scarce, studies performed at the time showed that cell throughput constituted a severe constraint when applying this technology at the single-cell level. Specifically, the first long-read single-cell publications presented data from <10 cells [53, 54]. Since flow-cells yield a limited amount of total reads per run, we found this to be associated to the design of PacBio and ONT technologies. Although trivial for a bulk population, where only a few samples are sequenced in each flow-cell, multiplexing many cells inevitably means limiting cell-level sequencing depth. Meanwhile, increasing the number of flow-cells came at a high cost at the time (further discussed in section 3.3.3 below).

3.2.4 Sequencing errors and artifacts

Sequencing errors are generated due to base miscalls during sequencing, while artifacts usually appear during library preparation and thus comprise products that were not originally present in the sample [50]. Both of these issues have been reported to negatively impact isoform characterization [50] and should therefore also be considered at the single-cell level.

Regarding sequencing errors, whereas Illumina sequencing has a long-standing history of high sequencing accuracy (0,005% error rate), long-read technologies present sequencing error rates that are several orders of magnitude higher. In particular, when this aspect was assessed at the start of the present thesis project, error rates for consensus PacBio sequences were in the range of 2-5% [50], whereas values reported for ONT went up to 7% [48, 51]. Sequencing errors often introduce false alternative donor or acceptor splice sites, many of them non-canonical, which results in the identification of false positive isoforms that are incorrectly characterized as novel transcripts [50, 112]. In the context of scRNA-seq, another relevant problem would be the resolution of cell barcodes and UMIs. Given that the technology relies on multiplexing strategies to minimize batch effects as well as on UMI counts to eliminate amplification bias, occurrence of errors in these regions would add an extra challenge to analysis pipelines.

Long-read sequencing errors are computationally corrected using three different strategies: (a) generation of a consensus sequence (as explained in section 3.1), (b) clustering of reads belonging to the same transcript and (c) complementary short-read sequencing, combining the accuracy of Illumina with the scaffolding potential of long-reads. Nevertheless, considering the state-of-the-art in the beginning of this thesis, compatibility with single-cell level studies could only be ensured in (a). First, the sequencing depth constraints discussed in this chapter preclude effective clustering, since high sensitivity is required to allow multiple instances of the same isoform to be captured during sequencing. Regarding complementary sequencing, no protocols in which the same cell was sequenced using two different technologies were available at the time. As a result, some errors may survive computational correction and result in missed barcodes and false-positive isoform discovery.

Artifacts generated during reverse transcription are also highly relevant to long-read isoform studies. First, intra-priming events in genes with internal poly-A sequencescan generate shorter cDNA artifacts that may be mistaken with isoforms with an upstream TTS [113]. Additionally, mRNA molecules form secondary structures that can prevent access of the reverse transcriptase to certain fragments of the sequence, favouring template switching and skipping of these segments, which will appear as alternatively spliced isoforms [114]. Long-read technologies have been shown to accumulate this kind of artifacts. These, in combination with the higher prevalence of sequencing errors, add to the generation of false positive isoforms. To detect and solve this type of issues, our group developed SQANTI [50], the first tool to control for the overestimation of novel isoforms in bulk PacBio RNA-seq data, although neither the validity of this tool nor the extent of these limitations in single-cell studies had been assessed when we first posed the research questions that are laid out in this chapter.

These four requirements (full-length transcript capture (a), high capture efficiency and sequencing depth (b), high number of cells sequenced (c) and low occurrence of errors and artifacts (d)) and how their technological and experimental limitations impact isoform detection in single cells are summarized in figure 3.1. These provide a framework to assess the success of single-cell isoform studies, and were latter used as criteria to assist the selection of our own data and methods in order to proceed with this research project.

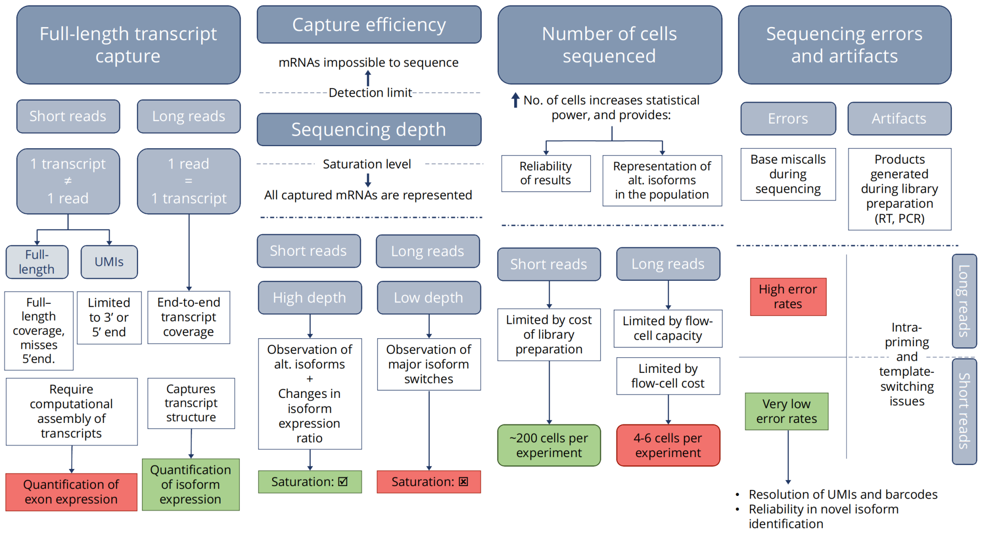

Full-length transcript capture
Short reads
Long reads
1 transcript ≠ 1 read
1 read = 1 transcript
Full-length
UMIs
Full-length coverage, misses 5'end.
Limited to 3' or 5' end
End-to-end transcript coverage
Require computational assembly of transcripts
Captures transcript structure
Quantification of exon expression
Quantification of isoform expression
Capture efficiency
mRNAs impossible to sequence
Detection limit
Sequencing depth
Saturation level
All captured mRNAs are represented
Short reads
Long reads
High depth
Low depth
Observation of alt. isoforms + Changes in isoform expression ratio
Observation of major isoform switches
Saturation: ☑
Saturation: ☒
Number of cells sequenced
No. of cells increases statistical power, and provides:
Reliability of results
Representation of alt. isoforms in the population
Short reads
Long reads
Limited by cost of library preparation
Limited by flow-cell capacity
Limited by flow-cell cost
~200 cells per experiment
4-6 cells per experiment
Sequencing errors and artifacts
Errors
Artifacts
Base miscalls during sequencing
Products generated during library preparation (RT, PCR)
High error rates
Very low error rates
Intra-priming and template-switching issues
Long reads
Short reads
Resolution of UMIs and barcodes
Reliability in novel isoform identification

Figure 3.1: Summary of limitations of the four ideal conditions for successful studies of scRNA-seq isoforms. From left to right, the importance and current limitations of full-length transcript sequencing, capture efficiency and sequencing depth, the number of cells sequenced, and sequencing errors and artefacts for isoform detection are presented in the diagram.

3.3 Expectations meet reality: what has been and remains to be done in single-cell isoform studies

At the time this thesis started, very few single-cell studies featuring isoform-level transcriptomics analyses had been published, all of which are summarized and compared in table 3.1. In particular, these had mainly been performed as a part of single-cell gene expression publications [98, 99] or in bulk studies of isoform diversity [100], but merely as proof-of-concept. Usually, the aim of these studies was not to address single-cell isoform diversity, but to test the performance of experimental protocols or computational tools in this scenario. In such a limited frame, these studies accomplished identification of only a small number of above-noise splicing differences among single cells and lacked in-depth evaluation of results. Moreover, it took a long time for computational strategies tailored to the particularities of scRNA-seq to appear [102–104], and methods developed for RNAseq (mainly "mixture of isoforms", i.e. MISO [81]) were most commonly used in initial single-cell isoform research [98, 101]. Notably, the use of short-read sequencing and the unavailability of tools for isoform structure analysis limited most research to quantification of exon inclusion levels [98, 99, 101, 102, 104], alternative polyA [105] or TSS usage [106]. Meanwhile, the first studies applying long-read technologies to single cells appeared, succeeding in characterizing full-isoform structures [53, 54], although on a limited number of cells and transcripts.

The scarcity of experimental and computational background made it challenging to design adequate analysis strategies without a previous in-depth evaluation of the literature. Therefore, after defining the theoretical framework of the potential pitfalls of cell-level isoform transcriptomics (section 3.2), we set out to review the state-of-the-art to explore how these limitations had been encountered in the field. Ultimately, this allowed us to devise a comparison of the performance of the three cited strategies -UMI-based, full-length and long-reads- in the isoform context and select the one that was optimal for the present research project.

3.3.1 Full-length transcript capture and isoform continuity

As it has been pointed out (section 3.2.1), employing full-length library preparation methods to generate the data constitutes a major requirement to successfully tackle isoform transcriptomics. In line with this, most studies published at the time when this evaluation was performed relied on the Smart-seq [99] and Smart-seq2 [34] protocols (table 3.1), all of which fall into this category. These studies, however, presented limitations regarding both evenness and completeness of coverage along the transcript, due to 3' bias, and isoform expression quantification accuracy, due to their incompatibility with UMIs. The nature and extent of these limitations is discussed in the two sections below.

Sequencing technology	Reference	Focus of study	Full-length isoforms?	Computational method	Aim	Organism, cell type	Library preparation	Feature or event targeted
Illumina	Ramsköld et al. [99]	scRNA-seq, genes	No	MISO (bulk RNA-seq)	Experimental method development	Human, cancer cells	Smart-seq	Exon inclusion quantification
	Shalek et al. [101]	scRNA-seq, genes and isoforms	No	MISO (bulk RNA-seq)	Cell heterogeneity in immune response	Mouse, BMCDs	Smart-seq	Exon inclusion quantification
	Zhang et al. [100] Data: Shalek et al. [101]	Bulk RNA-seq, isoforms	No	WemIQ (bulk RNA-seq)	Computational method development	Mouse, BMCDs	Smart-seq	Bias in differential isoform detection
	Marinov et al. [98]	scRNA-seq, genes and isoforms	No	Pervouchine et al. [115] (bulk RNA-seq)	Expression heterogeneity	Mouse, lymphoblas-toid cells	Smart-seq	Novel splice junctions, exon inclusion quantification
	Velten et al. [105]	scRNA-seq, isoforms	No	BATBayes	3'UTR variability among genes/cells	Mouse, ESCs	BATSeq	Alternative poly(A) sites
	Welch et al. [103] Data: Buettner et al. [13]	scRNA-seq, isoforms	No	SingleSplice	Computational method development	Mouse, ESCs	C1/Smart-seq	Differential isoform usage detection
	Karlsson et al. [106] Data: Zeisel et al. [15]	scRNA-seq, isoforms	No	Alignment to FANTOM5 database (CAGE)	Expression heterogeneity	Mouse, brain cells	C1/STRT-seq	Alternative TSS
	Song et al. [102]	scRNA-seq, isoforms	No	Expedition	Computational method development	Mouse; iPSCs, NPCs and MNs	Smart-seq, Smart-seq2	Exon inclusion quantification
	Huang et al. [104] Data: Wu et al. [109] Scialdone et al. [116]	scRNA-seq, isoforms	No	BRIE	Computational method development	Mouse, B1 cells	Smart-seq2	Exon inclusion quantification
Oxford Nanopore	Byrne et al. [53]	scRNA-seq, isoforms	Yes	Mandalorion	Computational method development	Mouse, B1 cells	Smart-seq2	TSS, TTS, exon inclusion, intron retention, alt. 3' and 5' splice sites
PacBio	Karlsson and Linnarson [54]	scRNA-seq, isoforms	Yes	Self-designed pipeline	Expression heterogeneity	Mouse, oligo-dendrocytes and VLMCs	C1/STRT-seq	TSS, TTS, exon inclusion, alt. 3' and 5' splice sites

Table 3.1: Comparative summary of scRNA-seq isoform studies published at this thesis' start time. Studies are classified by focus (i.e. bulk/single-cell RNA-seq and gene/isoform expression. Only computational methods used for isoform identification/quantification are specified. When not single cell specific, the original technology for which the computational method was developed is indicated. Studies are considered full-length if isoforms were reconstructed end-to-end, regardless of library preparation. When specified, studies were performed on data generated by other authors. Targeted feature/event refers to the approach taken to study isoform diversity or to the tackled aspect of transcript diversity.

Coverage-related limitations

The Smart-seq protocol [99] constituted the first significant improvement in transcript sequence coverage in comparison to prior methods, which showed preferential amplification of the 3' [11] or 5' [31] ends of mRNAs. Using an RNA dilution to mimic the amount of RNA in a single, eukaryotic cell (approximately 10 ng), Ramsköld et al. accomplished a remarkable 40% coverage of the 5' end. As a result of this, 25% of detected multi-exon genes were reportedly covered end-to-end, whereas twice as many differentially spliced exons were detected among cells when compared to previously published data [17].

Conversely, some studies covered mechanisms other than AS, which are differently affected by coverage limitations. In particular, Karlsson et al. targeted alternative TSS [106] using STRT-seq [31], whereas Velten et al. focused on alternative TTS/poly-A sites using a novel 3'-targeted method [105]. For these TSS and TTS-associated events, it could be assumed that even end-to-end transcript coverage was not essential. However, as it turns out, 3' bias could still compromise the amount of data that is usable for this purpose. For instance, Karlsson et al. [106] only obtained a rate of 14% of 5' end-aligned molecules (i.e. reads collapsed by UMI) from STRT-seq data, a manifestation of 3' end bias persistence in short-read sequencing.

In spite of the potential of these findings, many of the publications that addressed single-cell splicing at the time (table 3.1) constituted proof-of-concept studies. These publications aimed to demonstrate that AS could be studied using scRNA-seq, be it in the context of computational method validation [100, 102–104] or in order to better understand how the AS signal is affected by technical variability in single-cell data [98]. In addition, research efforts to characterize alternative splicing patterns, such as the pioneering paper by Shalek et al. [101], suffered from the limitations of short read full-length protocols regarding isoform-level resolution (described in section 3.2.1), being therefore limited to quantification of exon inclusion/exclusion. In line with this, most of the first set of single cell-specific computational methods for the study of AS dynamics were designed to capture exon-level differences [102, 104].

Remarkably, two isoform studies published in 2017 were the first to allow end-to-end characterization of transcript variants in single-cells owing to the application of the Oxford Nanopore [53]and PacBio [54] sequencing technologies. Using ONT, Byrne et al. [53] identified an impressive amount of alternatively spliced genes (696 using alternative TSS/TTS, and 354 undergoing exon inclusion/exclusion) in B1a cells. Although the expression levels of these isoforms were not measured -likely because of sequencing depth limitations-, the study demonstrated that long-reads could identify larger numbers of AS events than short-reads. In addition, the structure of complex isoforms (i.e. transcripts in which alternative TSS/TTS and alternative splicing occur simultaneously, as defined by the authors [53]) belonging to 169 genes was identified, an unprecedented level of isoform structure resolution in single-cells.

Quantification accuracy limitations

Having assessed the incompatibility of Smart-seq protocols with UMIs and its implications for the study of isoforms (see section 3.2.1), our next question was whether there were other strategies available to mitigate technical sources of variation in full-length, single-cell data. In the first Smart-seq studies, however, no technical variability correction strategy was put in place. As a result, it cannot be excluded that the above-cited results by Ramsköld et al. [99] suffered from some form of technical bias. Shortly after, however, Shalek et al. [101] incorporated validation of results in a dual manner: RNA-FISH, to compare the isoform ratio differences of two candidates, and a set of additional UMI libraries to exclude the possibility of PCR leading to an overestimation of expression. Validation was successful for 89 highly expressed isoforms undergoing differential exon inclusion across the population.

In a later publication, Zhang et al. [100] tested WemIQ -a tool to detect differential exon inclusion in bulk RNA-seq- on the single-cell dataset by Shalek et al. Interestingly, WemIQ removed a great degree of the cell-to-cell heterogeneity from the data, which authors attributed to technical bias. Meanwhile, Shalek et al. had reported high levels of heterogeneity in alternative splicing [101]. The WemIQ results therefore raised the question of whether this variability was biological or technical. Alternatively, however, this could be indicative of bulk RNA-seq tools mistaking the higher biological variability in single cell data for technical noise, pointing towards the necessity to develop single-cell-specific methods.

A first conclusion to be raised was that, in scenarios where technical bias cannot be properly accounted for, it would be advisable to make a qualitative approximation to isoform variability. In other words, strong changes -such as major isoform switches- and broad splicing patterns may be detected with confidence even if quantitative results are affected by technical noise. For a more accurate assessment, we anticipated that the lack of UMIs would require independent validation of the quantitative results, such as RNA-FISH or qPCR (as in [101] and [102], respectively), or the addition of spike-in RNA for noise reduction in downstream analysis [29, 58]. Although the latter had already been suggested as a reliable enough option to normalize single-cell count data [117, 118], they had not been implemented in any already-published isoform studies. Another interesting alternative to UMIs was presented in a later study by Marinov et al. [98]. As a means of estimating noise-contributing factors, and in combination with spike-ins, authors implemented pool/split controls, produced via pooling several single cells and then splitting the RNA into equal amounts prior to library preparation. Pooling evened out biological differences between the cells and guaranteed that any variability observed will solely be technical, including PCR bias. Differences between controls could then be used to re-estimate cell-to-cell differences. Marinov et al. hereby succeeded to validate isoform switches in 282 multi-exon genes [98], however, no later studies of isoform diversity at the single-cell level used pool/split controls at the time.

Regarding quantification and long-reads, in contrast to the non-quantitative study by Byrne et al. [53], Karlsson and Linnarsson [54]specifically addressed quantification of isoform expression by optimizing a protocol combining PacBio sequencing with effective resolution of UMIs. In this particular study, however, lowly expressed isoforms are found to be rarely shared among cells, which aligns with previously described sequencing depth limitations 3.2.3.

3.3.2 Capture efficiency and sequencing depth

In our study of ideal conditions for single-cell isoforms, low capture efficiency was defined as one of the main drawbacks with regard to accurately reflecting transcript abundances in the cell (section 3.2.2). As well as being at the root to known properties of single-cell data, such as high sparsity and variability, problems with transcript capture also jeopardize the reproducibility of some observations, especially if the transcript's expression level is close to the detection limit. As a result, the biological relevance of expression measures or changes cannot be established with confidence. For instance, in Marinov et al. [98], most novel splice sites were observed in only one cell, with no way of knowing whether these were true novel isoforms or false positive observations. To venture whether or not this was the case, authors turned to previous studies showing that lowly expressed transcripts were more highly affected by technical noise [108]. Since these rare novel junctions mostly belonged to lowly expressed genes, they ruled them out as a technical artifact. Similarly, Karlsson et al. [106] observed a TSS co-expression pattern in which correlation decreased with transcript expression level. In this study, an improvement in capture rate was proposed as the solution to verify whether TSS expression was also correlated in lowly expressed genes. Interestingly, the cells had been sequenced to an average of 0.5 million reads per cell in the study that generated the data [15]. Even though sequencing depth is another major aspect of detection quality and expression estimation accuracy in single-cell data, these do not improve beyond the saturation threshold (section 3.2.2) and, therefore, higher capture efficiency would indeed have yield more benefits than deeper sequencing.

Regarding sequencing depth-related constraints in the isoform studies included in our assessment, it should be pointed out that numbers laid far above the consensus saturation limits that the field had set at the time. In fact, 20 to 40 million reads per cell had been obtained in most of these scRNA-seq isoform studies [99, 101, 102, 105]. Even so, no study had yet addressed how isoform complexity changes with sequencing, and it was unknown whether this constituted as much of an excess of information as it seemed to be for genes. Regardless, it was clear that shallow sequencing prevented detection of multiple isoforms per gene. This was brought to light by Welch et al. in the SingleSplice publication [103], where the number of detected splice variants was shown to increase with sequencing depth. Another indicator of unsaturated libraries reported in this study was the detection of cells exhibiting less splice variants than genes [103]. Of note, this is an even more pressing issue in TSS, TTS or event-specific studies, where high depth is required to ensure that a sufficient proportion of the reads cover the region of interest. To illustrate, only 25% of total reads per cell in the study by Velten et al. [105] included polyA sites. A similar problem was encountered in the investigation of alternative TSS by Karlsson et al. [106], in which 3' bias significantly interfered with the amount of reads mapping to the 5' end (see section 3.3.1).

In contrast to short-reads, where isoform-level saturation could be anticipated to be higher than expected, it was reasonable to believe the saturation threshold for long-reads to be below 1 million reads/cell, given that reads correspond to a full-length transcript and could potentially be estimated as the number of transcripts in the cell lysate. Nonetheless, sequencing depth limitations were found to be exacerbated by the sequencing depth vs cell number trade-off described in 3.2.3. As an example, Byrne et al. [53] obtained roughly 57,000 to 128,000 reads per cell by multiplexing of 4 cells on a single MinION flow-cell, with authors reporting difficulties in the identification of low-abundance transcripts as well as the impossibility to use spike-ins. In the case of Karlsson and Linnarsson's long-read study [54], a total of 6 single-cell libraries were pooled and sequenced in a single PacBio RSII run. In this case, 61% of UMIs were observed only once per transcript, which was interpreted as an indicator of sequencing depth limitations. Moreover, the results of both studies should be taken with additional caution, as the large number of PCR cycles required to generate sufficient cDNA for sequencing may lead to repetitive sequencing of PCR duplicates of the most abundant transcripts, further limiting the amount of isoform diversity that is eventually captured in the experiment. These results revealed that the average read throughput advertised by both ONT and PacBio at the time was overestimated, and that sequencing depth limitation exhibited by long-reads was a technological one.

In spite of this, it would have been naive for us to compare the isoform detection potential of short vs long reads solely in terms of sequencing depth. Even though short-read protocols at the time presented more sensitivity and better detection limits, the possibility that shallow long-read sequencing was preferable under some circumstances could not be ruled out. We thus concluded that, since long-read technologies could detect fewer genes, but provided better resolution at defining isoforms for them, a trade of quantity for quality might be worth considering for single-cell isoform studies.

3.3.3 Cell throughput

As discussed in section 3.2.3, the ability to ensure access to large cell populations for scRNA-seq increases the chances of recurrent detection of rare events, such as novel isoforms or splice sites, which is ultimately related to statistical power. Cell throughput was thus recurrently discussed in single-cell isoform studies, especially considering the scalability constraints of SMART-based protocols. Welch et al. [103], for instance, observed that few splice variants were detected in more than one cell, and highlighted that a higher frequency of detection would have been obtained by sequencing a larger population. Related observations made by Marinov et al. [98], i.e. that the majority of novel splice sites were present only in a single cell, could have been similarly validated. Concerning not only characterization of events, but also quantitative patterns of alternative splicing such as the ones defined by Song et al. [102], cell throughput was also likely to play a relevant role. In this study, authors defined splicing bimodality and unimodality rates in mouse neural development in 200 cells, which was among the largest throughputs that we found in our evaluation (table 3.2). Even though no estimates of the minimal amount of cells necessary to confidently estimate isoform expression had been proposed at the time, deep, SMART-based Illumina sequencing was only possible in the range of the hundreds at the time [38]. Reassuringly, Song et al. set a stringent minimal coverage threshold for splice junctions by which only events covered by at least 10 reads were included in subsequent analysis [102]. As a result,200 to 10000 AS events per cell were identified, which suggested that appropriate data filtering could still yield some solid isoform-level insight in spite of throughput limitations.

In the case of long-read technologies, the sequencing depth and budget restrictions that existed in the early days of single-cell sequencing posed a limitation to the number of cells that could be processed. This resulted in a dual reads per cell and cells per experiment trade-off. As an estimate, we considered the MinION experiment by Byrne et al. [53], including 4 single cells per flow-cell, to be the current maximum capacity of the instrument. Based on this, for a 100-cell experiment, approximately 25 MinION flow-cells (which are disposable and could be used in up to runs of 72h) would have

Study	Ramsköld et al. [99]	Shalek et al. [101]	Marinov et al. [98]	Velten et al. [105]	Welch et al. [103]	Karlsson et al. [106]	Song et al. [102]
Data	-	-	-	-	Buettner et al. [13]	Zeisel et al. [15]	-
Cell no.	12	18	15	144	96	2816	206
Method	Smart-seq	Smart-seq	Smart-seq	BATSeq	C1/Smart-seq	C1/STRT-seq	C1/Smart-seq

Table 3.2: Number of cells sequenced in published short-read, single-cell isoform studies. For studies that re-used published data, data source studies are shown.

been necessary. Even though the MinION instrument was cheaper to acquire compared to bench sequencers such as those manufactured by PacBio ($1000 for a starter pack including 2 flow-cells and a reagent kit), one should note that the cost of the 23 extra flow-cells, plus any additional reagent kits necessary, would rapidly increase the budget to nearly prohibitive costs (source: https://store.nanoporetech.com/, visited December 2017).

3.3.4 Sequencing errors and artifacts

Sequencing errors and artifacts are particularly frequent in long-read technologies, where they prevent discrimination of true vs false positive isoforms (see section 3.2.4). For instance, differential or novel start and termination sites are hard to distinguish from degradation and reverse-transcription artifacts and, as a consequence, TSS and TTS are sometimes defined as nucleotide position ranges or bins. Moreover, given that some of these errors arise during reverse transcription, e.g. template-switching or intra-primming, artifact reads cannot be assigned to a true isoform using UMIs. Alternatively, spike-ins can be used to estimate error probabilities and correct them in sequencing data. Using this approach, Karlsson and Linnarsson [54] were capable of attributing an uncertainty of ±5bp to the premature termination of reverse transcription (i.e. 5' end variability), hence considering variation beyond this window to be true alternative TSS. Uncertainty in the identification of exon junctions was similarly characterized and corrected. Notably, Byrne et al. [53] used Illumina reads as additional support to curate ONT-defined isoforms, where novel splice junctions detected both in long and short reads were accepted as true. By splitting the cDNA from single-cells after library preparation and sequencing using both Illumina and ONT, they managed to generated both types of sequencing data, although this approach heavily relied on the extraction of a high amount of cDNA from the pooled libraries.

High error rates also interfere in barcode and UMI identification. In fact, Byrne et al. [53] reported the impossibility to use UMIs due to the high error rates of ONT sequencing. In order to be able to resolve them, authors suggested that UMIs longer than 30bp would be required, with the subsequent increase in RT and PCR artifacts that such long oligonucleotides would inflict. In contrast, Karlsson and Linnarson [54] managed to overcome sequencing erros in PacBio reads by correcting both reads and UMIs using circular consensus sequencing (CCS). It is interesting to keep in mind, concerning barcoding, that PacBio originally provided the users with a set of 384 barcodes that enabled multiplexing of samples, optimized for the technology's error model (source: SMRTlink v6.0.0; https://www.pacb.com/wp-content/uploads/SMRT_Analysis_Barcoding_Overview_v600.pdf). ONT, in spite of the 2D consensus system, relied on improvements on sequencing accuracy to incorporate UMIs and had not developed compatible barcodes for multiplexing at the time, hence the need for them to be designed by the user [53].

As it can be derived from this analysis, none of the scRNA-seq data types that were available when this project started fulfilled our four criteria for successful isoform studies (figure 3.2). Among them, we determined that SMART-based methods achieved the best balance, providing high sensitivity and capture efficiency in exchange for a reduction in statistical power due to their limited cell throughput. In addition, these methods achieved a good balance between the total number of accurately characterized isoforms and their potential for transcript-level expression quantification. However, the latter relies heavily on the computational method of choice, namely on the availability of tools that can assign short reads to the correct transcripts. Given that scRNA-seq-based AS and isoform analyses were still in its very early days, there was no consensus on the most robust approach to measure isoform usage changes. In order to better understand this aspect, we next reviewed the different software tools that were used to leverage Smart-seq data in published single-cell isoform studies, focusing on the assumptions that they relied on and the type of results that they produced.

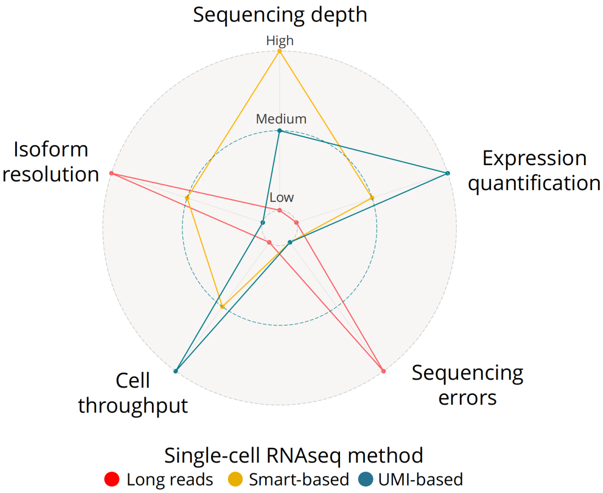


Figure 3.2: Qualitative performance of the three main scRNA-seq methods in the context of isoform transcriptomics. From the inside to the outside of the graph, the three dotted lines represent low, medium and high levels of each characteristic. The most prominent features of long reads (red), SMART-based methods (yellow) and UMI-based methods (blue) are shown.

	SingleSplice [103]	MISO [81]	BRIE [104]	Expedition [102]	RSEM [85]
Observation level	Gene	Exon	Exon	Exon	Isoform (full transcript)
Measure of expression	Alternatively spliced (yes/no)	PSI	PSI	PSI	Read counts per isoform
Single-cell specific	Yes	No	Yes	Yes	Unknown
Includes interpretation of changes	Yes	No	No	Yes	No

Table 3.3: Comparative summary of five computational approaches used to study splicing in scRNA-seq.

3.4 Navigating the computational landscape of single-cell isoform studies

Throughout this review, a number of published studies have been mentioned whose main focus was to present novel methods for the study isoform expression at the single-cell level (summarized in table 3.1). In this part of our study, we reviewed those tools that were developed specifically for short-read data, but also covered bulk-designed tools that have recurrently been applied to scRNA-seq analyses (MISO) and others that could be considered of interest for the future of the field (RSEM). These set of computational methods could be divided into three categories: (1) methods that detected alternatively spliced genes (i.e. SingleSplice [103]); (2) methods that worked at the event and exon levels (i.e. MISO [81], BRIE [104] and Expedition [102]) and (3) methods that provided a single expression value per transcript isoform (i.e. RSEM [85]) (table 3.3).

The first group of tools (1) included those methods that do not aim to discriminate all of the isoforms expressed by a gene, but rather to detect as many as possible based on what can be gathered from short-read data. To deal with these limitations, SingleSplice introduced the concept of Alternative Splicing Modules (ASM), which are combinations of events observed in aligned reads that generate a unique isoform or isoform fragment. The power of the ASM definition laid in the fact that isoforms that differed in junctions near the 5' end, where there is often a decrease in read coverage, were grouped under the same ASM and assigned a combined expression value. Once the tool had identified the different ASMs that belonged to each gene, it looked for cell-to-cell changes in the ratio of expression of two or more ASMs. Using this approach, genes could be flagged as alternatively spliced even if the exact transcript isoforms driving those changes could not be identified.

Among event-oriented methods (2), MISO was developed to detect alternative splicing in bulk RNA-seq, although it was the method of choice in most early single-cell studies that sought to obtain isoform-level insight [99, 101]. The tool uses reads aligned to splice junctions and a mixture model to estimate Percent Spliced In (PSI) values for alternatively spliced exons. PSI is defined as the fraction of mRNAs that represent isoforms where the exon is included, hence the metric depends on the number of reads aligning to the exon, the flanking constitutive exons and their junctions, but is also influenced by read counts across the bodies of other constitutive exons, which contain information on the abundance of both the exon-including and excluding isoforms. To incorporate the latter into the metric, the inference of PSI for each exon is treated as a Bayesian problem, and confidence intervals are used to evaluate the reliability of the PSI estimates. While BRIE [104] and Expedition [102] build on the same premises as MISO to assess expression at the exon level, these tools use new strategies to face challenges specific to single-cell data. In particular, they differ in the way they quantify events and in their approach to evaluate splicing differences across cells.

Regarding event quantification in BRIE, a mixture model approach similar to that of MISO is used for exons where read count is high. In addition, however, informative priors learned from the data are used in a Bayesian regression model in order to improve sensitivity and obtain accurate estimates in events where reads are scarce. Expedition, on the other hand, only uses junction-spanning reads for quantification, computing PSI as the proportion of reads covering the junctions of the alternative exon. In other words, Expedition PSI can be loosely interpreted as the percentage of transcripts per cell that include a given exon. Furthermore, the method is rather conservative when producing PSI estimates, only quantifying events covered by more than 10 reads, as opposed to the greedy approach in BRIE. Regarding the second differential aspect -that is, the detection of AS across cell types-, BRIE performs all possible pairwise comparisons between cells from different cell types, which can be computationally costly when high numbers of cells are analysed. Expedition, in contrast, avoids intensive calculations by instead classifying events into “modalities” according to their distribution of PSI scores among the overall cell population. The classification produced by Expedition can then be used to understand global trends for events of interest, as well as assess changes in these trends across cell types or conditions. We thus concluded that Expedition yielded more easily interpretable results than BRIE, as well as required less computational resources, which is an important aspect in a scenario where cell throughput was expected to increase significantly in the upcoming years.

Finally, among transcript-level quantification methods (3), the bulk-designed tool RSEM [85] incorporated a single-cell parameter option in a 2015 release which, when supplied, instructed RSEM to use a sparse prior to inform the underlying Expectation Maximization (EM) algorithm and account for scRNA-seq data properties when assigning reads to transcripts. Importantly, this tool was the only single cell-adapted tool available at the time to produce a single expression value per isoform. The new feature, however, had not yet been validated on scRNA-seq data, therefore raising the question of whether the expression estimates provided were are sufficiently accurate to be used for downstream analysis.

Choosing one of these tools over another arguably depends on the aim of the study. For instance, SingleSplice provided a general overview of the consistency of splicing for all multi-isoform genes in a given population, which we found could be useful when characterization of specific events of isoforms was not required. In turn, we found that Expedition was the most suitable method to obtain event-level information on splicing changes in a given population of cells. Finally, since RSEM was the only available tool that provided isoform-level expression estimates, it was found as the most suitable for full-length applications, even though its performance had not been tested on single-cell data.

3.5 Assessing the theoretical limits of current technologies for single-cell isoform studies

Having defined an optimal framework for alternative isoform detection and discussed results obtained in isoform-level studies using scRNA-seq, we set out to verify the extent to which single-cell isoform characterization was feasible given the state-of-the-art. To this end, we ran two sets of simulation experiments where single-cell transcriptomics data from short and long read sequencing was emulated. These experiments were designed to capture two features of single-cell sequencing technologies that dramatically affect isoform detection: full-length transcript coverage, in the case of short-read data, and the trade-off between cell throughput and sequencing depth per cell, in the case of long-read data. These simulations were based on bulk datasets [50] and on the simulation software available at the time [119], therefore, the technical biases of single-cell RNA-seq could not be specifically accounted for. As a result, this study assumed similar isoform diversity at the cell and bulk levels, which, although likely to verge on overestimation, sets a theoretical maximum for single cell transcriptome complexity. With this in mind, we set out to investigate how the two effects mentioned above hindered the detection of isoform diversity.

3.5.1 Methods

Data and software availability

The simulations presented in this chapter made use of a mouse neural transcriptomics dataset generated in our laboratory and subsequently published by Tardaguila, de la Fuente et al. [50]. This dataset was generated from mouse primary neural stem cells (NSCs) and NSC-derived oligodendrocytes and comprised ~0,6 million PacBio and ~60 million Illumina reads per replicate, with 2 replicates being generated per sample. Raw sequencing data used for this purpose is available at the Sequencing Read Archive (SRA), under study accession SRP101446.

Briefly, long reads were first used for transcript model definition, which was achieved using the ToFU pipeline [120]. Next, short, Illumina reads were mapped to the mouse reference genome (GRCm38, mm10) using the splice-aware STAR aligner [80]. Isoform expression estimates (transcripts per million, TPM) were subsequently obtained using RSEM [85]. To ensure the reliability of the long read-defined transcriptome, the SQANTI software [50] was used to perform quality control and filter false-positive isoforms. The resulting transcriptome contained 11511 mouse transcripts (after filtering by transcript expression > 1 TPM) belonging to 6956 genes, 2509 of them multi-isoform genes. This set of transcripts, together with the associated isoform expression estimates, constituted the template data in our simulation strategy.

The scripts used to run the simulations, together with the necessary data and documentation, are available at https://github.com/aarzalluz/singlecell-isoform-simulation. The original transcriptome files, i.e. transcript FASTA file and transcriptome annotation in GTF file format, are available under the `transcriptome` folder in the repository.

Short-read data simulation

To assess the effect of SMART vs UMI-based library preparation methods in isoform detection, we performed a controlled short-read simulation in which several degrees of 3'/5' coverage bias were recreated. To achieve this, transcript sequences from the long read-defined transcriptome (see section 3.5.1) were trimmed starting from the 3' and 5' ends to generate sequences of lengths 100, 200, 300, 500 and 1000 nucleotides or base-pairs (bp). Trimmed transcriptomes were then used as template for short-read sequencing data simulation with the `simulate_experiment()` function in the polyester R package [119]. This strategy was designed to capture the properties of UMI-based sequencing while an additional, full-length simulation was performed using the untrimmed transcript sequences, recreating Smart-seq data.

The TPM expression values computed by Tardaguila et al. using bulk Illumina data for NSC and oligodendrocyte samples [50] (see section 3.5.1) were used to set the transcript expression levels to be simulated via the `reads_per_transcript` parameter in `simulate_experiment()`. As a result, 2 samples and a total of 1 million reads/sample were simulated, with read count values distributed across transcripts according to their TPM expression. For UMI simulations, the `simulate_experiment()` function was run with the following additional non-default parameters: `num_reps = 1`, `paired = FALSE`. Briefly, `num_reps` establishes the number of replicates per sample, while `paired` controls whether to simulate single or paired-end short-read data. In this case, given the short lengths of the trimmed transcripts, single-end libraries were generated to avoid overestimating coverage for the sequenced sections. Read lengths were gradually increased to continue to ensure evenness of coverage, namely by setting `readlen = 25` for 100 and 200bp fragments, `readlen = 50` for 300 and 500bp and `readlen = 100` for the 1000bp-fragmented transcriptome. For Smart-seq simulations, however, `paired = TRUE` and `readlen = 250` were set in order to accommodate to the lengths of real transcripts.

Simulated short reads were mapped to the mouse reference genome (GRCm38, mm10) using STAR [80] and the mouse neural transcriptome by Tardaguila et al. [50] as the reference annotation. The expression values for isoforms in each of the simulated scenarios were next computed using RSEM [85].

Evaluation of isoform detection in UMI-based and Smart-seq sequencing data

Using RSEM results, we next evaluated isoform detection under UMI and Smart-seq simulation scenarios. To achieve this, we computed the percentage of isoforms per gene that were detected using simulated data (> 0 TPM) relative to the number of isoforms per gene in the long read-transcriptome by Tardaguila et al. [50], which was used as the reference annotation for read simulation. As a result, genes in the simulations that presented the same amount of expressed isoforms as in the long-read annotation were considered to be *fully resolved*. Meanwhile, genes where only some of the isoforms were detected as expressed were labeled as *partially resolved*. In order to asses partial resolution, genes were binned into four intervals according to their resolution percentage: (0, 25], (25, 50] (50, 75] and (75,100]. The percentage of genes falling into each bin was then computed. Of note, single-isoform genes were removed previous to computing these metrics.

Long-read count simulation

To illustrate the limitations in sequencing depth per cell imposed by long-read sequencing technologies, we simulated the effect of sequencing an increasingly large pool of single cells on a MinION or SMRT flow-cell, in which the total number of reads per sequencing run (i.e. the sequencing throughput) was fixed. For this purpose, the NSC and oligodendrocyte TPM expression values in the study by Tardaguila et al. [50] were used to define the initial number of long reads per transcript, thus establishing a maximum sequencing throughput of 1 million long-reads for the simulated flow-cell. This baseline scenario is equivalent to a situation in which a sequencing library corresponding to a single cell is run on said flow-cell, yielding 1 million reads per cell and reaching the theoretical maximum of transcript detection for our simulation.

Next, subsequent expression reduction steps were performed to replicate situations where 2, 4, 8, 16, 32 and 64 cells were pooled and sequenced in the same SMRT cell. In order to achieve this, the TPM expression values in the original isoform expression matrix were divided by the total number of cells in the simulation. In this process, transcripts yielding expression < 1 TPM after each expression reduction step were considered undetected and a TPM value of 0 was assigned in the expression matrix.

Evaluation of isoform detection using long-read simulated data

The trade-off between cell and read throughput imposed by long-read technologies was evaluated using expression-related indicators of isoform detection. First, the number of genes for which more than one isoform was detected was computed to assess the loss of alternative splicing complexity as throughput constraints became more stringent. Then, we used transcript expression results in NSC and oligodendrocyte to retrieve the number of genes with major isoform switches between samples in each of the simulations. Based on the definition by de la Fuente et al. [121], where the metric was introduced as part of the tappAS software, we considered a major isoform switch to have occurred if the most highly expressed isoform from a gene changed between conditions or samples. In each long read sequencing scenario, only genes genes that remained multi-isoform in both samples were considered. For evaluation purposes, the total number of switches preserved in each simulation was compared to that found in the 1-cell per SMRT cell scenario, which in this simulation was equivalent to the number of switches found in the real dataset. This meant that a 2-cell simulation would correspond to 1 oligodendrocyte and 1 NSC cell sequenced together, at a maximum throughput of ∼0.5 million reads/cell, while the same logic was applied for subsequent increases in multiplexing, i.e. 4, 8, 16, 32 and 64 cells.

3.5.2 Results

Short-read sequencing: UMI vs SMART-based approaches

To emulate the effect of full and non-full length transcript coverage on isoform detection, the polyester R package [119] was used to simulate reads from a growing length of the 3' and 5' ends of the mouse neural PacBio transcripts, i.e. 100, 200, 300, 500 and 1000bp fragments (detailed workflow in figure 3.3.a and 3.3.b and section 3.5.1). This phenomenon, as described in section 3.2.1, is intrinsic to UMI-based library preparation methods. Notably, even though the present simulation was not designed to capture a real-life scenario -in which covered lengths would be expected to vary from transcript to transcript and reads arising from PCR duplicates would be collapsed-, it was deemed sufficient to illustrate the coverage bias in extant in UMI-based methods (figure 3.4.a). In turn, the longer the covered fragments, the larger the number of AS events to be captured, which constituted a suitable scenario to evaluate isoform detection limitations in UMI protocols. To complement this, a set of short-reads spanning the entire transcript sequences was simulated to recreate a SMART-based library preparation strategy (figure 3.3.b). Differences across simulations were evaluated by computing the multi-isoform gene resolution percentage, which measures the relative amount of isoforms per gene that were detected as expressed in comparison to the reference long-read transcriptome (as described in section 3.5.1). Genes where 100% of isoforms were quantified (TPM>0) were thus considered to be fully resolved, whereas multi-isoform genes are considered to be partially resolved if they show a percentage of quantified isoforms (i.e. resolution percentage) ranging from 0 to <100%.

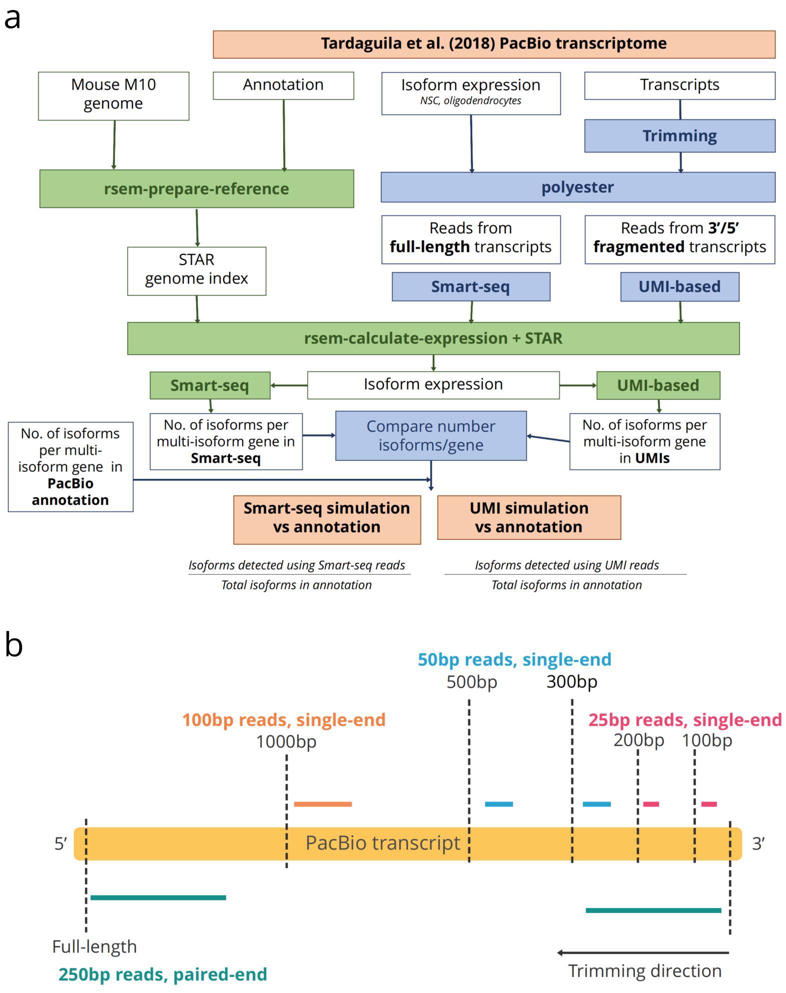

a
Tardaguila et al. (2018) PacBio transcriptome
Mouse M10 genome
Annotation
Isoform expression
NSC, oligodendrocytes
Transcripts
Trimming
rsem-prepare-reference
polyester
Reads from full-length transcripts
Reads from 3'/5' fragmented transcripts
STAR genome index
Smart-seq
UMI-based
rsem-calculate-expression + STAR
Smart-seq
Isoform expression
UMI-based
No. of isoforms per multi-isoform gene in PacBio annotation
No. of isoforms per multi-isoform gene in Smart-seq
Compare number isoforms/gene
No. of isoforms per multi-isoform gene in UMIs
Smart-seq simulation vs annotation
UMI simulation vs annotation
Isoforms detected using Smart-seq reads / Total isoforms in annotation
Isoforms detected using UMI reads / Total isoforms in annotation
b
50bp reads, single-end
500bp
300bp
100bp reads, single-end
1000bp
25bp reads, single-end
200bp
100bp
5'
PacBio transcript
3'
Full-length
250bp reads, paired-end
Trimming direction

Figure 3.3: Short and long-read simulation strategy. **a** Short-read simulation workflow. Transcript sequences from the mouse neural transcriptome were trimmed and reads simulated from fragments and full-length transcripts using polyester. Isoform expression was quantified using RSEM to calculate multi-isoform gene resolution percentages. **b** Read lengths (represented for 3' UMIs) and library types (single/paired-end) used in each short-read simulation.

In this context, while full resolution was achieved for up to 23.2% out of 2509 multi-isoform genes when reads were simulated from the 3' end, reads from 5' transcript fragments yielded full resolution of a maximum of $\sim$40% genes (1000bp UMI simulations, figure 3.4.b). Full-length reads, however, outperformed all simulated UMI scenarios with 52.1% fully-resolved multi-isoform genes, even when considering 5' fragments (figure 3.4.b). When analyzing partial as well as full resolution (figure 3.4.c), we observed that between 25% and 50% of expressed isoforms were discriminated for the majority of multi-isoform genes ($\sim$75%) when using 3' UMI sequencing. Of note, only subtle changes in the highly-resolved fraction (genes with $>$75% resolved isoforms) were observed when increasing 3' transcript fragment lengths (figure 3.4.c, upper panel), while 5' fragments of 500 and 1000bp slightly approximated the isoform resolution distribution achieved by full-length reads (figure 3.4.c, lower panel). Therefore, we speculated that, for most alternatively spliced genes, more than half of events decisive to discriminate same-gene isoforms tend to occur far from the 3' end and towards the TSS. This would require further validation, as long read datasets may include additional 5' end variability corresponding to RNA degradation products, which are difficult to distinguish from true, alternative TSS. The most likely explanation, however, is that our simulation is capturing a combination of both low accuracy when defining the 5' end of long read isoforms and an increase in alternative splicing nearer the TSS. Similar results were observed when using oligodendrocyte expression for simulation (data not shown).

In spite of the more favorable distribution of resolution percentages obtained when using Smart-seq reads, high resolution percentages were achieved for only ∼50% of multi-isoform genes, a proportion that was lower than initially expected. However, another aspect of multi-isoform gene resolution that ought to be considered was the absolute number of isoforms detected per gene. As an example, even though a gene with 2 isoforms in the reference annotation and 1 detected with simulated data may have 50% of its isoforms resolved, it would not contribute to gain insight on alternative splicing, owing to the fact that differential isoform usage analysis requires multiple isoforms to be detected per gene. Hence, we also reported the percentage of multi-isoform genes for which more than one isoform was detected (figure 3.4.d). This revealed that 75% of genes with multi-isoform resolution were obtained when using Smart-seq reads, while 3' and 5' UMI reads yielded up to 44 and 65%, respectively (1000bp fragments, figure 3.4.d). These results suggested that usage of the Smart-seq approach resulted in higher overall isoform diversity than UMI simulated reads, even if some of the multiple-isoform genes in the reference dataset were not fully resolved in quantification.

Long-read sequencing: exploring trade-offs between sequencing depth and cell throughput

With the purpose of understanding the implications of limited depth for long-read isoform detection, we simulated a scenario in which an increasing number of cells were multiplexed in one Pacbio Sequel run yielding 1 million full-length reads. In such a situation, where maximum depth is fixed, the number of reads per cell decreases as the number of cells per run grows larger.

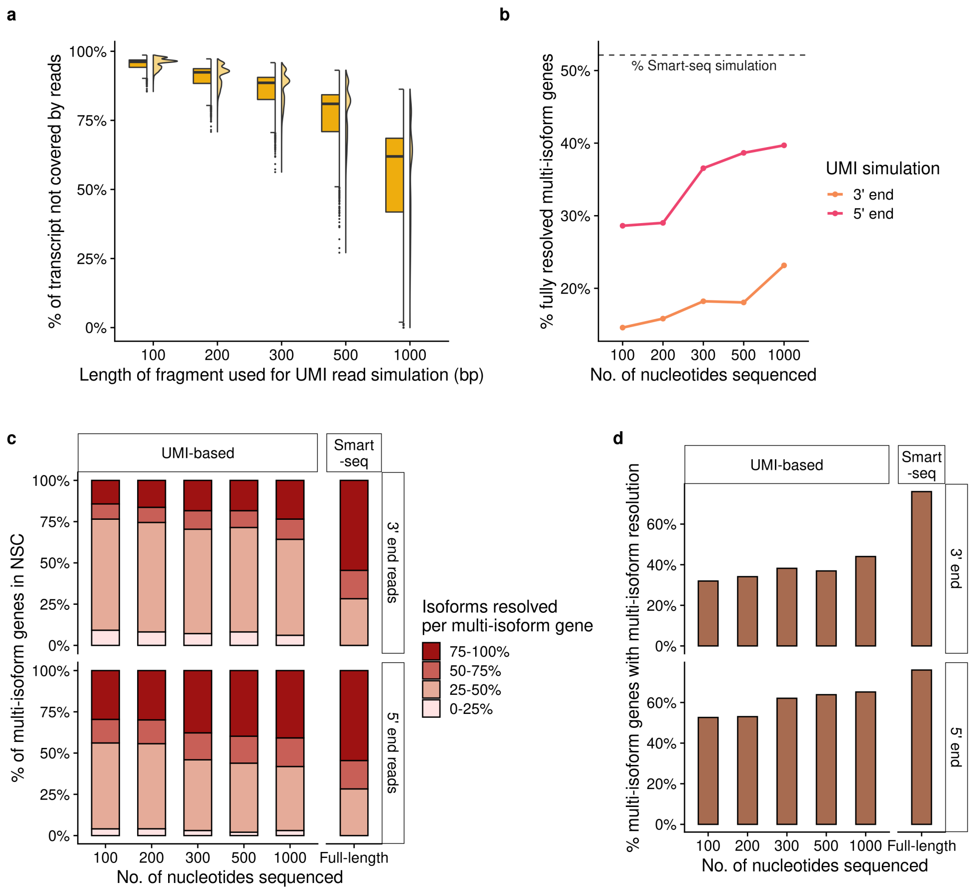


Figure 3.4: Short-read simulation results. **a** Percentage of the transcript sequence not covered by simulated reads (y axis) for each of the fragment lengths used in the UMI-based library preparation simulation. **b** Percentage of multi-isoform genes in the reference annotation (Tardaguila et al.) that are fully resolved in the NSC UMI simulations simulations (resolution percentage = 100%). The dashed line corresponds to fully resolved percentage achieved using Smart-seq simulated reads. **c** Resolution percentage results for NSCs. For each multi-isoform gene, partial and full resolution percentages in UMI (3' and 5') and Smart-seq simulations are binned into four intervals. The percentage of multi-isoform genes for which 0–25, 25–50, 50–75 and 75–100% of their isoforms have been resolved is represented in the y-axis, while the fragment lengths can be seen in the x-axis. Note that Smart-seq results have been plotted twice, in both the 3 end and 5 end rows, to ease visual interpretation. **d** Percentage of multi-isoform genes for which only one isoform was quantified, for both 3' and 5' UMI and Smart-seq simulations.

Then, the number of genes for which more than one isoform was detected, as well as the number of isoform switches that were observed between NSC and oligodendrocytes, were calculated for each each simulation (see section 3.5.1).

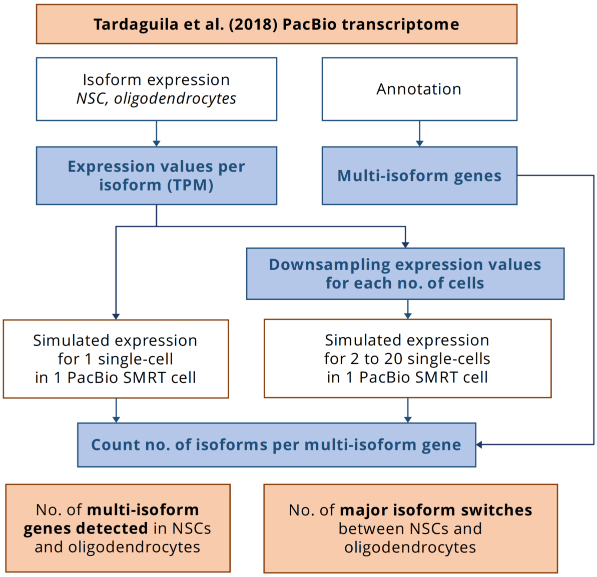

Tardaguila et al. (2018) PacBio transcriptome
Isoform expression
NSC, oligodendrocytes
Annotation
Expression values per isoform (TPM)
Multi-isoform genes
Downsampling expression values for each no. of cells
Simulated expression for 1 single-cell in 1 PacBio SMRT cell
Simulated expression for 2 to 20 single-cells in 1 PacBio SMRT cell
Count no. of isoforms per multi-isoform gene
No. of multi-isoform genes detected in NSCs and oligodendrocytes
No. of major isoform switches between NSCs and oligodendrocytes

Figure 3.5: Long-read simulation workflow. Short read-based TPM isoform expression from Tardaguila et al. was used to recreate a Sequel run of one million long reads where a single cell is sequenced. Values were downsampled to simulate scenarios where an increasing number of cells are pooled. The number of multi-isoform genes and isoform switches in the original study was then compared with the number detected in the simulated scenarios.

To simulate this, bulk transcript expression values obtained in NSCs and oligodendrocytes [50] were downsampled, assuming equal distribution of the reads among cells (see section 3.5.1). Using bulk data allowed us to work with a theoretical maximum of transcript detection. Presumably, however, the dropouts in a real single-cell scenario would play an important role, hence all results discussed below should be interpreted as upper-bound estimates. Long-read isoform expression results were generated for 2, 6, 10, 16 and 20 cells using the simulation workflow detailed in figure 3.5.

The simulation revealed that the number of genes for which more than one isoform was detected decreased with sequencing depth (figure 3.6.a). As expected, the downsampling strategy (designed to mimic shallow sequencing), resulted in the loss of lowly expressed transcripts, some of which constituted alternative isoforms. However, we hypothesized that single molecule technologies may still be able to capture differences in isoform expression when they implied drastic changes in expression, i.e. major isoform switches. To evaluate this, the number of isoform switches detected between NSC and oligodendrocyte was computed for each simulated multiplexing scenario (figure 3.6.b). Of note, only 337 (∼14%) out of 2509 multi-isoform genes in the original study [50] showed a major isoform switch, supporting the notion that most isoform changes constituted changes in isoform expression ratios. Reassuringly, most of these switches (305) were detected in the best-case scenario of our simulation (2 cells total, i.e. one per condition), although the number decreased with the no. of reads per cell. To illustrate, in a 20-cell experiment, only ∼30% of major isoform switches would have been detected, according to our simulation (figure 3.6.b). Therefore, this simulation suggested that favoring cell number over sequencing depth could greatly affect sensitivity, which would result in multiple isoforms being detected for only a reduced number of genes. Ultimately, this would hinder the characterization of alternative isoform usage changes, especially in those cases where no major isoform switches are produced, or where switching isoforms are not among the top most highly expressed.

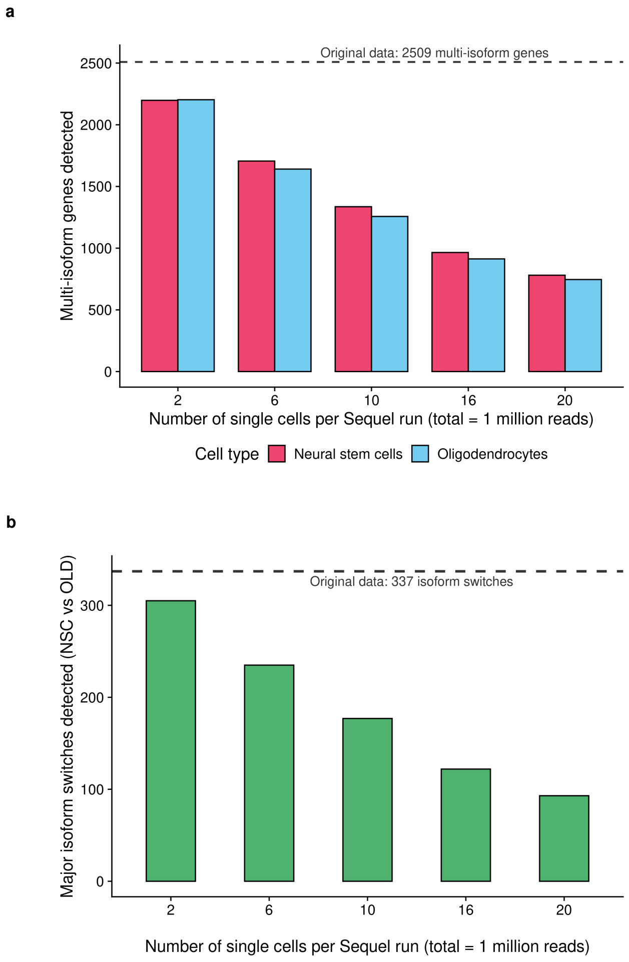


Figure 3.6: Long-read simulation results. a Number of multi-isoform genes detected by simulated sequencing depth per cell. The dashed line indicates the number of multi-isoform genes in the original neural long-read transcriptome. **b** Nnumber of isoform switches detected between NSCs and oligodendrocytes, assuming half of the cells belong to each cell type (i.e. two cells correspond to one oligodendrocyte and one NSC cell). The dashed line indicates the number of isoform switches detected in the original study.

All in all, the evaluation of the methodologies available at the time and the results of these computational simulations prompted us to design a hybrid strategy to study single-cell isoform expression. Briefly, we devised the generation of a bulk long-read transcriptome, and the usage of full-length, short-read scRNA-seq data, which was the best performing technology according to our study, for cell-level isoform quantification. Regarding quantification, RSEM, although not scRNA-seq specific, was deemed best for transcript-level expression estimation due to the timely incorporation of a single-cell sparsity prior to its EM algorithm. This pipeline, together with the results of its application, will be thoroughly described in Chapter 4.

3.6 Discussion

The study of isoform expression has presented the single-cell field with considerable challenges ever since the publication of the first proof-of-concept studies. In the early days of scRNA-seq, the lack of dedicated computational methods, together with the poor understanding of the unique properties of the data, hindered progress in isoform analyses. As a result, most studies published at the time focused solely on single splicing events [98, 101, 102, 104–106] and refrained from attempting full-length transcript quantification. The work that is included in this chapter, published in 2018 [37], helped bridge this gap in three different ways. First, by providing a set of guidelines for the generation of isoform-compatible scRNA-seq data. Next, the available literature was thoroughly reviewed, describing the pitfalls and limitations commonly faced in the field. Finally, a simulation-based, quantitative study supplied the first evaluation of the feasibility of scRNA-seq-based isoform expression analyses. In spite of having successfully accomplished these goals -which informed the design of our own pipeline for single-cell isoform quantification (see Chapter 4)- time has allowed researchers to find innovative ways to address the limitations that were originally outlined in our study. Therefore, to bring the framework of this chapter up to date, we will now discuss the latest advances in the single-cell isoform field, focusing on whether they have served to overcome its most pressing issues.

3.6.1 Advances in short-read data generation

Regarding short-read scRNA-seq, our study outlined the advantages of using full-length over UMI-based approaches for isoform studies. In spite of their advantages for certain applications, short-read protocols such as Smart-seq2 have largely been replaced by high-throughput, UMI-based protocols in the last few years, mainly by the Chromium Single Cell Gene Expression solution commercialized by 10X Genomics (https://www.10xgenomics.com/products/single-cell-gene-expression, accessed March 2023). Nevertheless, SMART-based library preparation has simultaneously become more scalable and cost-effective than reported in our 2018 study, thanks in part to protocol optimizations carried out with the resources of large research institutions [122, 123] and consortia [124–127]. These improvements to library preparation are mainly due to protocol automation [128] and reagent volume reduction (i.e. miniaturization) [129]. As a result, considerably larger full-length scRNA-seq datasets have been released over the last few years, with throughput often bordering on 10K cells. These have been collected and documented in the framework of a number of single-cell consortia, which have empowered users by easing access to data and metadata [130]. All in all, the increased cell throughput and availability of public full-length datasets has unlocked previously unfathomable opportunities for single-cell isoform research.

Also concerning short-read data, the recently released Smart-seq3 library preparation method [131] constitutes a groundbreaking opportunity to end the hereby discussed incompatibility between full-length transcript sequencing and UMIs. The protocol combines template switching for full-length transcript sequencing with 5' UMIs and additional *in silico* reconstruction of mRNA molecules. Using paired-end sequencing, information can be pooled across same-UMI fragments to partially resolve the structure of the transcript, allowing allele and isoform-specific read assignment. With significantly higher sensitivity, throughput and more competitive costs than its predecessor, Smart-seq3 may constitute a solid choice for future single-cell isoform studies. This strategy, however, is still far from achieving long read-level resolution, with only $\sim$40% of reads being unambiguously assigned to an isoform and transcript reconstruction proving difficult beyond 1.5kb [131]. Authors additionally reported that, on average, Smart-seq3 reconstructed molecules only covered 46% of the transcript sequences obtained using PacBio. Therefore, while Smart-seq3 constitutes a potentially game-changing technology, some of the constraints outlined in our study of scRNA-seq isoform study limitations regarding short-read isoform resolution still hold true after five years of single-cell technological advances.

Additionally, later simulation-based studies have extended the theoretical limit assessment in this chapter, further evaluating the ability to confidently resolve isoforms using short-read scRNA-seq data. Specifically, a study by Westoby et al. [132] systematically reported a tendency to underestimate the number of detected isoforms per gene in each cell. While these conclusions align with our own simulation results, the severity of the dropout problem for isoform detection was further illustrated in this study, in which the simulation framework specifically accounted for the single cell-specific data structure and biases. As a result, authors were able to point out that a mechanistic understanding of cell-level isoform choice could lead to better modelling of dropout probabilities and more accurate isoform detection [132]. At the experimental level, and also in line with our study, Westoby et al. pointed at the necessity to increase sequencing depth to alleviate the burden of dropouts, while signalling the capture efficiency of the protocols as the most limiting factor. This highlights the relevance of the simulations in this chapter, while suggesting that future studies will build on the work that ourselves and others have performed to achieve a sound understand of cell-level splicing and the potential of scRNA-seq data to unravel its enigmas.

3.6.2 Advances in computational method development

Promising insight has been obtained regarding the suitability of bulk-designed transcript quantification tools for single-cell research. Shortly after the publication of the work presented in this thesis [37], the first benchmarking of scRNA-seq isoform quantification was published by Westoby et al. [133]. By designing an innovative strategy for transcript-level scRNA-seq data simulation, the study reported that the performance a number of widely-used tools, including RSEM [85] and Kallisto [87], was near-optimal in scRNA-seq data, both regarding isoform detection and expression estimation accuracy. Reassuringly, there were only marginal improvements in performance when the benchmark was repeated in bulk data, and only a slight tendency to call false positive isoforms in scRNA-seq data (i.e. isoforms for which reads were not simulated) seemed concerning to the authors [133]. These results supported our decision to incorporate RSEM into our single-cell isoform analysis pipeline (see Chapters 4 and 5), whereas other authors have used Kallisto for scRNA-seq transcript quantification [77].

Over the years, however, the complexity of isoform-level transcriptomics using short-read data alone has prompted authors to persist in improving the quantitative analysis of single-cell splicing events. Some method publications following the release of BRIE [104] and Expedition [102] focused on improving the sensitivity and detection limits of AS analysis. This included strategies to mitigate the sparsity of scRNA-seq data by aggregating reads across groups of exons (SCATS [134]) and tools that increased the accuracy of splice junction detection in single cells (SICILIAN [135]). The VALERIE software [136], on the other hand, tackled computational efficiency issues by enabling comparisons of PSI value changes across multiple groups of cells. Other methods, namely ISOP [137] and MARVEL [138], further explored the concept behind Expedition by extending and improving the classification of AS events into splicing modalities. Moreover, some of these tools have enabled the usage of UMI data for AS analysis [134, 135, 138–140] which, although less optimal, may be of interest considering that UMI-based methods are the leading single-cell data generation strategy. These methods have contributed to fill the reported gap in computational method development (section 3.4) for AS event analysis, however, similar innovations have rarely been made regarding isoform-level strategies, with few exceptions [141, 142]. Among them, the DTUrtle pipeline [141] was the first to leverage transcript-level single-cell counts to detect differential isoform usage. Although this work unlocked isoform-level analysis in scRNA-seq data, it was based on bulk-designed methods DRIMSeq [143] and stageR [144], with little to no single cell-specific method development other than data wrangling adaptations. Later, Gilis et al. released satuRn [142], which introduced a novel statistical framework for modelling transcript-level counts and test for changes in isoform usage. While authors reported that satuRn outperformed a number of bulk methods regarding both scalability and performance, a more comprehensive benchmark is still required to compare available approaches and set gold standards for differential isoform usage analysis in single-cell data.

3.6.3 Advances in long-read data generation

The single-cell field has also been witness to the inception of several long-read library preparation methods that have progressively addressed known limitations, among which low sequencing accuracy constituted a main drawback, particularly for ONT data. ScISOr-Seq [35] was the first method to provide simultaneous generation of short and long-read scRNA-Seq data, namely by splitting cDNA -including UMIs and cell barcodes- into two separate pools. This implied parallel demultiplexing of both sets of reads, a task that is hindered by sequencing errors, as was reported in section 3.2.4.

Although agnostic to the long-read method, ScISOr-Seq yielded better results when coupled with PacBio IsoSeq, where the higher accuracy of the technology's multi-pass system resulted in a much larger number of reads being successfully demultiplexed and assigned to individual cells, i.e. ∼58% PaccBio CCS vs ∼33% 2D ONT reads. To alleviate the data loss caused by ONT sequencing errors, Volden et al. developed the R2C2 strategy, in which the same principle as in circular consensus sequencing was applied to increase the accuracy of ONT [145]. R2C2 achieved 94% base accuracy, in comparison to 87% and 95% yielded by 1D and 1D^2 reads, respectively. As a result, ∼74% of R2C2 generated reads were unambiguously assigned to cells, which constituted a major improvement over previous results [35]. Perhaps even more interestingly, these numbers rivaled with the properties of the PacBio IsoSeq data generated in the same study, i.e. 99% accuracy [145]. The scCOLOR-seq protocol was similarly born out of the need to improve long-read single-cell assignment, which was tackled using dimeric nucleotide building blocks in UMI and cell tag sequences to allow intuitive error correction [146]. Besides these innovations in library preparation, a number of computational tools have been designed to overcome barcode resolution challenges caused by ONT sequencing errors [55–57]. In addition, efforts have been made to characterize and mitigate errors occurring across the transcript sequence [147, 148], which generate artifacts and often result in incorrect isoform detection. Strategies to improve Nanopore read accuracy are particularly relevant given the larger sequencing yield provided by ONT sequencers, as they have vast potential to increase the amount of usable data for isoform transcriptomics applications.

In spite of the read quality improvements presented by novel long-read sequencing strategies, these methods were still affected by the cell throughput vs sequencing depth per cell trade-off that we reported in our theoretical study (section 3.2.3). For instance, although Volden et al. achieved a large increase in sequencing depth with R2C2 (∼7,600 reads/cell at an average detection rate of 532 genes/cell), data generation was achieved for as little as 96 cells [145]. The ScISOr-Seq study, on the other hand, reported large cell throughput (∼7,000 single-cells) with low depth (270 median reads/-cell), leading to the detection of only 129 median genes/cell. In spite of this, follow-up studies by the same authors have reported remarkable advances to mitigate these limitations, with up to 1,000 reads/cell for 7,000 cells in Joglekar et al. using ScISOr-Seq [78] and 4,000 reads/cell for 3,000 cells in Volden et al. with an improved R2C2 chemistry [149], whereas other laboratories have successfully developed methodologies that yielded similar (i.e. LR-Split-Seq, ∼1,000 cells and ∼500 reads/cell [150]) or even superior results (i.e. ScNaUmi-seq, ∼1,000 cells and 6,047 genes/cell [55]). Even though these numbers are still far from those obtained using SMART-based methods, they have undoubtedly broken new ground for the single-cell transcriptomics field (section 3.3).

Moreover, the development of novel library preparation strategies has been boosted by the decrease in sequencing cost and the release of enhanced long-read sequencers, such as PacBio Sequel II and ONT PromethION, with studies involving the latter achieving particularly promising results [55]. All in all, this has begun to unlock some computational applications for long-read data, such as cell clustering [149], cell type-specific isoform identification [78, 149, 151], differential isoform usage detection [78, 151] or spatial transcriptomics [78, 152]. Furthermore, most aforementioned methods allow the generation of long-read and UMI-tagged Illumina data from the same cells [55, 78, 149, 150]. Given the complementary properties of both data types, this has contributed to enhance computational analysis, for instance, to achieve higher cell clustering accuracy [55, 78, 153].

Long-read scRNA-seq throughput constraints, however, are not only explained by instrumental limitations and sequencing errors in barcodes leading to unusable data. Authors have recently pointed out that, when using the PacBio CCS system, the shorter insert length of transcript cDNA molecules leads to a waste of sequencing throughput. Specifically, whereas long DNA inserts (optimally 10-20 kb) are sequenced 10-15 times, which is enough to achieve excellent read quality, transcript-derived inserts ($\sim$1.5 kb on average for human) are often sequenced up to 50-60 times [154, 155]. In this context, several protocols aiming to optimize data generation through the PacBio platform have recently been released: HIT-scISOseq [155], SnISOr-Seq [153] and MAS-IsoSeq [154], which are based on similar principles. First, template-switching oligonucleotide (TSO) artifacts are removed, preventing the generation of barcode-free reads that generate a waste of sequencing potential. Next, multiple cDNA molecules are concatenated to create long-insert SMRTbell structures for sequencing. As a result, authors have reported x15 [154] and x8 [153, 155] increases in sequencing throughput. The progressive implantation of these protocols, particularly the adaptation of the

MAS-IsoSeq method by PacBio (MAS-Seq for 10x Single Cell 3' kit, product no. 102-659-600, https://www.pacb.com/products-and-services/applications/rna-sequencing/single-cell-rna-sequencing/, accessed May 2023), have the potential to boost single-cell isoform transcriptomics to unprecedented levels.

Nevertheless, the different study results summarized above raise a number of pressing concerns. First, each of the above-mentioned studies seem to report the effectiveness of their protocols using different metrics -for instance, the number of reads and UMIs per cell is not always reported and, when supplied, it often corresponds to different stages in data pre-processing. We therefore believe that a systematic benchmark of long-read scRNA-seq sequencing methods, including an evaluation of throughput, accuracy and sensitivity under the same conditions and biological samples, is essential to adequately measure and compare the benefits and pitfalls of each protocol. Furthermore, in spite of the progress that has been made in data generation, there is still a lack of dedicated computational methods for the analysis of scRNA-seq long-read data. For instance, single cell-adapted isoform discovery pipelines, such as FLAMES [151], C3POa/Mandalorion [149], IsoQuant [156] and IsoSeq3 (https://isoseq.how/umi/, accessed May 2023), are based on pre-existing bulk approaches. Regarding long-read isoform expression analysis, only a couple of resources have been released so far [78, 151]. Among them, scisorseqr [78] constitutes the only method including a comprehensive gene-level test for differential isoform usage, since FLAMES requires the selection of the two most highly expressed isoforms prior to testing [151], which only partially accounts for the diversity of the alternative isoform landscape. Moreover, the large sample size and broad cell type diversity that is typically included in single-cell datasets could foster the development of innovative strategies for downstream analysis, fully unlocking the potential of long-read scRNA-seq for the detection of isoform usage patterns across cell types.

All in all, it can be concluded that most of the limitations that we depicted in the present chapter still operate: from sequencing depth and throughput limitations to the incompleteness in short-read transcript coverage and challenges in long-read accuracy, the single-cell isoform transcriptomics field has a challenging road ahead. Nevertheless, as it has been shown throughout this discussion, researchers have made remarkable progress in all of these aspects, rapidly exceeding expectations regarding what was attainable when the present study was conducted. Thanks to novel protocols and tools, this trend can only be expected to continue, gradually bringing single-cell transcriptomics closer to achieve complete and accurate isoform resolution.

Chapter 4

Integration of bulk long-reads and single-cell short-read data to enhance isoform analyses

Chapter partially adapted from the following publications:

Arzalluz-Luque, Á., Salguero, P., Tarazona, S., Conesa, A. *acorde* unravels functionally interpretable networks of isoform co-usage from single cell data. *Nature Communications* **13**, 1828 (2022).

Pardo-Palacios, F., Arzalluz-Luque, Á., et al. SQANTI3: curation of long-read transcriptomes for accurate identification of known and novel isoforms. *Accepted for publication in Nature Methods*.

4.1 Introduction

Traditionally, RNA-seq studies have used publicly available reference transcriptomes such as RefSeq and ENSEMBL for short read isoform quantification. However, most tissues and cell types will express only a subset of the genes and isoforms contained in the reference, including sample-specific isoforms that may not be present in reference annotation [52]. In addition, previous studies have shown that isoform detection accuracy increases when adopting a reduced reference catalog, i.e. using only tissue-specific isoform sets as a reference for mapping [157]. As opposed to short-read data, which requires the application of complex transcriptome assembly methods [79], long-read RNA-seq (lrRNA-seq) technologies have the potential to achieve both of these goals. Currently, PacBio and Oxford Nanopore Technologies (ONT) are the leading sequencing methods in the lrRNA-seq field. Although based on different underlying transcript capture and sequence detection mechanisms (described in Chapter 1 and thoroughly reviewed in [45]), both PacBio and ONT are able to generate one sequencing read per cDNA molecule in the library, spanning the entire length of transcripts and reducing the complexity of *de novo* transcriptome reconstruction [158].

Even though long read-based transcriptome generation pipelines are largely dependant on the tool of choice, most of them rely on a series of common data processing steps [158]. First, long reads are generally assembled, either by mapping to a reference genome (e.g. [156, 159, 160]) or *de novo*, i.e. by clustering similar reads (e.g. PacBio's IsoSeq). This process can be accompanied by previous read error correction (e.g. [160]) and downstream assembly correction, which can be reference (e.g. [156]) or data-based (e.g. IsoSeq3 polishing). Additionally, isoform models sharing a large part of their sequence are frequently collapsed to eliminate redundant models from the transcriptome [160, 161]. This transcriptome reconstruction process, however, poses complex challenges regarding sequencing errors and library preparation or degradation artifacts that may be mistaken for novel isoforms [50]. As a result, a detailed quality control and filtering process is often required before the assembled transcriptome can be used for downstream analysis.

SQANTI [50] was the first toolkit for the characterization and quality control of long-read transcriptomes, and its most recent version, SQANTI3 [162], remains one of the most widely used software tools in the lrRNA-seq field. The approach is based on the incorporation of several additional data types, including, but not limited to, short-reads, CAGE and polyadenylation (polyA) sites, which are used to compute a large number of quality attributes related to splice-junction (SJ), transcription start site (TSS) and transcription termination site (TTS) support. In addition, isoforms are characterized using the SQANTI classification scheme (figure 4.1), which divides them into categories and subcategories depending on their degree of novelty or their similarity to reference transcriptome isoforms. As a result, transcript isoforms can be thoroughly evaluated and subsequently filtered to enhance the quality of the long-read transcriptome.

In spite of its potential for isoform studies, single-cell lrRNA-seq data is limited in regarding sensitivity and transcriptome coverage due to its inability to produce deeply-sequenced datasets at an effective cost [37]. As a result,

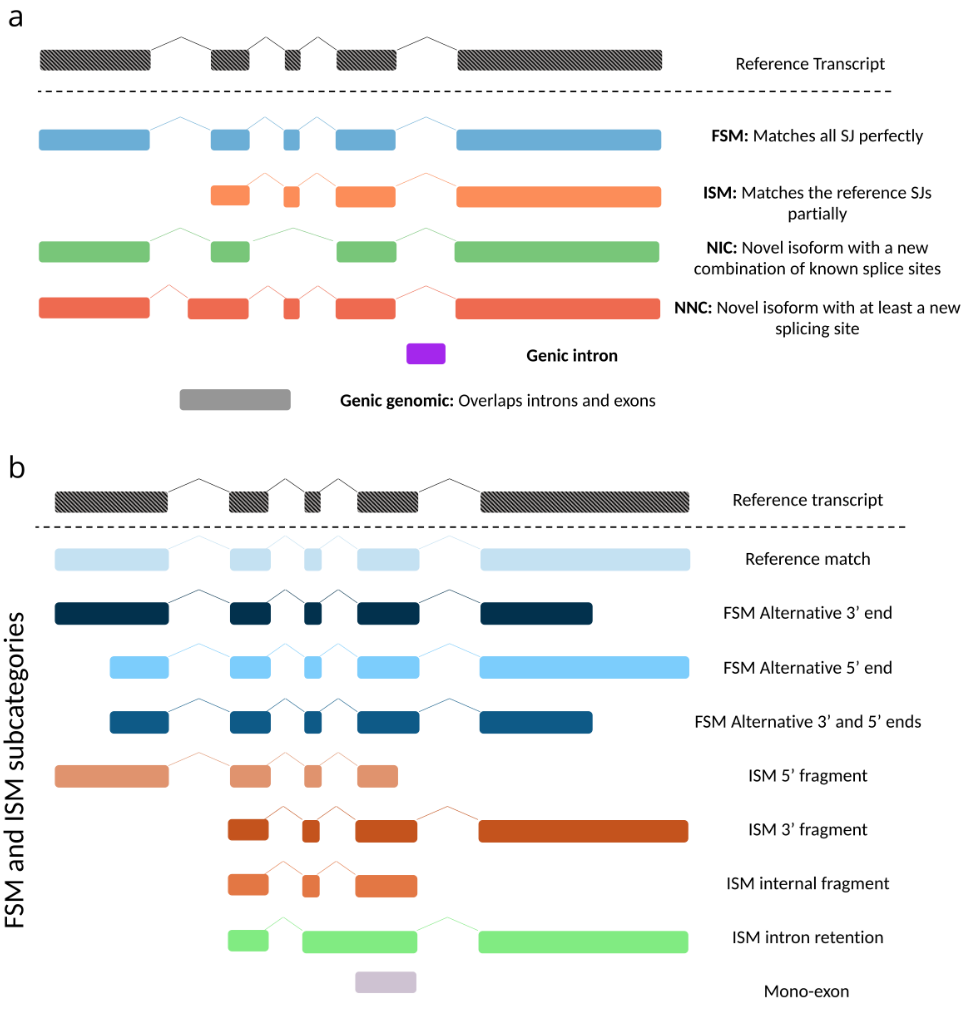

a
Reference Transcript
FSM: Matches all SJ perfectly
ISM: Matches the reference SJs partially
NIC: Novel isoform with a new combination of known splice sites
NNC: Novel isoform with at least a new splicing site
Genic intron
Genic genomic: Overlaps introns and exons
b
Reference transcript
Reference match
FSM Alternative 3' end
FSM Alternative 5' end
FSM Alternative 3' and 5' ends
ISM 5' fragment
ISM 3' fragment
ISM internal fragment
ISM intron retention
Mono-exon
FSM and ISM subcategories

Figure 4.1: SQANTI3 transcript classification. **a** Main structural categories for transcript models belonging to known genes. Categories are defined based on the completeness and novelty status of the string of detected splice junctions. FSM: full-splice match; ISM: incomplete-splice match; NIC: novel in catalog; NNC: novel not in catalog. **b** Structural subcategories for FSM and ISM transcripts. Isoforms are grouped based on TSS and TTS diversity, relative to their associated reference transcript.

the number of unique isoforms per cell that can be detected is generally very limited [55, 78, 149, 150], which precludes the generation of cell-specific transcriptomes. Given these pitfalls, this type of data is still far from suitable for comprehensive single-cell isoform characterization (see Chapter 3). Instead, the work presented in this chapter is based on the usage of bulk long-read data for isoform definition, followed by cell-level isoform quantification using compatible full-length, short-read scRNA-seq data and the Expectation Maximization (EM) algorithm in RSEM [85]. This hybrid strategy benefits from the advantages of long-reads for transcriptome reconstruction without relying on the scarce depth and quantitative information of single-cell applications, while simultaneously benefiting from the high sensitivity of full-length single-cell data and RSEM's ability to generate transcript-level expression estimates.

In this chapter, we describe the steps followed for the generation of the bulk long-read transcriptome (section 4.3.1) and its functional annotation (section 4.3.2). Next, we explain how cell-level isoform expression estimates were used to put a multi-group Differential Expression strategy into place 4.3.3, which will be crucial for the downstream analyses presented in Chapter 5. Last, but not least, the curation of bulk lrRNA-seq data also provided an opportunity for the development of improved transcriptome curation strategies. These are hereby characterized and compared to our previously employed manual filtering strategy (section 4.3.4), and ultimately resulted in additions to the SQANTI3 software (https://github.com/ConesaLab/SQANTI3) in the form of novel Filter and Rescue modules. All in all, this study managed not only unlock single-cell isoform analyses using a data integration approach, but also to improve transcriptome curation resources for the lrRNA-seq community.

4.2 Methods

4.2.1 Data availability

Single-cell, short-read RNA-Seq data from mouse primary visual cortex used in the analysis of neural broad types, generated by Tasic et al. [163], was downloaded from SRA accession SRP061902. Single-cell, short-read RNA-Seq data from mouse primary visual cortex used in the analysis of GABA neuron cell subtypes, generated by Tasic et al. [122], was downloaded from SRA accession SRP150473 after selecting accessions corresponding to GABA neurons and primary visual cortex tissue, described by authors in the study metadata available at GEO accession GSE115746.

Mouse reference genome and transcriptome used for long-read processing were downloaded from the RefSeq96 database (global release 96, annotation release 108, September 2019), from genome version GRCm38.p6 and assembly accession GCF_00001635.26.

Long-read datasets form mouse hippocampus and cortex, generated by Wyman et al. [159], were downloaded from ENCODE accessions ENCSR214HSG and ENCSR340GWV, respectively. Long read-defined transcriptome files (generated using above-cited long-read data and reference files, details in Supplementary Note) have been made available at the tappAS repository of annotation files. These include the GTF file used for quantification (https://app.tappas.org/resources/downloads/gtfs) and the tappAS-

formatted GFF3 file obtained after transferring functional features using isoAnnotLite (http://app.tappas.org/resources/downloads/gffs). Both files are named using the following file prefix:
Mus_Musculus_GRCm38.p6_PacBioENCODE_RefSeq108.

Several orthogonal data sources were used to perform transcriptome quality control. Mouse CAGE peak data was obtained from the FANTOM5 database [164, 165] and used to curate transcription start sites (TSS) for our isoforms. Consistent with previous reports stating that human and mouse polyA motif sequences show high levels of conservation [166, 167], we supplied a ranked list of common polyA motif sequences from human (included in SQANTI3) to curate transcription termination sites (TTS). Finally, to obtain short-read coverage information, and given the lack of matching Illumina data for these long read datasets, we sampled 20 cells from each of the 7 broad cell types in the Tasic et al. dataset and pooled all Illumina reads to generate a pseudo-bulk sample. We mapped this cell pool to the long read-generated transcriptome (genome version GRCm38.p6) using STAR [80] and the following parameters: `-outFilterType BySJout -outFilterMultimapNmax 20 -alignSJoverhangMin 8 -outFilterMismatchNmax 999 -outFilterMismatchNoverReadLmax 0.04 -alignIntronMin 20 -alignIntronMax 1000`.

4.2.2 Transcriptome reconstruction using long reads

Long read data pre-processing

Reads from both long-read RNA-seq samples and their replicates (2 replicates/sample) were first pooled and pre-processed using the IsoSeq3 pipeline (v3.3.0, https://github.com/PacificBiosciences/IsoSeq) (including inter-read correction step, i.e. polishing), with default parameters. Next, redundant isoforms were merged using TAMACollapse (https://github.com/GenomeRIK/tama/wiki/Tama-Collapse, [161]) in order to mitigate transcript-level redundancies. TAMACollapse was run with the following parameters: `-c 95 -i 85 -x no_cap -a 10 -m 10 -z 10`. The resulting mouse neural transcriptome was characterized using SQANTI2 (https://github.com/Magdoll/SQANTI2) and SQANTI3 Quality Control [162] and the supporting data sources outlined in section 4.2.1.

Transcriptome filtering and curation

An initial transcriptome strategy was based on the quality attributes computed by SQANTI2 Quality Control (https://github.com/Magdoll/SQANTI2) (v.7.4.0, March 2020). Several orthogonal data sources, including short-read, CAGE peak and polyA motif data (section 4.2.1 were used to compute TSS, TTS, junction support and expression information metrics. This information was subsequently used to define a two-step curation strategy to ensure the reliability of the retained transcript models.

First, the machine learning-based filter (ML filter) included in SQANTI [50] (v.1.0, March 2018) was applied, allowing automatic discrimination true isoforms from artifacts. In this process, a random forest classifier is trained using full-splice match (FSM) and incomplete-splice match (ISM) isoforms as the True Positive (TP) set and novel not in catalog (NNC) transcripts containing at least one non-canonical junction as True Negatives (TN), which constitute isoforms with patterns of high and low-quality attributes that will be learned and detected throughout the remaining long-read transcripts. The model hence uses 14 isoform-level features calculated by SQANTI (table 4.2) to discriminate high and low-quality isoforms across structural categories. Transcripts for which the obtained classifier probability was < 0.5 were flagged as artifacts and subsequently removed from the transcriptome.

To be able to detect artifacts from the full-splice match (FSM) and incomplete-splice match (ISM) categories (figure 4.1), SQANTI2 QC attributes, including those related to CAGE peak data, polyA motif information and reference annotation similarity, were used to evaluate redundancy. Redundancy was defined as 3' and 5' end variability leading to the detection of multiple FSM and/or ISM isoforms per reference transcript. First, transcripts that constituted the only matching FSM of a given reference transcript (unique-FSM) were automatically preserved, regardless of TSS/TTS support. In this manner, we prioritized evidence of detection of junction combinations present in reference transcripts over TSS/TTS definition accuracy. In cases showing multiple FSM per reference transcript (multiple-FSM), TSS and TTS end positions were evaluated to filter poorly-defined isoforms. Regarding the 5' end, we kept isoforms whose TSS was included within or situated up to 50bp downstream of a CAGE peak and whose 3' end region included a polyA motif no further than 50bp upstream of the TTS. Isoforms lacking CAGE or polyA support were preserved if their TSS matched or was situated up to 50bp downstream of an annotated TSS and their TTS also matched or was within ±50bp of an annotated TTS. TSS/TTS annotation information was extracted from the reference transcriptome. In this manner, we considered both data and reference-based evidence and included all possible combinations of these two sources in our transcriptome. For multiple-FSM cases where no FSM met the TSS/TTS requirements, we preserved the FSM transcript with the highest random forest probability score, as output by the ML filter in SQANTI. Finally, ISM isoforms were filtered according to polyA/CAGE and annotated TSS/TTS evidence.

Soft rescue strategy to mitigate the loss of expressed genes

Transcriptome filtering was found to result in gene loss, in cases where multiple isoforms had been detected using long reads, but none of them could be validated. We therefore performed a pseudo-rescue (i.e. soft rescue) process in which reference transcripts (RefSeq) associated to discarded FSM were added to the transcriptome, discarding the unsupported long read-defined isoforms. At this stage, we focused on removed genes that had multiple FSM isoforms associated to $\geq$2 reference transcripts, a condition that was enforced given that our planned study aimed to find co-expression relationships across multi-isoform genes (see chapter 5).

4.2.3 Functional annotation of long read-defined isoforms

Functional annotation is hereby understood as the process of predicting the function of the different elements (e.g. motifs, domains, sites) encoded by transcript and protein isoform sequences. In the case of the present study, long read-defined transcript models were annotated using IsoAnnotLite (https://isoannot.tappas.org/isoannot-lite/), a tool that incorporates information from sequence-based predictors and biological databases into any transcriptome, provided that a compatible functional annotation -i.e. available within the tappAS framework [121]- is supplied as the annotation source. In this scenario, the transcripts in the source annotation constitute *feature donors*, whereas the transcripts in the query annotation -often a long-read transcriptome- are the *feature acceptors*. The process, referred to as *positional transfer of features*, is described in detail below.

Functional categories and features

Since the nature of the sequence elements analyzed during the functional annotation process can be vastly different, these can be classified into various categories, which often exhibit several levels of hierarchy. In this thesis, we employ the terms *functional category* and *feature category* to refer to the top level in the annotation hierarchy. Categories may be equivalent to the database from which the information was retrieved (e.g. GeneOntology), to the computational tool used (e.g. RepeatMasker) or to a specific type of motif encoded in the sequence (e.g. 3' UTR motifs). Each functional category may include one or more *functional features*, depending on the specific definition of the category. For instance, nonsense-mediated decay (NMD) constitutes a functional category that only includes one feature (that is, NMD), whereas categories such as miRNA binding include multiple biological entities or feature identifiers (IDs) -in this case, binding sites for all miRNAs present in the database.

Even though the source functional annotations used to run IsoAnnotLite were generated by other authors (see previous work by de la Fuente [168]), a brief description of the sources of information they include is provided in table 4.1. For reference, table 4.3 includes a comprehensive list of functional categories and the database or predictor from which they originated.

Positional feature transfer algorithm

Positional information for feature acceptors, i.e. long-read transcripts, is first converted to genome positions using SQANTI3 characterization information. Similarly, transcript-level functional feature positions from the donor transcripts are transformed into genomic coordinates using the information in the reference GFF3 file. Next, functional features are transferred across transcript models by matching genomic positions, meaning that features from the donor transcripts whose genomic positions span a feature acceptor will be annotated as belonging to that transcript.

Importantly, different transfer rules have been implemented depending on the type of functional feature that is being handled. To transfer UTR features, genomic feature positions must be inside the transcript's exons and outside its CDS region. For CDS transcript features (namely transcript-level features

Database or predictor	Description	Website (if available)	Reference(s)
COILS	Prediction of protein coiled-coil conformation		[169]
CORUM	Database of protein complexes	https://mips.helmholtz-muenchen.de/corum/	[170]
GeneOntology	Database of gene functions	http://geneontology.org/	[171, 172]
miRWalk	Database of experimentally-validated miRNA-target interactions	http://mirwalk.umm.uni-heidelberg.de/	[173]
MOBIDB LITE	Prediction of intrinsically disordered regions (IDRs) in proteins	http://old.protein.bio.unipd.it/mobidblite/	[174]
NLS mapper	Prediction of protein nuclear localization signals (NLS)	https://nls-mapper.iab.keio.ac.jp/cgi-bin/NLS_Mapper_form.cgi	[175]
NMD	Prediction of transcript nonsense-mediated decay		[168, 176]
PAR-clip	Integration of PAR-clip data to infer RNA binding protein sites		[168]
PFAM	Database of protein domains and families	https://www.ebi.ac.uk/interpro/	[177, 178]
RepeatMasker	Scanner for the detection of repetitive sequences	https://www.repeatmasker.org/	[179]
scanForMotifs	Prediction of post-transcriptional regulatory elements in 3'UTRs		[180]
SIGNALP EUK	Prediction of signal peptides		[181]
TMHMM	Prediction of protein transmembrane domains	https://services.healthtech.dtu.dk/service.php?TMHMM-2.0	[182]
UniprotKB	Database of functional protein features (active sites, binding sites, etc.)	https://www.uniprot.org/help/uniprotkb	[183]
PhosphoSitePlus	Database of pos-translational modifications (PTMs)	https://www.phosphosite.org/	[184]
UTRsite	Database of cis-regulatory motifs in UTRs		[185]

Table 4.1: Functional information sources in the reference annotations used by IsoAnnotLite.

situated within the coding region), 1) the feature must be contained within the acceptor transcript's exons as well as inside the CDS region; and 2) if a feature has start and end positions situated in different exons, the end and the start of the exons for the donor and acceptor transcripts must be the same for feature transfer to occur. In the case of protein features, both donor and acceptor transcripts must be coding and share the same CDS. If all CDS exons are the same for both transcripts, all protein features are automatically transferred. If not, IsoAnnotLite requires the genomic positions of at least one CDS exon to be a partial match, that is, for the feature donor and acceptor to share part of one exon in the transcript's CDS. If at least one CDS genomic region overlaps between both transcripts, IsoAnnotLite checks for protein features that fall within that region and can therefore be transferred. For gene-level characteristics (e.g. Gene Ontology terms), information is always transferred across matching gene IDs. Finally, IsoAnnotLite verifies whether the same feature has been transferred from multiple donor transcripts to the same acceptor, and performs the removal of duplicated annotations.

Running IsoAnnotLite

IsoAnnotLite requires four inputs, namely the `*classification.txt`, `*junctions.txt` and `*corrected.gtf` files output by SQANTI (where the latter is the query annotation), and a source annotation, which must be a tappAS-compatible GFF3 file. For this study, the required soure annotation file was generated by combining two source transcriptomes: the mouse RefSeq functional annotation available in tappAS (RefSeq78) and a mouse neural transcriptome belonging to a pool of neural progenitors and cells derived from them, obtained from [168]. Specifically, several functional categories were selected from each GFF3 file (see table 4.3 in section 4.3.2), based on the quality and completeness of the original annotation. For reproducibility, the unfiltered GFF3 files, together with the merged annotation, are available at https://app.tappas.org/resources/downloads/gffs/. After filtering the GFF3 files and merging the annotations, we run IsoAnnotLite (v2.7.3) adding two non-default flags: `-intronic` to allow feature transfer even of regulatory motifs included in intronic regions and `-novel` to make the process independent from the associations between long-read and reference transcripts.

Metrics for functional annotation characterization

To understand the results of the feature transference process, the functional annotation generated by IsoAnnotLite was compared to both source annotations, computing a number of metrics per functional category:

1. Total number of functional features transferred for each category. Feature IDs were considered as many times as they had been annotated, independent of whether the same ID had been transferred to the same transcript multiple times. This metric works as an indicator of the frequency of transfer, irrespective of how many features the functional category had in the source annotation.

2. Percentage of features that had been transferred, relative to the total number of features in the source annotation (computed using (1)). This metric serves as an estimate of feature transfer success.

3. Percentage of transcripts in the long read-defined transcriptome that had received at least one feature from a given functional category. This metric captures functional annotation coverage in the long read transcriptome.

4.2.4 SQANTI3 machine learning-based filter

To improve the initial long-read transcriptome filtering strategy, the machine learning-based filter in SQANTI [50] (ML filter) was expanded to incorporate the novel features in SQANTI3 [162] in order to provide a comprehensive filtering strategy that eliminates the need for threshold-based, *ad-hoc* curation (section 4.2.2). This enhanced version of the ML filter was implemented as collection of R and python scripts and included in the SQANTI3 toolkit (https://github.com/ConesaLab/SQANTI3) as of v.5.0. For ease of reference, we hereby provide a full overview of the filter workflow (figure 4.2) and a list of the QC attributes used in the model training process for the SQANTI and SQANTI3 versions of the ML filter (table 4.2), however, only the novel features and modifications made with respect to the first release of the SQANTI software (https://github.com/ConesaLab/SQANTI) will be described in detail in this section.

Similarly to the original release, the updated SQANTI3 ML filter is based on the training of a random forest model and its subsequent usage to classify isoforms and artifacts based on features learned from True Positive (TP) and True Negative (TN) isoform sets, respectively. The tool, however, was modified to accept user-defined TP and TN lists, while simultaneously including built-in selection of training data using SQANTI3 categories and

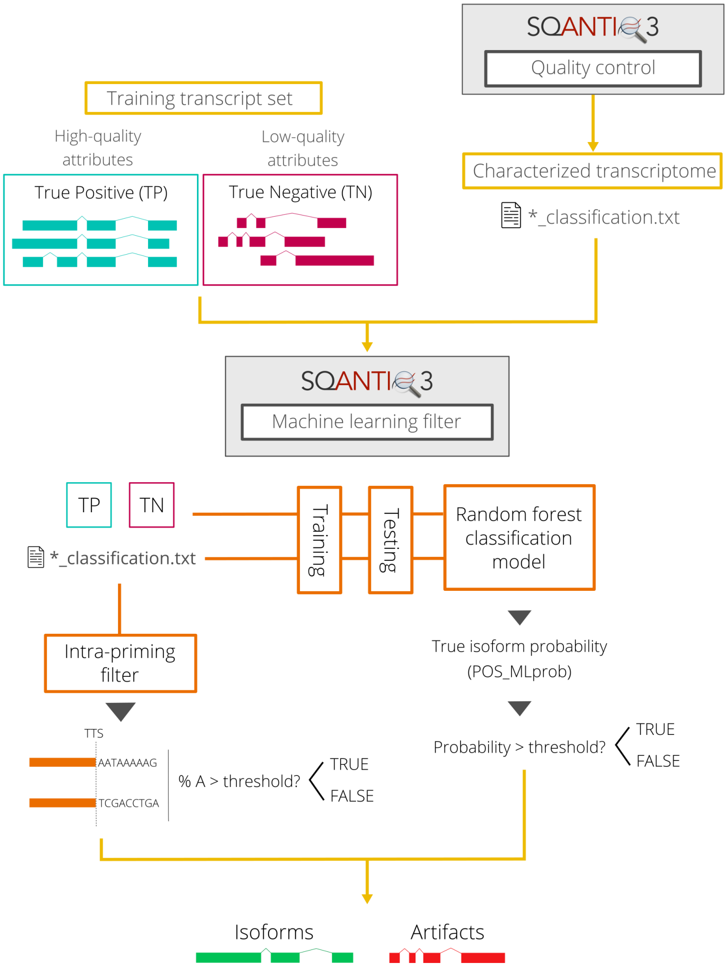


Figure 4.2: SQANTI3 machine learning filter workflow. Transcriptome quality attributes computed by SQANTI3 QC, together with a set of true positive and true negative isoforms, are used to train a random forest classifier. The model is then used to compute probabilities and, based on a user-defined threshold, flag reliable isoforms and artifacts. An intra-primming filter is simultaneously applied. Results are merged to create a final list of excluded and included transcripts.

subcategories: by default, NNC non-canonical isoforms are used as true negatives and reference match (RM) transcripts (figure 4.1.b) as true positives. Additionally, both groups are now balanced automatically, downsizing the largest to match the number of transcripts in the shortest training list. Notably, the maximum number of elements allowed can be adjusted by the user to alleviate the computational burden of model training. After training, cross-validation and testing (see [50] and the SQANTI3 Filter documentation https://github.com/ConesaLab/SQANTI3/wiki/Running-SQANTI3-filter for details), the resulting model is used to obtain the probability to classify each of the transcripts as isoforms or artifacts. The SQANTI3 implementation of the ML filter was designed to be more stringent on the isoform than on the artifact condition, requiring that the probability that the transcript is a true isoform be $\geq$0.7 by default. This threshold, nevertheless, can be modified by the user. Finally, the ML filter now allows users to force the inclusion or exclusion of some specific isoform groups. For FSM transcripts, users may indicate that all FSM should be included as true isoforms in the filtered transcriptome. In addition, since mono-exonic transcripts are not evaluated during the ML filter due to the lack of junction-related attributes, these can be automatically removed from the transcriptome if desired.

As an additional improvement, the SQANTI3 ML filter can make use of TSS and TTS validation metrics and orthogonal data, i.e. CAGE peak data, the short read-based TSS ratio metric [162], polyadenylation (polyA) motif information and polyA peaks obtained with technologies such as Quant-Seq [186]. As a result, and in contrast to the previous release, the SQANTI3 ML filter can be used to detect artifacts belonging to the FSM and ISM categories, which may present poorly-defined TSS and TTS in spite of showing splice junctions (SJs) with high quality attributes. By default, the enhanced filter uses all variables in the SQANTI3 classification file except those related to genome structure (i.e. chromosome/strand), associated reference transcript or gene and SQANTI categories and subcategories. However, this behavior can be modified by the user to further exclude variables, a feature designed to prevent overfitting in cases where one or more of these variables have been used to define the TP and TN sets. For instance, the default usage of the RM subcategory as TP, which is defined based on the distance to the 5' and 3' ends of the associated reference transcript, is by default coupled with exclusion of these two distance variables.

4.2.5 Filtering of potential artifacts using the SQANTI3 ML filter

As a way to test and characterize the improved SQANTI3 ML filter (v5.1.2), the tool was run on the collapsed transcriptome, resulting from the application of IsoSeq3 followed by TAMACollapse (section 4.2.2). The output files obtained after SQANTI3 QC using the supporting data in section 4.2.1 were used as input. The results were subsequently compared to the previously generated transcriptome (section 4.2.2) in order to understand the advantages and pitfalls of using an automated vs a manual filtering strategy.

The definition of True Positive (TP) and True Negative (TN) transcript model sets is critical to ensure the reliability of the filtering process. Given the availability of different sources of TSS/TTS orthogonal data, several QC

Table 4.2: Usage of Quality Control attributes in the ML filter in SQANTI vs SQANTI3. Attributes are listed as named in the SQANTI3 classification table. For reference, a brief description of each variable is provided as per the tool's documentation.

QC attribute	SQANTI ML filter	SQANTI3 ML filter	Description
length	✓	✓	Isoform length
exons	✓	✓	Number of exons
diff_to_TSS		✓	Distance of query isoform 5' end to reference transcript start site.
diff_to_TTS		✓	Distance of query isoform 3' end to reference transcript end site.
diff_to_gene_TSS		✓	Distance of query isoform 5' end to the closest start site of any transcript from the same gene
diff_to_gene_TTS		✓	Distance of query isoform 3' end to the closest end site of any transcript from the same gene
RTS_stage	✓	✓	Indicates whether any of the splice junctions has been flagged as a potential RT switching artifact
min_sample_cov	✓	✓	Sample with the minimum short-read coverage
min_cov	✓	✓	Minimum short-read coverage found across splice junctions
min_cov_pos	✓	✓	Position of the splice junction showing the lowest coverage
sd_cov	✓	✓	Standard deviation of short-read junction coverage across supplied short-read data samples
FL	✓	✓	Long-read full-length counts associated to an isoform in each sample
n_indels	✓	✓	Total number of indels based on long-read alignment
n_indels_junc	✓	✓	Number of junctions in this isoform that have alignment indels near the junction site
bite	✓	✓	Indicates whether there are junctions whose associated intron completely overlaps an annotated intron as well as part of the flanking exons, creating a novel splice junction
iso_exp	✓	✓	Short-read isoform expression
gene_exp	✓	✓	Short-read gene expression
ratio_exp	✓	✓	Ratio of iso_exp and gene_exp.
FSM_class	✓	✓	Classifies transcripts according to the expression of other isoforms in the same gene: single isoform (A), multi-isoform with no FSM (B), multi-isoform, at least one FSM (C)
coding	✓	✓	Coding potential predicted using GeneMarkS-T
predicted_NMD		✓	A transcript is predicted as NMD if the ORF ends at least 50bp before the last junction

QC attribute	SQANTI ML filter	SQANTI3 ML filter	Description
perc_A_downstream_TTS		✓	Percent of genomic A's in the 20bp window downstream of the TTS
dist_to_CAGE_peak		✓	Distance to CAGE peak situated closest to the TSS
within_CAGE_peak		✓	Indicates whether the TSS is situated within a CAGE peak
ratio_TSS		✓	Short-read coverage ratio between the 100bp downstream and upstream of the TSS
dist_to_polyA_site		✓	Distance to the polyA site situated closest to the TTS
within_polyA_site		✓	Indicates whether the TTS is situated within a polyA site
polyA_dist		✓	Difference between the putative polyA site and the detected polyA motif

attributes were used to define the TP and TN set. Specifically, we selected a set of 3,000 FSM multi-exonic isoforms showing 5' end support by CAGE data (`within_CAGE_peak = TRUE`), a detected polyA motif at the 3' end (`polyA_motif_found`, i.e. polyA motif found up to 50bp upstream of the TTS) and exhibiting only canonical junctions (`all_canonical = TRUE`). To define the TN set, we considered 3,000 isoforms from all non-FSM categories lacking 5' end support by CAGE, a detected polyA motif or containing a non-canonical junction. Consequently, the `dist_to_CAGE_peak`, `within_CAGE_peak`, `polyA_motif_found`, `polyA_dist`, `polyA_motif` and `all_canonical` (see table 4.2), which were used to constitute the test sets, were excluded from model training to prevent overfitting and bias towards these variables in classifier performance. The SQANTI3 ML filter was subsequently run using these input sets and the classification and GTF files generated by SQANTI3.

4.2.6 SQANTI3 rescue strategy

The application of the pseudorescue strategy described in section 4.2.2 revealed the importance of post-filter refinement to avoid losing complexity in long read-defined transcriptomes. The goal is to avoid the loss of transcripts and genes that constitute part of the transcriptional signal, but for which a correct transcript model could not be generated when processing the long-read data. To extend and generalize the methodology in section 4.2.2, a complete rescue pipeline was designed and integrated as a new module within SQANTI3 as of v.5.1 of the software.

The novel Rescue module is designed to be run after SQANTI3 Filter and operates by selecting a replacement transcript from the reference transcriptome, which is ultimately added to the set of long read-defined, filter-passing isoforms to generate an expanded, final version of the transcriptome. This strategy is based on two principles: consistent quality, meaning that rescued transcript models should meet the QC criteria of the filtering settings used to call isoforms, and non-redundancy, meaning that when the identified replacement transcript for a given artifact is already part of the filtered transcriptome, no transcript model is added. The algorithm operates in four steps, described in detail in the next sections.

Step 1: automatic rescue and selection of rescue candidate and target transcripts

The first step in SQANTI3 rescue consists in the retrieval of isoforms with reference-supported junction chains for which TSS/TTS could not be vali-

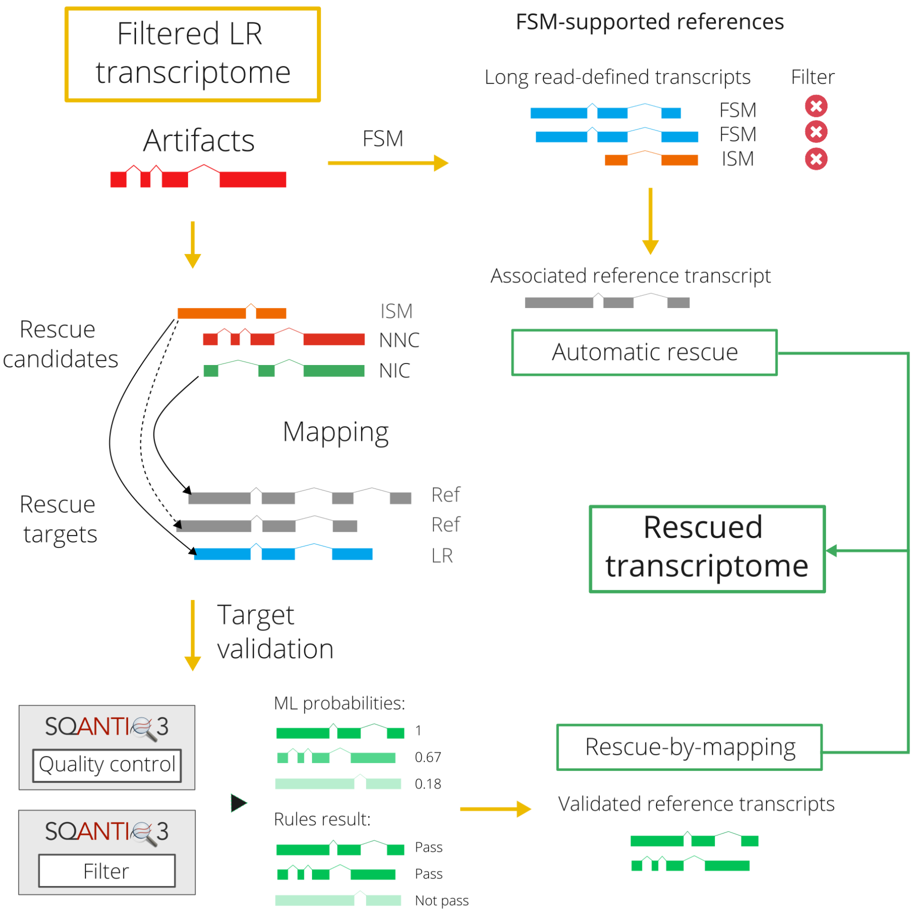


Figure 4.3: Summarized SQANTI3 rescue workflow. The rescue pipeline consists in four steps, including 1) automatic rescue, 2) mapping of rescue candidates, 3) validation of reference transcriptome targets and 4) rescue target selection. As a result, the module provides a rescued transcriptome, including previously validated isoforms and replacing artifacts with matching reference transcripts.

dated. This applies to FSM artifacts, which are often generated as a result of 3'/5' end definition inaccuracies. In the most severe cases, FSM filtering results in the removal of all representatives of a given reference transcript. To mitigate this, reference transcripts matching the junctions of filtered FSM are retrieved and added to the transcriptome. To avoid introducing unwanted redundancy, the reference transcript is added only once in cases where multiple FSM artifacts have the same associated reference transcript, i.e. same junctions, but different TSS.

This process, however, leaves out artifact transcripts from the ISM, NIC and NNC categories. These are considered *rescue candidates* and will continue to be analyzed by the rescue pipeline, as long as the following criteria apply:

- ISM artifacts are only considered if they do not have any FSM counterparts associated to the same reference transcript. This case therefore corresponds to non-FSM supported reference transcripts.
- Artifacts from the NIC and NNC categories are always included in the rescue candidate group, since SQANTI3 QC generates no information on their association with reference transcripts.

Similarly, all long-read and reference isoforms from the same genes as the selected rescue targets are selected as the group of *rescue targets* from which matching replacement isoforms will be found.

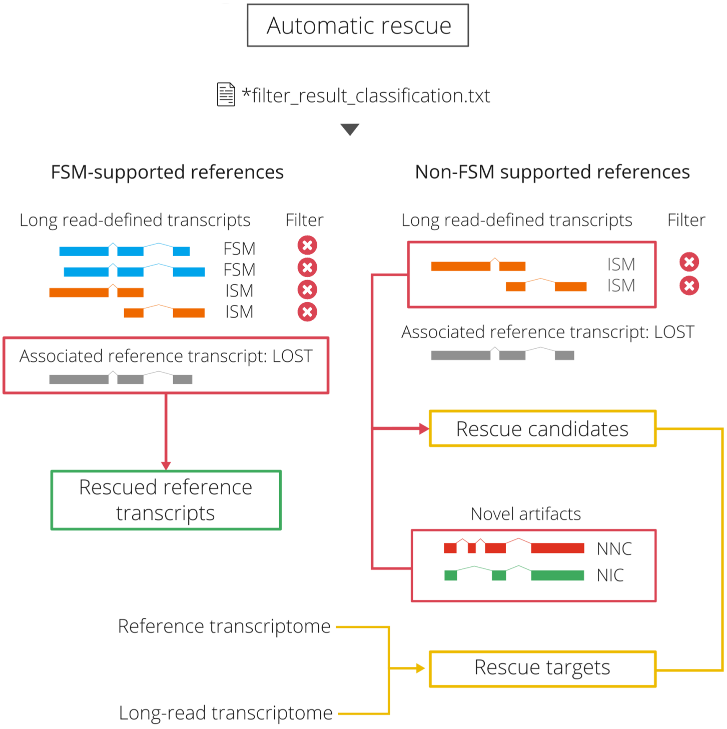

Automatic rescue
*filter_result_classification.txt
FSM-supported references
Non-FSM supported references
Long read-defined transcripts
Filter
FSM
FSM
ISM
ISM
Associated reference transcript: LOST
Rescued reference transcripts
Long read-defined transcripts
Filter
ISM
ISM
Associated reference transcript: LOST
Rescue candidates
Novel artifacts
NNC
NIC
Reference transcriptome
Long-read transcriptome
Rescue targets

Figure 4.4: Automatic rescue. Reference transcripts for which all associated FSM and ISM isoforms have been flagged as artifacts during filtering are automatically included into the rescued transcriptome. When reference transcripts have only been associated to discarded ISM isoforms, these are added to the rescue candidate list, together with artifact NIC and NNC transcripts.

Step 2: mapping of rescue candidates to same-gene targets

Matches between each rescue target and its same-gene candidates are next found by mapping candidate sequences to target transcripts. To achieve this, minimap2 [187] was set to long-read alignment mode using the `-a` parameter, combined with the `-x map-hifi` preset option (i.e. PacBio high-fidelity read alignment) to reflect the accuracy of the processed transcript sequences. Secondary alignments were allowed and set to the default number of 6 to allow multiple mappings to be reported per candidate. This process yields a series of alignments that pair each rescue candidate to multiple possible targets, pairs that are hereby referred to as *mapping hits*.

Step 3: validation of reference transcriptome targets

To ensure that reference transcriptome targets included in the rescue process comply with the same quality requirements as long-read targets and minimize the risk of retrieving non-sample-specific transcripts, SQANTI3 QC and Filter modules are run on the reference transcriptome supplying the same additional data and quality criteria as used for the long-reads transcript models, with a few adaptations. First, as full-length read count data is unavailable for the reference transcriptome, this will not be included as an orthogonal data source. In addition, the QC script is run setting `-min_ref_len 0` in order for short reference transcripts (<200bp) to be correctly classified as FSM. Finally, SQANTI3 Rescue applies the pre-trained ML filter classifier used to filter long-read transcripts, establishing the same probability threshold to call isoforms/artifacts ($\geq$0.7).

Step 4: rescue-by-mapping and target selection

Finally, mapping hits and reference transcriptome filter results are integrated to obtain a selection of transcripts for inclusion in the final transcriptome. Several criteria are applied to evaluate different aspects of rescue target suitability:

1. Validation by orthogonal data: mapping hits are not considered if the rescue target did not obtain a sufficiently high isoform probability. If

multiple mapping hits passed the filter, the target transcript with the highest ML filter probability (long-read or reference) will be considered to be the best match for the candidate and therefore selected for rescue.

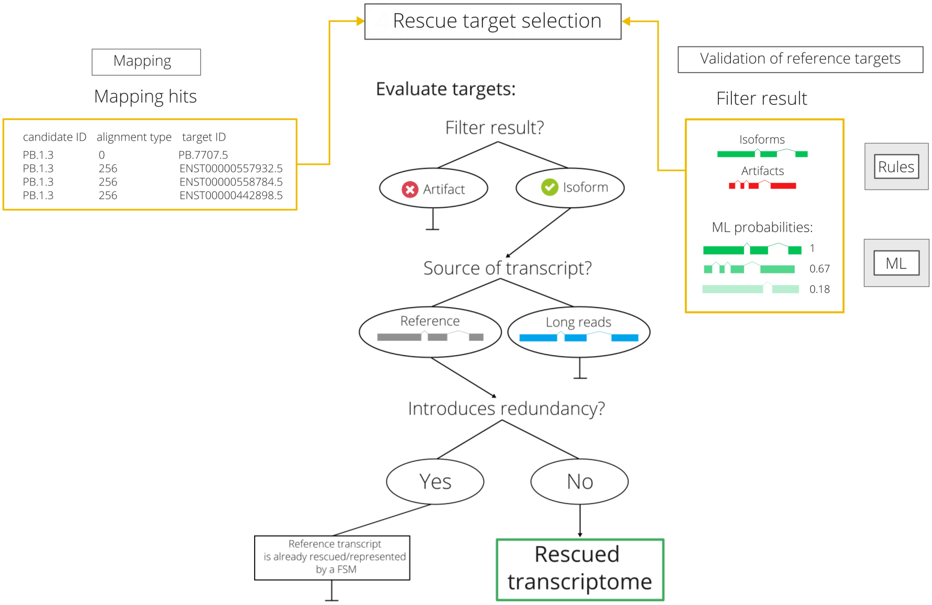
2. Removal of long-read rescue targets: those cases where the best match target was already included in the long-read transcriptome will not be further considered. If a rescue candidate matched a transcript that is already present, it could be considered as an artifact of that validated transcript, which then becomes its representative.

3. Removal of redundant reference transcripts: during the rescue process, some of the selected reference targets may already be represented by a same-gene FSM or by a transcript retrieved during automatic rescue. In this situation, the best matching transcript for the artifact is already included in the transcriptome, and no further action is needed.

Rescue candidates that pass all three filters are to become part of an expanded transcriptome, which is the final output of the SQANTI3 transcriptome curation pipeline.

4.2.7 Rescue of ML-filtered artifacts using SQANTI3

In order to run SQANTI3 Rescue, the reference transcriptome (see section 4.2.1) was first characterized using SQANTI3 QC. The same orthogonal data sources as for long-read transcriptome QC were used, excluding long-read counts, and the same GTF was supplied as both query and reference, with the parameter adjustments described in section . SQANTI3 Rescue (v5.2.1)



candidate ID	alignment type	target ID
PB.1.3	0	PB.7707.5
PB.1.3	256	ENST00000557932.5
PB.1.3	256	ENST00000558784.5
PB.1.3	256	ENST00000442898.5

Figure 4.5: Rescue target selection. Mapping results and reference transcriptome validation generated by SQANTI3 Filter are integrated to select suitable rescue targets to be included in the final transcriptome.

was run using the SQANTI3 QC classification file from the reference and the SQANTI3 Filter output table as inputs, as well as the pre-trained random forest classifier obtained after runnning the ML filter (see section 4.2.5). Finally, the rescued transcriptome was re-evaluated using SQANTI3 QC and the same supporting data sources as in the first run.

4.2.8 Single-cell data pre-processing and quality control

The mouse neural single-cell RNA-Seq dataset used in this study (mouse primary visual cortex [163]) consisted in single-end, Illumina reads generated with the Smart-seq2 protocol [34], which enables isoform-level quantification (as discussed in chapter 3). Reads were mapped to the mouse genome (GRCm38.p6) using STAR [80]. We performed expression quantification of the long read-defined isoforms using RSEM [85], and used the labels provided by Tasic et al. in the original study [163] to assign 1,679 cells to 7 broad cell types: 5 glial (microglia, endothelial cells, oligodendrocytes, oligodendrocyte precursor cells (OPCs) and astrocytes) and 2 neural (GABA-ergic and glutamatergic neurons).

The effect of isoform length on expression was evaluated using the NOISeq R package [188], where mean expression was shown to be highly correlated with transcript length (adjusted R^2 = 0.81; p-value = 2.2e-16). Using isoform i effective length (l_i) and cell-level j estimated counts (c_{ij}), both output by RSEM and, after testing several alternatives, we devised a custom formula (equation 4.1) to minimize the impact of length on isoform expression for each isoform i:

$$y_{ij} = \frac{c_{ij}}{(10^{-6}\sum_{i=1}^{I} c_{ij})\sqrt{10^{-3}l_i}} \tag{4.1}$$

The transformed expression value for isoform i in cell j (y_{ij}) was again tested for length bias and a low correlation was found (adjusted R^2 = 0.25; p-value = 6.84e-8). Next, we inspected the library size distribution and filtered both high and low-count outliers due to potential premature cell death or library preparation duplets, with a total of 1591 cells passing quality control. Feature-level quality control was performed in a cell type-aware manner, keeping isoforms that showed non-zero expression in at least 25% of one cell type. Out of the 36986 isoforms and 12692 genes in the PacBio-defined transcriptome, we retained 16240 isoforms and 8814 genes for downstream analysis.

4.2.9 Single-cell multi-group Differential Expression analysis

Differential Expression (DE) analysis among the 7 cell types was performed by combining ZinBWaVE weights [189] and bulk-designed DE methods edgeR [190] and DESeq2 [191], which enable multiple group testing and were among the best-performing methods when combined with the ZinBWaVE method. Briefly, ZinBWaVE calculates cell-level weights for each isoform, effectively downweighting zeros during modelling for differential expression in single cell data (see van den Berge et al. [189] for details), and hence unlocking bulk RNA-Seq computational methods for single-cell data. Of note, generalized linear models (GLM) within edgeR and DESeq2 were built and run following the pipeline used by van den Berge et al. to make them suitable for single-cell RNA-seq data, and are implemented in a wrapper function within the acorde R package as described in 4.2, where y_{ij} is expression of isoform i in cell j, T_{kj} is a dummy variable which takes value 1 when cell j is assigned to cell type k $(k = 1, \cdots, K)$ and 0 otherwise, β_{ki} are the regression coefficients for isoform i, ϵ_{ij} represents the error term, and $h()$ is the link function of the GLM (natural logarithm in this case).

$$h(y_{ij}) = \beta_{0i} + \sum_{k=2}^{K} \beta_{ki} T_{kj} + \epsilon_{ij} \tag{4.2}$$

Differential Expression was defined using a significance threshold of FDR<0.05 when testing the significance of the model for each isoform i, that is, $H_0 : \beta_{2i} = \cdots = \beta_{Ki}$. Isoforms considered DE were preserved for downstream analysis if detected by at least one of the two methods edgeR or DESeq2, since this indicated a change in expression for any of the cell types considered rather than a flat expression profile.

4.2.10 Neural cell sampling strategy to select a consensus set of DE isoforms

Prior to DE testing, and to balance sample sizes across cell types, we performed 50 independent runs of random neural cell sampling (without replacement) followed by zero-expression filtering (expression above zero for at least 25% of cells in at least one cell type to control zero abundance among iterations and avoid problems during GLM modelling) and DE testing with edgeR [190] and DESeq2 [191]. Specifically, the two neural cell types (GABA-ergic neurons, n = 729; glutamatergic neurons, n = 711 cells) were downsampled by randomly selecting 45 cells, keeping N=241 cells for multi-group DE testing.

To measure the consistency of each method independently, we calculated the mean and standard deviation of the number of DE isoforms across all same-method sampling runs ($R = 50$). To check the level of within-method agreement, we next considered isoform IDs labeled as DE in each independent method run, and calculated the Jaccard Index (J_{rs}) between DE results of that same method for all possible pairs of random sampling runs r and s ($r, s = 1, \cdots, 50$, $r < s$, a total of 1225 comparisons). To summarize this information, we relied on the mean and standard deviation of these two sets of J_{rs} values. Finally, we measured the level of agreement between edgeR and DESeq2 regarding our DE criteria, that is, considering isoforms detected by at least one of the methods to be significantly DE (FDR<0.05). To achieve this, we calculated the union of DE isoforms between one-to-one pairs of edgeR and DESeq2 runs ($R = 50$), and computed the Jaccard Index between all possible pairwise combination of global DE results, i.e. isoforms detected by at least one method (again, 1225 comparisons).

Using the 50 independent set of DE results, including both edgeR and DESeq2, we set out to define a consensus set of DE isoforms. To maximize sensitivity, we first considered the union of DE isoforms obtained by edgeR and DESeq2 on each of the downsampled versions of the data ($R = 50$), that is, isoforms detected to be significantly DE (FDR<0.05) by at least one of the applied methods. Among the 50 lists of DE isoforms, those that were significant in at least 50% of the runs were selected. In addition, minor isoforms, i.e. those accumulating less than a 0.1 proportion of the absolute expression of their gene, were filtered. Finally, we retained only isoforms from genes with more than one DE isoform, hence removing cases where no AS-directed, isoform-level co-expression relationships can be established. This sets a general requirement for the entire study, which is that all isoforms retained must have, at all times, at least one same-gene counterpart to establish regulatory relationships that can be based on differential splicing of that gene, given that no AS regulation can be detected if a gene's total expression is represented by a single isoform.

4.3 Results

4.3.1 Defining high-quality long read transcripts: insights from pre and post curation

As described in Chapter 3 and in section 4.1, long-read technologies provide most of the required conditions for successful isoform-level analyses, however, its application for single-cell transcriptomics has been hindered by sequencing depth constraints [37]. Given the severity of these limitations, we chose to use bulk-level data (section 4.2.1) to build a mouse neural transcriptome. Specifically, PacBio long-reads were supplied to the IsoSeq3 pre-processing pipeline (section 4.2.2), which generated a total of 178,507 isoforms from 20,307 genes. Transcript characterization using the SQANTI classification scheme (figure 4.1) revealed that the transcriptome was highly enriched in incomplete splice match (ISM) transcripts (figure 4.6.a). To avoid an overestimation of the number of isoforms per gene (figure 4.6.b), redundant transcript models were collapsed using TAMA [161]. This strategy mitigates transcript-level redundancies by merging isoform models that differ slightly due to RNA degradation or sequencing artifacts, but are very likely to have originated from the same transcript in the sample. As a result, 93,698 unique isoform models were obtained, collapsing a big proportion of ISM (figure 4.6.a) and successfully reducing the within-gene complexity of the transcriptome (figure 4.6.b).

In order to enhance the quality of the long-read transcriptome, a dual curation strategy was applied, starting with the application of the SQANTI ML filter

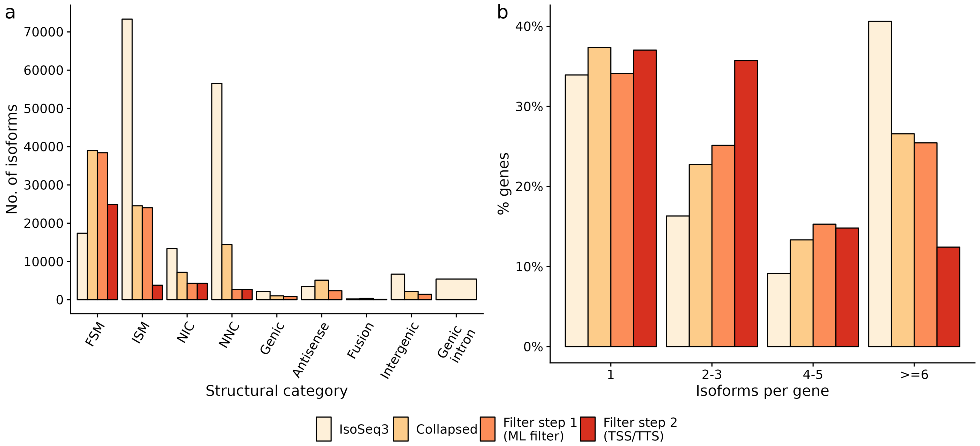


Figure 4.6: Transcriptome overview after successive processing and filtering steps, including IsoSeq3 pre-processing, collapse by TAMA, filtering using SQANTI ML filter and filtering based on CAGE/polyA data support. **a** Isoform distribution across structural categories. **b** Number of isoforms per gene. FSM: full-splice match, ISM: incomplete-splice match, NIC: novel in catalog, NNC: novel not in catalog.

and followed by manual filtering using short-read, CAGE and polyA motif data to evaluate isoform support (see section 4.2.2). The rationale behind this combination of filters is lays in the design of the first release of the SQANTI software and the ML filter. Originally, the SQANTI quality control (QC) process was largely based on short-read coverage of the junctions, with many of the computed quality features being associated with this data source. This precluded the validation of novel TSS and TTS and prevented the removal of unreliable FSM and ISM transcripts, which were often fully supported at the SJ level. The random forest classifier was trained using full-splice match (FSM) and incomplete-splice match (ISM) isoforms as true positives (TP), thus ignoring the low-quality properties of these categories upon ML-based filtering. To mitigate this, we benefited from the integration of region-level data (CAGE-seq data, polyA motifs) in SQANTI2 to create a series of rules for TSS and TTS support, unlocking the removal of FSM and ISM isoforms with false start and end sites from the transcriptome.

During the first step of curation, 19,550 isoforms were flagged as artifacts by the ML filter, resulting in 74,148 isoforms from 17,026 genes being retained in the transcriptome. In this process, ∼40% NIC and ∼80% of NNC isoforms were classified as artifacts, whereas isoforms from the remaining minority categories were subject to equally stringent filtering (figure 4.6.a). This result aligned with the ML filter's adherence to junction-level properties and with the SQANTI evaluation of the collapsed transcriptome (i.e. before filtering), which showed accumulation of non-canonical SJ in NNC (figure 4.7.a) and generally low short-read support for this junction type (figure 4.7.b). Novel canonical SJ, found in both NIC and NNC transcripts (figure 4.7.a), showed similar properties (figure 4.7.b). In line with this, NIC and NNC isoforms discarded after applying the trained classifier were enriched in low coverage junctions (figure 4.7.c) and novel splice sites (figures 4.7.a-b), highlighting the effectiveness of ML-driven filtering to detect junction-level artifacts.

The variability at the 3' and 5' end among FSM and ISM can result in the detection of multiple long-read isoforms per reference transcript, a phenomenon that we hereby refer to as *redundancy*. Importantly, redundancy can arise from true TSS/TTS diversity or stem from library preparation artifacts, such as RNA degradation or intra-primming. In our transcriptome, high levels of redundancy were observed for both categories before the second step of curation (figure 4.8.a). To mitigate this, the amount of redundancy that

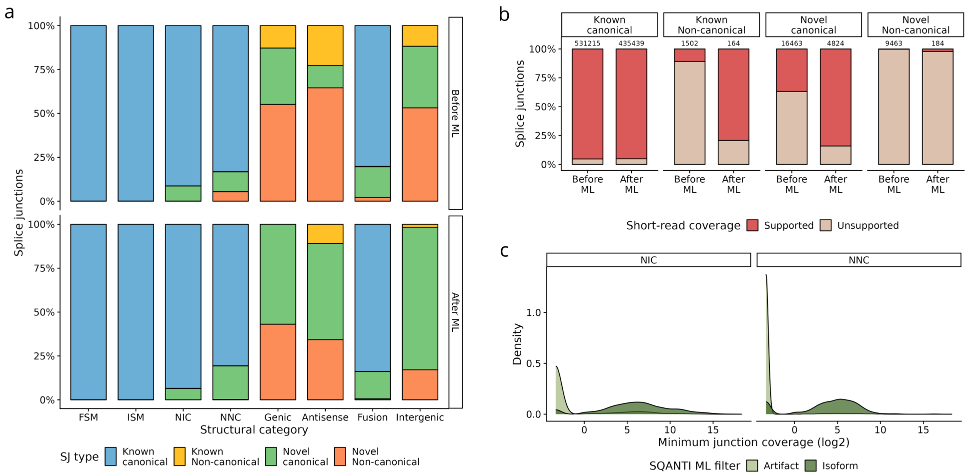


Figure 4.7: Junction-level properties of collapsed transcriptome and influence in ML filter artifact selection. a Splice junction type (%) present in each SQANTI structural category, before and after the application of the ML filter. **b** Percentage of junctions with or without short-read support for each type of SJ, before and after the application of the ML filter. Supported junctions are defined as total those covered by at least one short-read, i.e. total coverage geq 1. Total number of SJ in each type before and after filtering is shown above the corresponding bars in the plot. **c** Density distribution of the minimum short-read coverage of SJ per transcript (computed using the `min_cov` attribute in SQANTI, see table 4.2, shown for NIC (left) and NNC (right) isoforms and stratified by SQANTI ML filter result. FSM: full-splice match, ISM: incomplete-splice match, NIC: novel in catalog, NNC: novel not in catalog.

could be confidently preserved was evaluated using CAGE peak and polyA motif data, as well as reference annotation similarity. Briefly, FSM were preserved in the transcriptome according to three criteria, in the following order of priority: (a) detection of a reference-compatible FSM transcript, in cases where there was no redundancy; (b) filtering of FSM with no support for TSS (CAGE peak or reference TSS similarity) and TTS (polyA motif or reference TTS similarity), if there was redundancy; (c) selection of the FSM that best matched the reference transcript, when there was redundancy but no 3'/5' end support for any of the FSM. For ISM, given that missing junctions from matching reference transcripts increase the probability of false positive TSS/TTS, we only applied criteria (b). For a detailed description of the filtering rules, see section 4.2.2.

As a result of these constraints, 35,700 isoforms from 12,183 were preserved in the transcriptome. This filtering strategy not only managed to enrich the transcriptome in FSM (24,915 total FSM, ~67%, figure 4.6), but also pointed out the importance of combining multiple criteria to prevent the loss of relevant FSM transcripts that would have been lost based solely on 3'/5' end support. Specifically, ~50% of FSM remained in the curated transcriptome based on unique FSM or best match criteria (figure 4.8.b). Furthermore, filtering effectively reduced redundancy in both FSM and ISM while simultaneously validating several novel TSS and TTS for a relevant proportion of reference isoforms (figure 4.8.a). Part of the 3'/5' diversity in the long read-defined transcriptome could therefore be attributed to true TSS and TTS.

Since different data sources were combined in the second step of transcriptome curation, we decided to inspect whether supporting data or agreement with the reference annotation were the most frequent reasons for inclusion in the transcriptome. In doing so, we found only moderate levels of agreement between CAGE and polyA data and the reference annotation, highlighting the importance of supporting data for transcriptome curation. In the case of FSM, more than 50% of validated TSS and TTS could only be confirmed using CAGE and polyA motif data (figure 4.8.c) a proportion that was even higher for some ISM subcategories (figure 4.8.d). For instance,

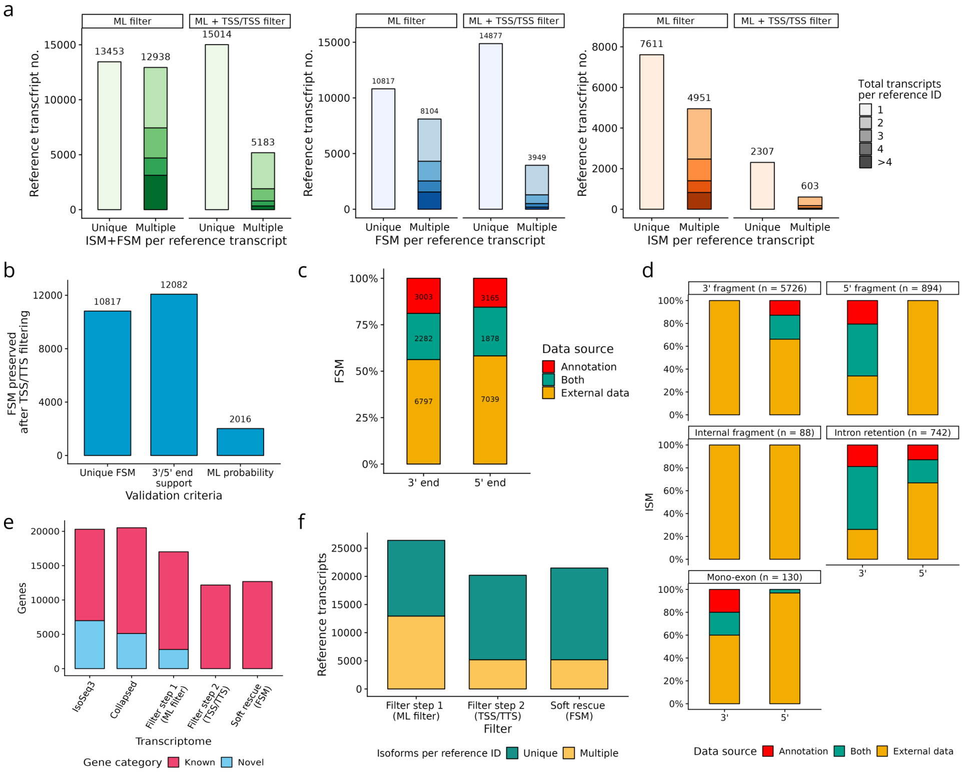


Figure 4.8: Characterization of redundancy and filtering performance for FSM and ISM transcripts. **a** From left to right, redundancy levels for FSM+ISM, FSM only and ISM only for all unique reference transcripts associated to long-read isoforms by SQANTI2, before and after filtering using TSS/TTS evidence. **b** FSM included in the transcriptome after filtering, stratified by validation criteria. Supporting data sources by which the 3' end (TTS) and 5' end (TSS) of **c** FSM and **d** ISM transcripts were validated. External data sources correspond to CAGE peaks (TSS) and polyA motif data (TTS). **e** Number of known and novel genes included after each transcript consruction step, including soft rescue. **f** Number of unique reference transcripts represented by FSM and ISM before and after TSS/TTS filtering, and after conducting soft rescue. FSM: full-splice match, ISM: incomplete-splice match.

internal fragments could only be validated using CAGE or polyA information (figure 4.8.d). All in all, these analyses suggest that reference transcripts may contain multiple, unannotated TSS and TTS that can only be observed using long reads.

Data-driven filtering also resulted in the loss of entire genes (figure 4.8.e) when no long read-defined isoforms could be validated using CAGE and polyA data. A number of these genes, however, had associated FSM transcripts, which was interpreted a strong indicator that the reconstructed isoform models were reliable at the junction level. In these cases, we devised a soft rescue strategy by which the reference transcripts associated to these discarded FSM were added to the transcriptome. As a result, we added 1,286 missing isoforms and 509 genes for which there was evidence of alternative isoform expression, even if their isoforms could not be confidently defined using long reads (figure 4.8.f). Finally, no isoforms from novel genes were preserved, given their lack of 3'/5' end support (Figure 4.8.d), whereas for minority categories, checking for CAGE/polyA and annotation support resulted in the removal of almost all isoforms (figure 4.6.a). All in all, 36,986 isoforms from 12,692 genes were included in our final, curated transcriptome.

4.3.2 Characterizing IsoAnnotLite: performance assessment on long-read transcript annotations

Recent studies by several research groups, including our own, have made a strong case for the advantages of coupling expression-level with functional analysis when it comes to fully understanding alternative splicing regulation [121, 192]. To be able to consider alternative isoform function in the interpretation of our downstream analyses (fully described in chapter 5), IsoAnnotLite was used to functionally annotate the isoforms in the curated transcriptome (see section 4.2.3). Two pre-annotated transcriptomes were used to this end: the mouse RefSeq78 annotation and an already-published, manually-annotated, long-read mouse neural transcriptome [168].

As a result of running IsoAnnotLite for our curated transcriptome, 35,028 isoforms in the GTF target annotation file (94.71% of isoforms) were annotated, meaning that at least one functional feature could be successfully transferred from the source annotation file. The remaining 696 isoforms received no functional information due to the fact that the corresponding gene was not found in the GFF3 file, however, these represented but 1.88% of the long read-defined isoforms, meaning that feature transference was effective even though the reference GFF3 annotation files used did not exactly correspond to the same mouse samples, or even the same reference transcriptome version (RefSeq96 was used as the reference annotation for transcriptome construction and quality control using PacBio ENCODE data, while tappAS' annotations were generated under RefSeq78).

Regarding the comprehensiveness of the annotation process, IsoAnnotLite reported ∼47% of transcripts in the target GTF as having the same associated reference transcript as one of the isoforms in the GFF3 reference annotations, which meant total agreement between their splice junctions and therefore ensured perfect positional transfer of features in the case of approximately half of the long read-defined isoforms. These rates could explain the high annotation rates that we observed for some of the protein-level functional features, which require CDS position matching in order to be transferred by the IsoAnnotLite algorithm. This included PFAM domains ($\sim$67% transcripts annotated with ≥ 1 domain) and post-translational modifications (PTM, $\sim$65% transcripts with ≥ 1 PTM). Feature transference in the UTRs -regions that may differ even when the reference associated transcript is the same- was successful for the 3' end, with $\sim$48% of transcripts being annotated with ≥ 1 3'UTR motif, $\sim$68% with ≥ 1 miRNA binding site and $\sim$45% with ≥ 1 repeat region. Meanwhile, 5'UTR feature annotation was only moderately successful, with a transcript annotation percentage of $\sim$2% and $\sim$21% for 5'UTR motifs and upstream Open Reading Frames (uORF), respectively. Of note, for long read-defined transcripts not matching by ID, transference depended on the differences among alternative isoforms from the same genes present in the target and reference functional annotations, which may be intrinsically larger at the 5'UTR, given the technical sources of variability affecting this region during sequencing. However, these percentages were highly dependant on the annotation coverage accomplished for the different functional databases -or feature categories- when generating the original annotations. Less well-annotated categories will therefore result in fewer transcripts with transferred features from these categories.

Taking this into consideration, we also inspected the total and relative number of features recovered from the original annotation per functional feature category (table 4.3). A complementary explanation has to do with the fact that the mouse neural transcriptome used as one of the reference annotations was considerably more condition-specific than RefSeq, decreasing the possibility to match the genes and isoforms included in the target GTF, and making transference challenging for categories extracted from it, such as RBPs and NLS.

4.3.3 Enabling multi-group differential expression of isoforms in single-cell data

To quantify the expression of the long read-defined transcripts at the single-cell level, we made use of a mouse neural short-read scRNA-seq dataset by Tasic et al. (see section 4.2.1). Importantly, this dataset was deeply sequenced (median of 4.4 million mapped reads per cell) and comprised full-length reads obtained using the Smart-seq2 library preparation method [33], which made it suitable for isoform detection. In total, 1,591 cells and 16,240 isoforms from 8,814 genes were retained after quality control (see section 4.2.8). Using the labels from the original characterization of the dataset, cells were assigned to 7 broad cell types, 5 glial (microglia, endothelial cells, oligodendrocytes, oligodendrocyte precursor cells (OPCs) and astrocytes) and 2 neural (GABA-ergic and glutamatergic neurons), each of which were divided into several, distinct subtypes (figure 4.9.a). Sensitivity for the quantified and filtered data remained high, with a median of 8,613 detected isoforms per cell, although this number was largely driven by the higher cell abundance of neural cell types, for which the number of expressed isoforms was markedly higher (figure 4.9.b). Even so, all cells from non-neural cell types presented >3,000 isoforms per cell, which surpassed the

Database or predictor	Functional category	Feature annotation level	Total features transferred	Source entries transferred (%)	Transcripts with ≥ 1 feature transferred (%)	Source annotation
COILS	COILED	Protein	1184	7.5	1.65	Neural long-reads
CORUM	Complex	Protein	197	3.65	0.53	Neural long-reads
GeneOntology	Gene Ontology	Gene	92552	9.2	83.41	RefSeq78
miRWalk	miRNA binding	Transcript	186755	23.32	68.14	RefSeq78
MOBIDB LITE	DISORDER	Protein	2732	7	3.37	Neural long-reads
NLS mapper	MOTIF	Protein	1020	7.84	1.9	Neural long-reads
NMD prediction	NMD	Transcript	268	48.82	0.72	RefSeq78
PAR-clip data	RNA binding protein sites	Transcript	3000	1.9	68.14	Neural long-reads
PFAM	CLAN	Gene	23526	18.97	63.61	RefSeq78
PFAM	DOMAIN	Protein	50736	28.19	67.32	RefSeq78
RepeatMasker	Repeat	Transcript	35075	3.51	44.97	RefSeq78
scanForMotifs	3UTRmotif	Transcript	4078	4.16	45.82	Neural long-reads
SIGNALP EUK	SIGNAL PEPTIDE	Protein	2641	85.19	7.1	Neural long-reads
TMHMM	TRANSMEMBRANE	Protein	1379	6.56	15.67	Neural long-reads
UniProtKB/PhosphoSitePlus	ACTIVE SITE	Protein	4790	93.73	8.61	Neural long-reads
UniProtKB/PhosphoSitePlus	BINDING	Protein	32126	86.31	23.76	Neural long-reads
UniProtKB/PhosphoSitePlus	COILED	Protein	3201	78.24	6.46	Neural long-reads
UniProtKB/PhosphoSitePlus	COMPBIAS	Protein	6392	72.01	11.12	Neural long-reads
UniProtKB/PhosphoSitePlus	INTRAMEMBRANE	Protein	385	111,5	0.49	Neural long-reads
UniProtKB/PhosphoSitePlus	MOTIF	Protein	15126	76.17	21	Neural long-reads
UniProtKB/PhosphoSitePlus	PTM	Protein	237128	79.76	64.5	Neural long-reads
UniProtKB/PhosphoSitePlus	TRANSMEMBRANE	Protein	19293	95.51	14.67	Neural long-reads
UTRsite	3UTRmotif	Transcript	44013	22.38	45.82	RefSeq78
UTRsite	5UTRmotif	Transcript	623	21.19	1.66	RefSeq78
UTRsite	PAS	Transcript	8787	15.39	19.73	RefSeq78
UTRsite	uORF	Transcript	13370	14.31	21.42	RefSeq78

Table 4.3: Results of IsoAnnotLite functional feature transference. *Functional database* details the original source of the functional information, i.e. sequence-based predictors or biological databases. *Feature categories* constitute broad terms under which one or more functional features encoding similar or related functions are classified. *Total features transferred* are defined as the total number of annotation entries in our long read-defined transcriptome after running IsoAnnotLite. *Source entries transferred (%)* refer to the amount of feature entries that are successfully transferred relative to the total number of features in the original annotation. *Transcripts with $\geq$ 1 features transferred* is an estimation of the final annotation coverage obtained using IsoAnnotLite.

results obtained in published single-cell lrRNA-seq studies (see section 3.6 for a thorough discussion).

Next a multi-group strategy was used to detect isoforms showing Differential Expression (DE) in at least one cell type. This decision precludes usage of pairwise comparisons between cell types while simultaneously leveraging the high amount of biological diversity captured in the scRNA-seq dataset, in which multiple biologically distinct groups can be simultaneously compared. For this purpose, the ZinBWaVE zero-expression weighting strategy [189] was combined with bulk-designed DE methods DESeq2 [191] and edgeR [190] (see section 4.2.9). Using this strategy, we intended to select isoforms with robust co-variation in anticipation for the detection of co-expression signal among isoforms, which is not possible when cell types are considered in a pairwise manner.

The Tasic dataset presented a drastic cell number imbalance between neural ($\sim$720 cells/cell type) and glial cell types ($\sim$30 cells/cell type), as per the annotation performed by the authors [163]. This resulted in the underestimation of transcriptional differences between non-neural types when analyzing the global sources of variation in the data (figure 4.9.c). To balance sample sizes, we performed 50 rounds of random neural cell sampling ($n = 45$ cells) followed by DE testing using both edgeR and DESeq2 (see section 4.2.9). Although DESeq2 proved to be slightly more robust across independent runs than edgeR, Jaccard Index values indicated that the majority of isoforms were consistently detected as DE (DESeq2: mean no. of DE isoforms = 6,908±101, mean Jaccard Index = 0.84±0.02; edgeR: mean no. of DE isoforms = 6,016±410, mean Jaccard Index = 0.74±0.02). In addition, this strategy revealed that considering the union of edgeR and DESeq2 results contributed to improve robustness (mean no. of DE isoforms in union between methods = 9,399±248, Jaccard Index = 0.82±0.01). Upon testing, we selected a consensus set of isoforms consisting in those transcripts detected as significant (FDR<0.05) by at least one DE method in ≥50% of downsampling runs. This consensus set included 9,393 isoforms from 4,223 genes, however, minor isoforms (those accumulating <10% of total gene expression) were additionally filtered, resulting in the removal of ∼10% of the consensus DE set, i.e. 969 isoforms.

4.3.4 Design of an enhanced, automated filtering strategy: the SQANTI3 ML filter

The SQANTI tool for transcriptome quality control [50] is among the most widely used tools in the long-read transcriptomics field, thanks to its pioneering isoform classification scheme 4.1 and its ability to compute a large number of quality control (QC) descriptors 4.2. When considered together, QC attributes can help users evaluate the reliability of the generated transcript models. Moreover, the first SQANTI release featured a machine learning-based filter (ML filter) that discriminated true isoforms from potential artifacts using a model trained on these descriptors (see sections 4.2.2 and 4.2.4). In spite of the convenience of this automated strategy, the original ML filter was limited regarding flexibility and user-friendliness, and not prepared for the removal of artifacts from known transcript categories, namely FSM and ISM (section 4.3.1). Later, the release of SQANTI2 and subsequently

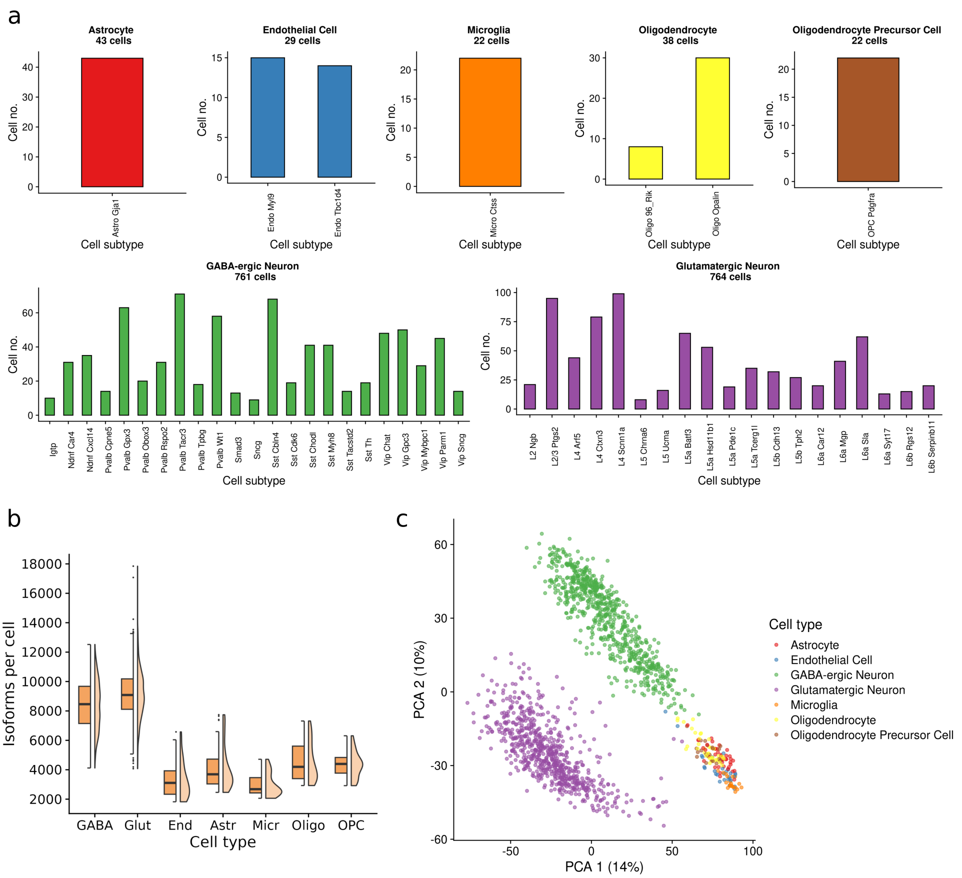


Figure 4.9: Overview of the Tasic mouse neural dataset. a Number of cells assigned to each cell type and subtype by authors in the original manuscript (post-QC, n = 1591 cells). **b** Number of isoforms per cell (x-axis) for each cell type in the Tasic dataset. Median and quantiles are depicted using a boxplot (left), general distributions are represented by a violin plot (density distribution, right). GABA: GABA-ergic neurons, Glut: glutamatergic neurons, End: entothelial cells, Astr: astrocytes, Micr: microglia, Oligo: oligodendrocytes, OPC: oligodendrocyte precursor cells. **c** Principal Component Analysis (PCA) of cell level isoform counts. PC1 (x-axis) and PC2 (y-axis) are shown.

SQANTI3, bot of which allowed the integration of additional validation data and thus the computation of novel QC attributes (table 4.2), caused the ML filter to become outdated. Notably, most of these new variables were related to TSS and TTS validation, requiring the definition of *ad hoc* thresholds in order to consider these aspects of transcript quality during artifact detection, as shown in section 4.3.1. Although time-consuming, this strategy proved effective for the removal of FSM and ISM false positive isoforms, and we set out to automate the task by refactoring and expanding of the ML filter to create a new Filter module for the SQANTI3 software. As a result, the SQANTI3 ML filter now allows the leveraging of new QC attributes, as well as the adaptation of the filter to each user's needs thanks to the implementation of new parameters to enhance its flexibility. For a full description of SQANTI3 ML filter improvements, see section 4.2.4.

To make results comparable, we run the SQANTI3 ML filter on the collapsed transcriptome and defined custom TP and TN sets mimicking the criteria used during our manual curation (see section 4.3.1). Specifically, FSM transcripts including a CAGE-supported TSS and polyA motif-supported TTS were used as the TP set, whereas a a set of transcripts from all the remaining structural categories lacking support at the TSS, TTS or at least one SJ (see section 4.2.5) were used as TN. As a result, out of the 93,698 transcripts included in the transcriptome after collapse, 43,348 isoforms from 21,941 genes passed the filter (46%, isoform probability > 0.7). Similarly to the results observed after manual curation (figure 4.6.a), the filtered tran-

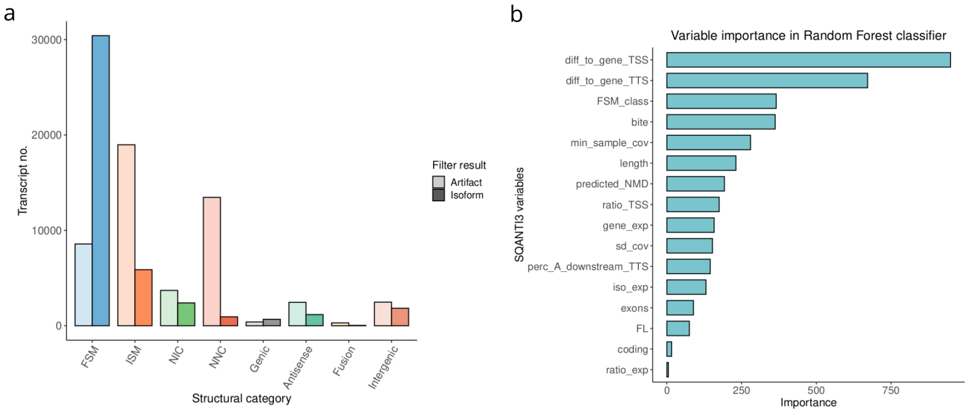


Figure 4.10: SQANTI3 ML filter results. **a** Artifacts and isoforms flagged by the ML filter for each SQANTI3 structural category. FSM: full-splice match, ISM: incomplete-splice match, NIC: novel in catalog, NNC: novel not in catalog. **b** Variable importance in random forest classifier. Importance values are dimensionless and arbitrarily established by the random forest algorithm, however, they are valid for within-filter analyses of the trained classifier model.

scriptome contained a larger proportion of FSM isoforms, whereas the ISM and NNC categories were the most enriched in artifacts (figure 4.10.a).

Detailed characterization of SQANTI3's ML filter results

When examining the results obtained using the trained random forest classifier, SQANTI3 QC attributes associated with the distances to an already-annotated TSS or TTS were found to be particularly relevant for artifact/isoform classification (figure 4.10.b). When exploring the distribution of values for these variables across isoforms and artifacts, TSS/TTS distances were found to have discriminant power for filter-passing FSM and ISM, with the latter showing especially high distance values for both sites (figure 4.11.a). These effects were associated with the ML filter handling of different ISM subcategories (figure 4.11.b). Artifacts from the 3' fragment subcategory (i.e. ISM skipping 5' exons, figure 4.1.b) showed markedly larger distances to the gene TSS than isoforms, whereas the strong gap in TTS distance was found to be associated to 5' fragment artifacts (figure 4.11.b). Internal fragments, as expected given their exon structure, followed both patterns. ISM filtering therefore varied largely depending on the source and structure of the transcript, however, more tolerance towards large 3' end differences, together with more stringent filtering of large 5' end variability, was observed when inspecting the density distributions of TSS/TTS distances (violin plots, right side, figures 4.11.a-b). Regarding FSM, although subcategory-level analyses revealed subtler differences between artifacts and isoforms than those observed for ISM (figure 4.11.c), these followed the same filtering pattern; namely, larger variability respective to the reference was accepted for the TTS than for the TSS. This was also true for NIC and NNC isoforms. Considering that FANTOM database CAGE data was used for QC and for TP set definition, these results suggested that the usage of non sample-specific CAGE data for 5' end validation could cause filter-passing isoforms to be more similar to the reference.

To verify this, we investigated the TSS ratio values of FSM isoforms and artifacts. The TSS ratio is computed using short-read coverage upstream and downstream of the transcript TSS (table 4.2). Degraded transcripts are expected to display uniform coverage on both sides of the TSS (TSS ratio$\approx$1), whereas true TSS are expected to have much lower upstream coverage (TSS ratio$>>$1). Since CAGE peaks and polyA motifs were used to define the TP set, all related variables were removed before classifier

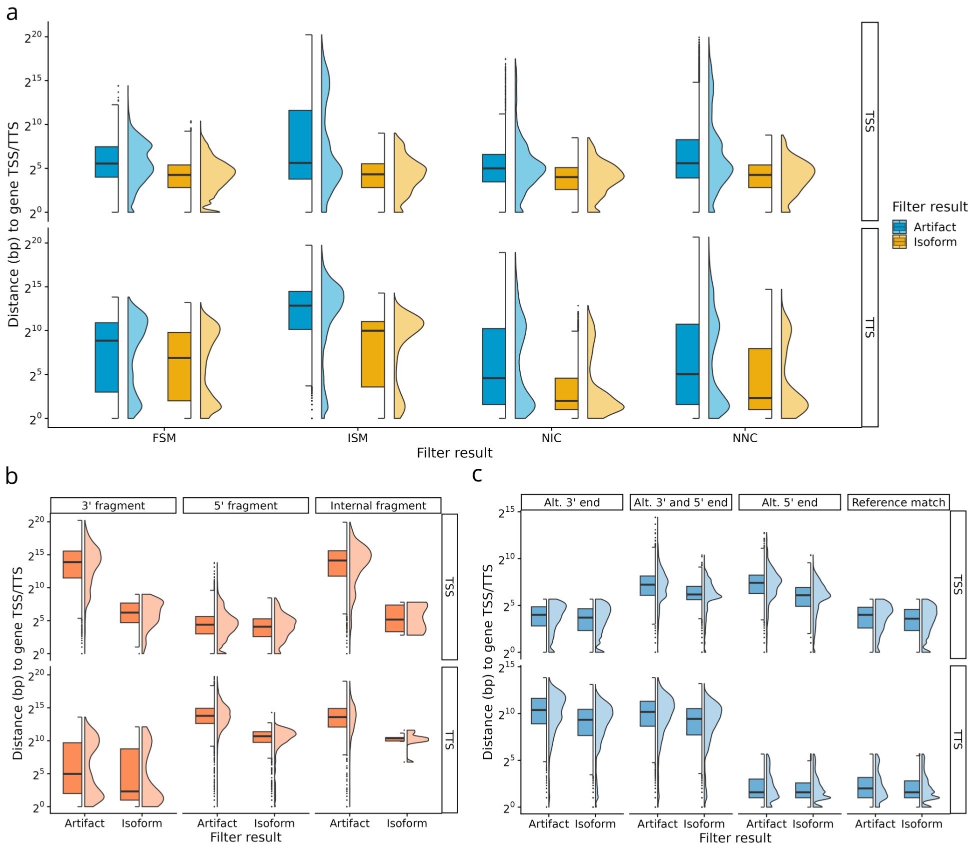


Figure 4.11: Distance to gene TSS/TTS value distribution for multi-exon artifacts and isoforms across **a** main structural categories, **b** ISM subcategories and **c** FSM subcategories. Distance values (bp) are converted using a Log2 transformation. Distributions are represented by median and quantiles as boxplots (left half) as well as density distributions in the form of violin plots (right half). FSM: full-splice match, ISM: incomplete-splice match, NIC: novel in catalog, NNC: novel not in catalog.

training, leaving the TSS ratio as the only data-based attribute used for TSS validation. After classifier training, the TSS ratio only proved to have medium importance for classification using the random forest model (figure 4.10.b) and only slightly higher ratios were found when comparing isoforms and artifacts for FSM and ISM (figure 4.12.a). Reassuringly, high levels of agreement among ratio TSS, CAGE and the reference annotation were found when inspecting the TSS support of filter-passing FSM and ISM (figure 4.12.b), suggesting that the stringent 5' filtering pattern described above originated from a combination of the various data sources used upon QC, and not only from similarities to the reference annotation. Moreover, only 4,011 out of 31,093 filter-passing transcripts from these two categories (3,878 FSM and 133 ISM) had an unverified TSS, and were likely classified as isoforms using other criteria.

Conversely, the random forest model was solely trained on sequence-level TTS validation attributes, specifically polyA motif detection (see section 4.2.5). Differences in distance to the detected polyA motif between isoforms and artifacts were found to be larger for ISM than for FSM, with the former showing a stronger signal peak at the expected polyA motif site, i.e. 18bp upstream of the TTS (figure 4.12.c, top panel). ML results partly recapitulated TP set properties, with the detection of a polyA motif next to the TTS being more frequent for filter-passing FSM and ISM transcripts than for artifacts (figure 4.12.c, bottom panel). Agreement between polyA motif detection and reference annotation similarity, however, was mild in comparison to TSS data sources (figure 4.12.d), indicating that polyA motif and TTS annotation constitute complementary supporting data and thus, as reported above, allow more lenient filtering on TTS variation.

Distance attributes were followed in importance by variables related to junction novelty and coverage, i.e. bite and minimum sample coverage, respectively (see table 4.2 for definitions). The bite variable, computed only for novel junctions, was shown to be positive for most NNC artifacts (figure 4.13.a). Meanwhile, SJ short-read coverage was markedly higher for ML-reported isoforms in the cases of the NIC and NNC categories (figure 4.13.b), indicating its relevance for junction validation. Validated FSM also exhibited SJ coverage more frequently than artifacts from the same category, which may explain the classification of some FSM as isoforms regardless of TSS/TTS validation. Indeed, ~79% of the filter-passing FSM lacking TSS support when using thresholds (figure 4.12.b) were positive for SJ coverage (minimum sample coverage = 1). This showcases the ML filter's ability to aggregate different QC attributes and generate complex filtering criteria, as opposed to rule-based curation.

Regardless the robustness of these results, we anticipate that the usage of short and long-read data from slightly different biological sources (see section 4.2.1) has in all likelihood played a part in the stringency of the ML filtering process, leading to generally low TSS ratio values and contradicting published SQANTI3 performance descriptions for known transcript categories [162]. Novel TSS and TTS are also expected to remain unverified in the absence of sample-specific supporting data, e.g. CAGE-seq [193] or Quant-seq [186] (see [162]). Conversely, inflicting severe SJ coverage requirements on NIC

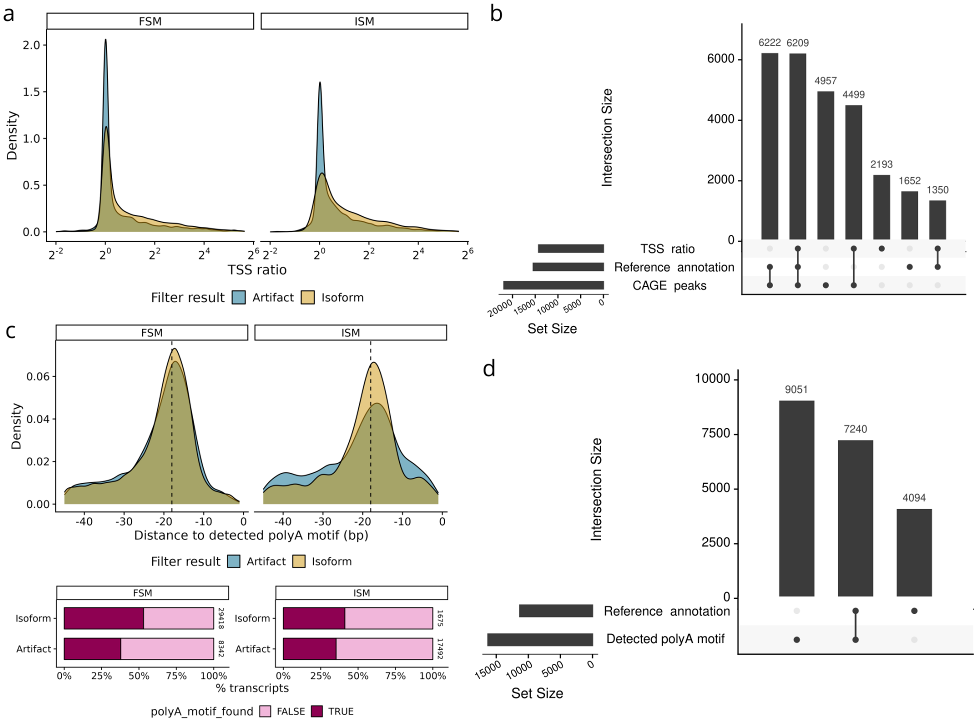


Figure 4.12: Agreement between ML filter results and TSS/TTS validation data sources. a TSS ratio (log2 scale) density distribution for multi-exon isoforms and artifacts from the FSM and ISM structural categories. **b** Upset plot showing the intersection between TSS validation sources. Filter-passing FSM and ISM are interrogated for TSS validation based on different QC attributes, TSS ratio (TSS ratio ≥ 1.5), reference annotation (`diff_to_gene_TSS` $\geq$ 50bp upstream start site) and FANTOM CAGE peak data (`within_CAGE_peak = TRUE`). **c** Density distribution of distances to detected polyA motif for multi-exon isoforms and artifacts from the FSM and ISM structural categories (top panel) and percentage of FSM isoforms and artifacts for which a polyA motif was found (bottom panel). In the latter, labels show the total number of transcripts aggregated in each bar. **d** Upset plot showing the intersection between TTS validation sources. Filter-passing FSM and ISM are interrogated for TTS validation based on polyA motif detection (`polyA_motif_found = TRUE`) and reference annotation (`diff_to_gene_TTS` $\geq$ 50bp downstream end site). For upset plots, dots indicate intersections (x-axis), bar height (y-axis) indicate intersection sizes and horizontal bars indicate the number of transcripts for which the site was validated using each source. FSM: full-splice match, ISM: incomplete-splice match. TSS: transcription start site.

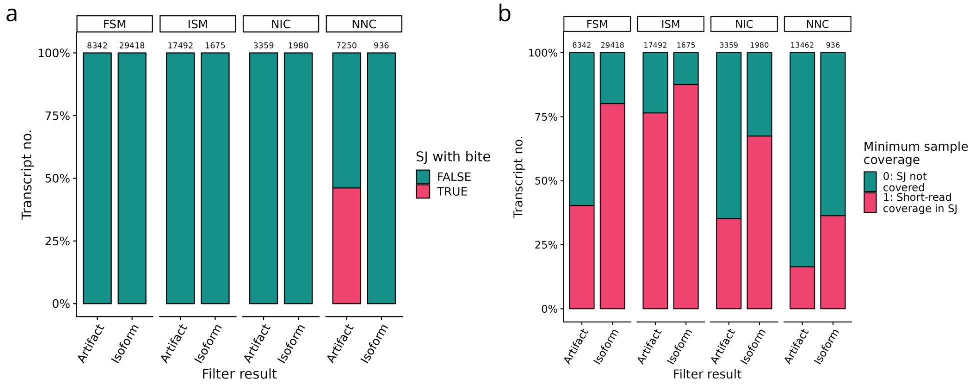


Figure 4.13: Junction-related quality attributes used by random forest classifier: a minimum sample coverage and **b** bite. The percentage of multi-exon isoforms and artifacts assigned to each level is shown for the main structural categories. FSM: full-splice match, ISM: incomplete-splice match, NIC: novel in catalog, NNC: novel not in catalog. SJ: splice junction.

and NNC transcripts may lead to the preservation of a reduced, but more reliable set of novel isoforms, as single-cell reads mapping to novel junctions will serve as proof of sample-specific expression. Although this can mitigate errors caused by the lack of matching short-read data for the selected mouse neural ENCODE dataset, it can also operate to the detriment of the amount of novelty that can be confidently introduced in the transcriptome, limiting the discovery potential of downstream analyses.

Comparing transcript filter outcomes: SQANTI3 ML filter vs. manual curation

To better understand filter performance, SQANTI3 ML filter results were compared to those previously obtained using manual curation. First, the level of agreement in artifact detection was measured across the four main structural categories (figure 4.14). SQANTI3 largely recapitulated filter re-

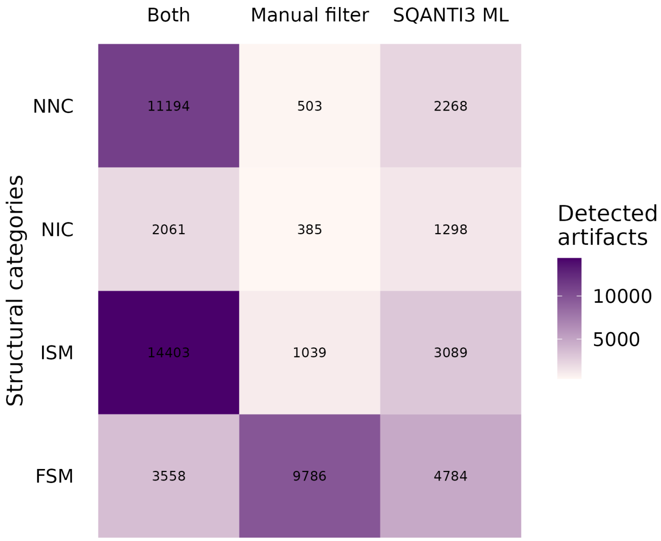


Figure 4.14: Agreement and discrepancies in detected artifacts when using manual curation vs the SQANTI3 ML filter. *Manual curation* includes the application of the original SQANTI ML filter for the removal of novel transcripts and threshold-based filtering of known transcripts based on TSS/TTS diversity. The intersection of artifact IDs between filters, as well as the number of unique artifact IDs for each of the filters, are represented by heatmap color. FSM: full-splice match, ISM: incomplete-splice match, NIC: novel in catalog, NNC: novel not in catalog.

sults for novel transcript categories. For NNC, only 16.2% of transcripts flagged as artifacts by SQANTI3 were classified as isoforms by the original filter, whereas this number increased to 34.7% for the NIC category. In the case of known transcript categories, the manual curation strategy removed a significantly larger proportion of FSM than the new ML filter (figure 4.14). Conversely, both filters identified the same ISM as artifacts, with only 5.6% and 16.7% being uniquely removed by the manual and SQANTI3 filters, respectively. Category-level differences in artifact detection, however, are helpful to better understand the behaviour of the SQANTI3 ML filter.

First, we compared the properties of NIC transcripts removed by each filtering strategy. Since novel categories were not considered during manual curation, discrepancies found in novel transcript filtering only correspond to the results of the original SQANTI ML filter, that is, a classifier model that did not consider the TSS/TTS attributes introduced in SQANTI2 and SQANTI3 (see section 4.3.1 and figure 4.6.a). Characterization of artifacts from the NIC therefore allowed us to gain insight into the differential properties of the original and the improved ML filters. Regarding start and end site diversity, NIC transcripts uniquely removed by the SQANTI3 ML filter exhibited larger differences to the annotation than those identified only by SQANTI, regardless of orthogonal data support (figure 4.15.a-b). The effect of 3' differences was particularly stringent, with $>$50% of SQANTI3-unique NIC artifacts being excluded despite presenting a polyA motif near the TTS (figure 4.15.c). CAGE data was similarly disregarded, and the proportion of filtered transcripts whose TSS was situated within-peak was close to that of SQANTI ML (figure 4.15.d), that is, to a model that was blind to TSS support data. Finally, while SQANTI ML and common artifacts had consistently low SJ coverage, those uniquely flagged by SQANTI3 often presented supported SJ, showing a bimodal distribution (figure 4.15.e). We additionally found an association between the removal of short read-supported NIC and end-level divergence from the reference annotation, particularly for the TTS (figure 4.15.f), confirming that SJ validation criteria were insufficient to pass the ML filter when low-quality end sites were present in the transcript model. All in all, these results suggest that the inclusion of TSS and TTS supporting data in SQANTI3 increases the stringency of novel transcript filtering, since junction-level quality can be deemed insufficient for ML filter validation.

A similar evaluation was performed for the properties of artifacts from the FSM category. Differences in this case can be attributed to threshold-based filtering according to CAGE, polyA and reference annotation support, since our manual curation strategy did not consider junction-level evidence for the removal of transcripts from known categories (see section 4.2.2). Taking this into consideration, we set out to find differences between manual and SQANTI3-unique FSM artifacts across these three TSS and TTS validation sources (figures 4.16.a-b).

FSM artifacts detected solely using manual curation showed markedly larger distances to the TSS than those inferred using SQANTI3, especially for those transcripts for which a polyA motif had been detected near the TTS (figure 4.16.a). Since threshold-based filtering imposed the requirement that both the 3' and 5' ends of the transcripts were validated using at least one type of evidence (see section 4.2.2), the divergence between these two metrics is to be expected. Namely, this effect corresponds to transcripts with reference-supported TSS for which the TTS could not be similarly verified. Conversely, manually-detected artifacts lacking a polyA motif showed a much more similar TSS distance distribution to that of transcript discarded by SQANTI3 (figure 5.16.a). The opposite pattern was observed for this set of artifacts in the case of the TTS, with CAGE-supported transcripts being discarded due to large differences with the annotated site and the lack of a polyA motif (figure 5.16.b), a finding that was not replicated by SQANTI3-unique

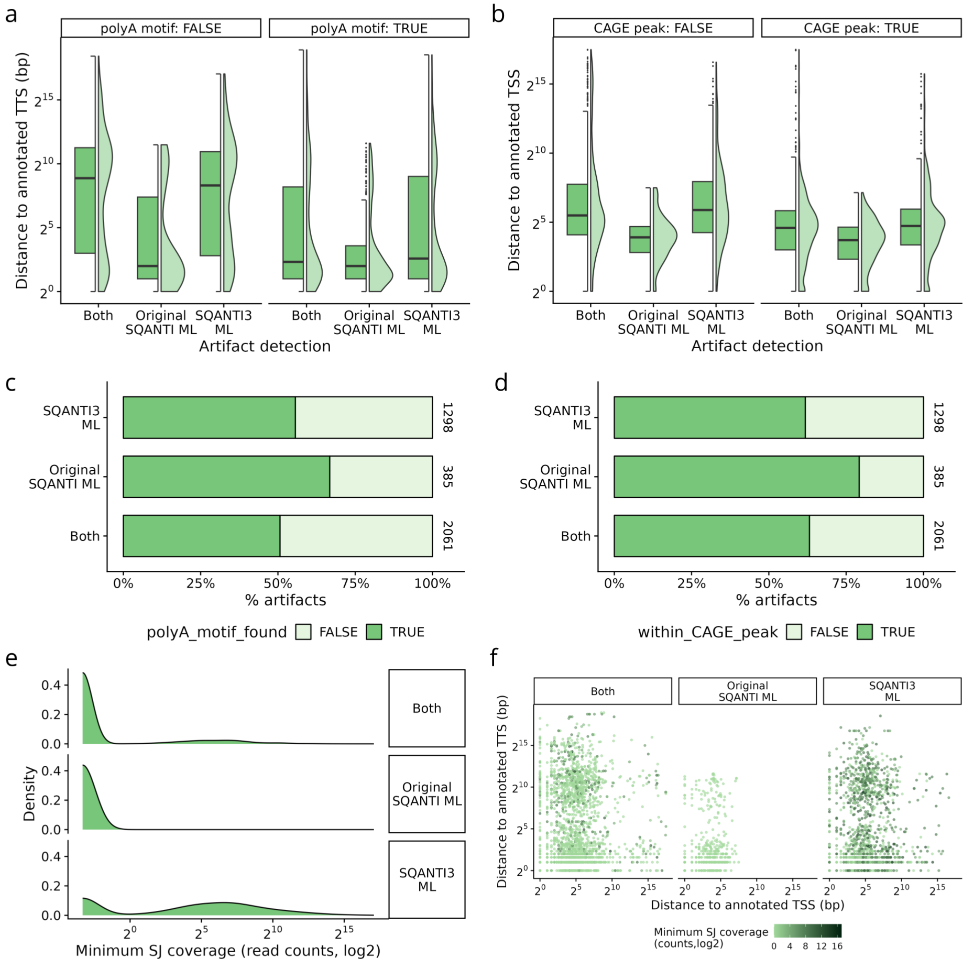


Figure 4.15: Characterization of novel in catalog (NIC) transcripts classified as artifacts by the original SQANTI and SQANTI3 ML filters. Values are computed for artifacts detected uniquely by individual filters, as well as by both. **a** Distances to annotated TSS for artifact transcripts, stratified FANTOM CAGE peak detection. **b** Distances to annotated TTS for artifact transcripts, stratified by polyA motif detection. Both distance variables are shown on a log2 scale. Distributions are represented by median and quantiles as boxplots (left half) and density distributions as violin plots (right half). **c** Percentage of artifacts for which a polyA motif was detected near the TTS (≤50bp uspream). **d** Percentage of artifacts for which the TSS was found within a FANTOM CAGE peak. **e** Density distribution of minimum splice-junction (SJ) coverage. Values are supplied as read counts and converted using a Log2 transformation. **f** Distance to annotated TSS (x-axis) and TTS (y-axis) for unique and common artifacts. Dot color corresponds to minimum junction coverage (log2 read counts).

artifacts. Moreover, this remarkable increase in TTS distance for manually-filtered transcripts was only mitigated when accompanied by the detection of a polyA motif and the lack of CAGE support (figure 5.16.b), a complementary situation to that described before. Notably, the elimination of transcripts due to lack of polyA motif validation in spite of having a CAGE supported-TSS was much more frequent than any other possible data combination scenario (figure 5.16.c). This could be attributed to the unspecificity of polyA motifs as a validation method, making a case for the usage of other 3' data sources (e.g. Quant-seq [186]). In addition, this observation is aligned with the TTS filtering stringency that was previously observed. Data-driven validation of start and end sites therefore succeeded in mitigating redundancy at the cost of becoming exceedingly demanding towards FSM. Meanwhile, the automated strategy employed by the SQANTI3 ML filter -at least with the provided TP set- favored the preservation of known transcripts rather than the exact definition of their TSS and TTS.

SQANTI3, however, also removed a large number of FSM that were classified as isoforms using threshold-based filtering (figure 4.14), 65.2% of which showed deficiencies in SJ coverage. This constitutes a much higher proportion than that of common and manual curation unique artifacts (figure 4.16.d). Arguably, however, reference similarity could be considered a sufficiently reliable criteria for SJ validation in known categories, especially considering that the short and long-reads datasets used did not come from the same study (see section 4.2.1). On the other hand, the different biological properties of bulk vs single-cell data may call for the establishment of sample-specific expression as a relevant filtering criteria. Usage of a trained model can therefore act to the detriment of aspects that would be more controlled using a rules-based strategy, although it holds potential for the detection of biases intrinsic to the data that would otherwise not be accounted for.

4.3.5 Enhancing transcriptome rescue in SQANTI3: a comprehensive approach

The curation process, performed either manually or automatically, aimed at eliminating unreliable and low-quality isoforms. This, however, occasionally resulted in the exclusion of entire genes from the transcriptome, despite the presence of evidence supporting the expression of these genes in the lrRNA-seq data (see section 4.3.5). To address this challenge, we initially devised a soft rescue strategy to recover known genes and transcripts in situations where no long-read isoforms could be validated, resulting in the removal of all transcripts associated with a given known reference transcript. Notably, this rescue operation benefited from SQANTI associations between FSM and the reference annotation, and thus demanded the presence of at least one FSM isoform per reference transcript. This assignment therefore relied solely on the assessment of junction similarity between the isoform and the reference. Consequently, this initial rescue approach was limited in scope, precluding the recovery of information that may have been lost due to the removal of ISM, NIC and NNC transcripts. Recognizing the need for a more systematic and comprehensive approach, we embarked on the development of an enhanced rescue strategy, characterized by the identification of suitable replacements for all filtered transcripts, transcending the constraints of

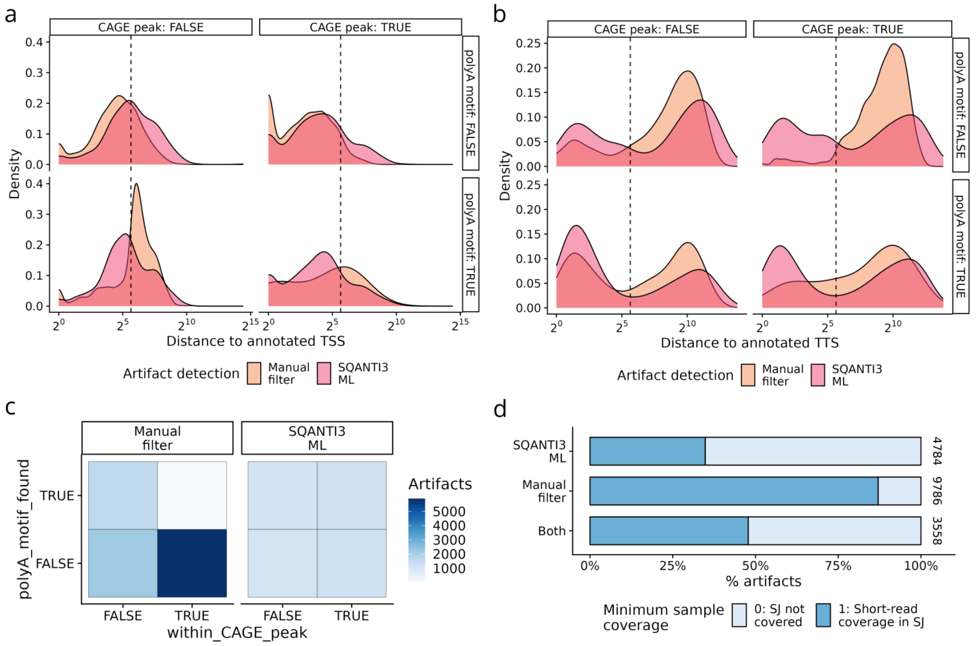


Figure 4.16: Characterization of full-splice match (FSM) transcripts classified as artifacts using manual curation vs the SQANTI3 ML filter. Values are computed for artifacts detected uniquely by individual filters (and the intersection of both filters, where applicable). Density distributions of distances to the annotated **a** TSS and **b** TTS, stratified by detection of polyA motifs and CAGE peaks. For TSS, only transcripts with distance values downstream of the annotated TSS are considered to allow comparison with the rules set during manual curation. Distance values (bp) are converted using a Log2 transformation. The vertical line indicates the 50bp threshold used in manual curation. **c** Artifact transcripts for which a polyA motif was detected near the TTS ($\leq$50bp upstream) and/or had a TSS situated within a FANTOM CAGE peak. Heatmap color indicates the number of artifacts found for each validation data combination. **d** Percentage of artifacts showing (minimum sample coverage = 1) or lacking (minimum sample coverage = 0) splice junction (SJ) coverage. Labels indicate the total number of artifacts represented by each bar.

exclusively FSM-supported genes and transcripts. This novel rescue module, which was integrated within the SQANTI3 framework, is fully described in section 4.2.6.

In total, 8,965 reference transcripts were incorporated into the transcriptome after running SQANTI3 Rescue (figure 4.17.a). This process encompassed the rescue of 17,226 long-read artifacts, meaning that a suitable replacement was found for 34.1% of transcript models removed by the SQANTI3 ML filter (50,350 total artifacts). Among this set of artifacts, however, 32,254 met the necessary requirements for rescue (see section 4.2.6), increasing the percentage of successful rescue events to 53.4%. Importantly, the application of SQANTI3 Rescue increased the number of reference transcripts represented by isoforms in the transcriptome to pre-ML filter levels (figure 4.17.a). Since this directly corresponds to unique junction chains encoded in FSM, ISM and rescued references, the former result can be interpreted as a measure of the large levels of isoform diversity in the final transcriptome. The introduction of rescued references was additionally accompanied by the recovery of 2,560 genes that had been completely removed by the ML filter (figure 4.17.a), successfully leveraging long-read data evidence to mitigate gene loss.

After rescue, we observed the number of artifacts for which a replacement was found to be significantly larger than the number of unique reference transcripts selected for rescue. This, however, is to be expected, as the independent processing of discarded transcripts during rescue often results in the selection of the same replacement transcript for several artifacts (see section 4.2.6). In fact, on average, we found that each rescued reference transcript

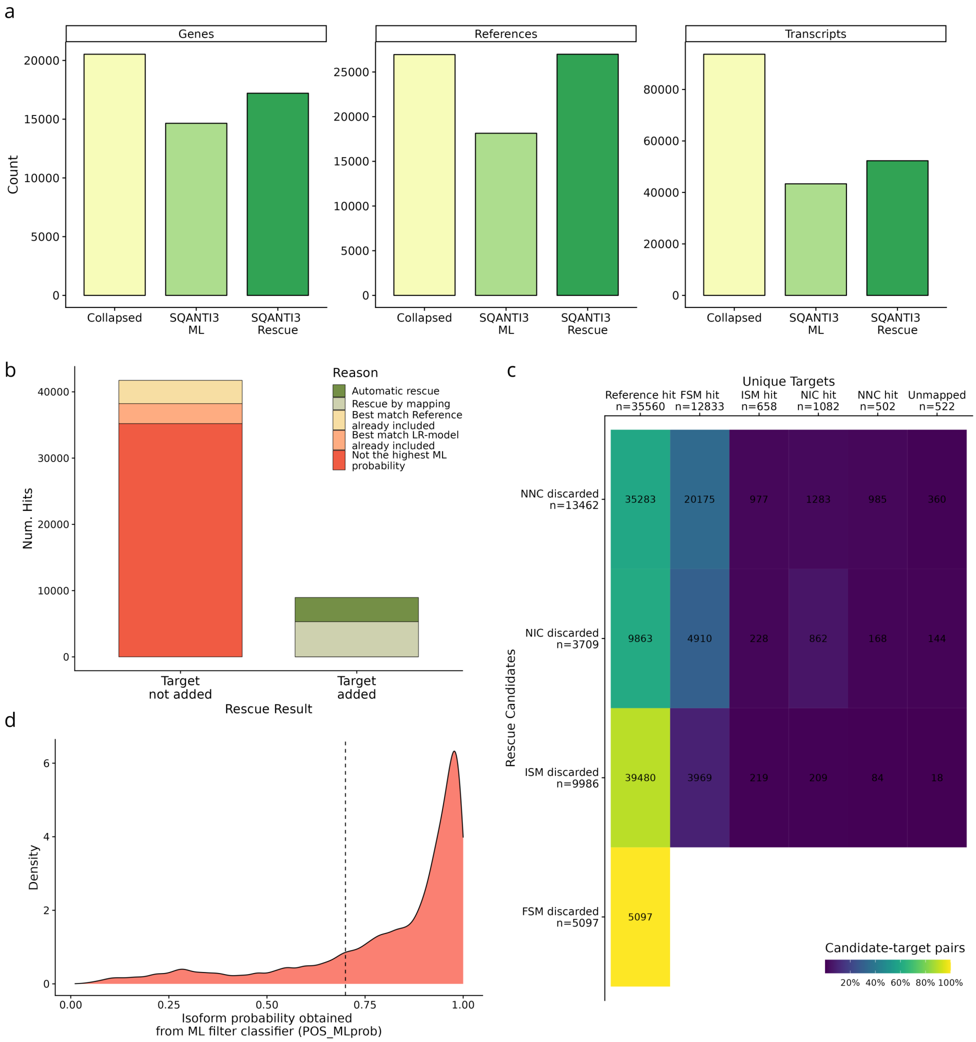


Figure 4.17: SQANTI3 Rescue results. **a** Total genes, unique references and transcripts represented in the final rescued transcriptome. **b** Number of rescue targets (y-axis) and whether or not the target was added into the transcriptome (x-axis). This includes all candidate-target pairs selected during mapping. Targets are flagged by the final reason for which they were include or excluded. **c** Relationship between structural categories of discarded transcripts (rescue candidates, y-axis) and the rescue targets to which they were associated during mapping (x-axis). Heatmap color corresponds to the proportion of mapping hits from a candidate category that represent each pairwise association. The label indicates the total number of candidate-target pairs represented. **d** Density distribution of positive ML probabilities for target reference transcripts, obtained using the pre-trained random forest classifier.

was matched by 2.4 artifacts. This rate, although moderate, suggests that multiple false positive variants can be generated from same true transcript.

In a similar spirit to that of our initial soft rescue approach, SQANTI3 Rescue first retrieves reference transcripts with FSM-based evidence for which all associated isoforms had been removed by the ML filter, a process known as automatic rescue (section 4.2.6). For our data, 40.8% of rescued transcripts were recovered using this process (figure 4.17.b), which entailed the substitution of 6,043 FSM artifacts by their originally assigned reference transcript. Next, artifacts from the remaining structural categories (rescue candidates) are mapped against both reference and long-read defined isoforms (rescue targets) to find structurally related, high-quality transcripts (section 4.2.6). In this process, multiple alignments are enabled to allow for a broad search of the best match for each rescue candidate, which results in the selection of multiple targets per candidate. To better understand this process, we evaluated the associations formed between rescue candidates and targets depending on their structural category (figure 4.17.c). Among them, ISM candidates showed the largest preference for reference transcripts, with only 10.2% of candidate-target mapping associations involving a non-reference isoforms. Even though discarded NNC and NIC also showed high reference hit frequencies, they also formed a large number of mapping associations with FSM targets. The prevalence of low mapping rates to long-read models for all candidate categories aligns with our previous observation that most rescued references were retrieved via mapping (figure 4.17.b). This result suggests that automatic rescue alone is insufficient to make use of all relevant lrRNA-seq evidence of transcriptome complexity, highlighting the importance of considering non-FSM artifacts and the power of mapping to find adequate replacement transcripts.

Importantly, all reference targets are validated using the trained random forest classifier used during the application of the SQANTI3 ML filter (section 4.2.7). While most of these transcripts had reportedly high true isoform probabilities (figure 4.17.d), not all filter-passing references were introduced in the final transcriptome (figure 4.17.b), as selected targets are evaluated to avoid the introduction of redundant isoforms (section 4.2.6). Out of the 35,561 unique reference targets mapped by at least one artifact, only 25.2% were incorporated into the final transcriptome, which added up to 8,965 reference transcripts, as reported above (figures 4.17.a-b). Some targets selected during mapping did not make it into the final transcriptome to avoid introducing unwanted redundancy, either because they corresponded to validated long-read isoforms or to already-represented reference models. These, however, represented a small proportion, meaning that most of the targets that were not eventually added had a same-candidate, filter-passing counterpart with a higher probability of being a true isoform (figure 4.17.b). The small amount of potentially redundant rescue events serves as proof that stringent artifact filtering can entail the loss of unique information, stressing the importance of including a rescue step in transcriptome curation pipelines.

Conversely, no replacement transcript was added for 15,028 (46.6%) rescue candidates, due to none of the associated targets fulfilling the requirements for rescue. Most of these artifacts (65%) belonged to the NNC category, which is likely related to challenges associated to the analysis of junction-level novelty. Among unsuccessful rescue events, 74.1% of candidate-target associations established during mapping were discarded due to the target's low ML filter probability. Specifically, the validation of targets using the ML filter avoided the introduction of ∼15K transcripts in the final transcriptome, which may have been of low quality, in the case of long-read models, or non sample-specific, in the case of reference targets. This finding, along with the results presented in this section, serve as evidence of SQANTI3 Rescue's effectiveness in preserving transcriptome complexity while also reinforcing filtering decisions that prevent the inclusion of unreliable isoforms.

4.4 Discussion

Long-read RNA sequencing (lrRNA-seq) offers the potential to elucidate transcript isoform diversity at unprecedented levels, providing a rich catalog of novel and sample-specific isoforms that expand and complement reference annotations [52]. However, the technology falls short in capturing the full spectrum of isoform diversity when applied to single cells [37]. To address this limitation, we have developed and successfully implemented a hybrid approach, which harnesses the strengths of both bulk long-read and single-cell short-read sequencing. The pipeline presented in this chapter constitutes a creative solution to bridge the gap between the comprehensiveness of bulk lrRNA-seq and the cell-level resolution of scRNA-seq, and successfully enhances the sensitivity of single-cell isoform analyses to a higher level than that achieved in previous studies (see [55, 78, 149, 150] and section 4.3.3).

In the course of our study, we recognized a critical need for comprehensive quality control to mitigate the inclusion of erroneous isoforms within the long-read transcriptome. These issues had been previously highlighted by the work of others within the long-read community [50] and addressed in the form of dedicated software tools, among which SQANTI constitutes the most widely-used example. The process of generating the bulk transcriptome presented in this chapter unveiled the inadequacy of available filtering strategies, most notably the SQANTI ML filter, which had previously demonstrated effectiveness in eliminating junction-level artifacts associated with novel in catalog (NIC) and novel not in catalog (NNC) transcript categories [50]. However, the SQANTI ML filter fell short when confronted with the task of detecting spurious novel transcription start sites (TSS) and transcription termination sites (TTS). To address this shortcoming, we leveraged the integration of supplementary data sources in the SQANTI2 and SQANTI3 software releases, as well as the formulation of a new set of stringent criteria for filtering full-splice match (FSM) and incomplete-splice match (ISM) transcripts. Although threshold-based and thus time-consuming, this filtering procedure proved particularly effective for the removal of fragments and degradation products within the ISM category (see section 4.3.1), and resulted in the introduction of two pioneering modules within the latest iteration of the SQANTI framework, SQANTI3: the Filter and Rescue modules.

As a result of this work, the SQANTI3 Filter module [162] has evolved to encompass an expanded version of the original SQANTI ML filter. This advanced filter can now incorporate TSS and TTS-related data sources supplied when running SQANTI3 QC, allowing it to automatically filter not only NIC and NNC artifacts, but also FSM and ISM transcripts with unreliable start and end sites. This enhanced capability ensures a more comprehensive and precise filtering of transcripts, addressing the intricacies of TSS and TTS-related artifacts, which are often challenging to manage when no orthogonal data is brought into the process. Additionally, the SQANTI3 Rescue module [162] has been introduced to ensure the completeness of the long-read transcriptome, ensuring that no relevant gene and transcript-level diversity is inadvertently lost due to stringent filtering. This dual approach, involving both the Filter and Rescue modules, ensures that the community benefits from more accurate and comprehensive transcriptome curation tools in the context of long-read sequencing studies, be they single-cell or bulk (see sections 4.3.4 and 4.3.5). Moreover, the curated isoforms were additionally annotated with functional domains, motifs and sites (see section 4.3.2). Although introduced by others for bulk transcriptomics [121], the application of this functional annotation strategy is pioneering in the single-cell field, allowing the coupling of single-cell isoform expression results with an interpretable, biological readout (discussed in Chapter 5). Thus, our efforts in the development of novel strategies within the SQANTI3 framework represent a significant advancement in addressing the inherent challenges associated with long read-based transcriptome generation.

Understanding the properties of low and high-quality transcripts, particularly when using a machine learning-based filter such as the ones implemented in SQANTI [50] and SQANTI3 [162], can be very challenging. By conducting an extensive evaluation of automated and manual filtering results (sections 4.3.1 and 4.3.4), we have hereby contributed to the enrichment of the transcriptome curation road-map, providing hallmarks for the establishment of filtering rules as well as evidence-based insights regarding the reasons behind random forest model classification decisions. Importantly, the definition of the TP set, as stated throughout this study, is key to the generation of a model that yields an adequate performance. While it is difficult to control for the way in which variables are integrated in the random forest model, the comparison to manual curation results leaves several important take-home messages. First, the integration of multiple data sources per transcript feature (TSS, TTS, SJ) allows for a more precise model. Omitting variables to prevent overfitting, although required, can sometimes result in the removal of transcripts that would be considered reliable in threshold-based filtering. When various types of data are available, however, each supplied data source can partially recapitulate the rest, permitting the usage of some attributes to define the TP set without risking the loss of validation information, as demonstrated in [162]. In the present study, this is exemplified by the removal of a large number of transcripts due to large differences to the annotated TTS, which was not observed for the TSS given the existence of three related attributes: the TSS ratio, FANTOM CAGE peaks, and the reference annotation. Secondly, it should be noted that performance under conditions of imperfect data matching can yield unstable filtering results, as was observed for FSM in the present study. The usage of sample-specific short-reads would likely increase the discriminant power of the TSS ratio metric, as would the usage of a set of CAGE peaks or TSS sites obtained from matching samples. Finally, threshold-based curation seems to be more advisable in cases where the relationship and level of agreement between data sources (i.e. transcriptome reconstruction and validation datasets) is unknown, whereas the SQANTI3 ML filter shows improved performance in more controlled settings [162]. Even so, the update and automation of the SQANTI3 Filter module constitutes a timely development, especially given the current relevance of long-read transcriptomics in the bioinformatics community [194] and the publication of several benchmark studies [195, 196] that highlight the discrepancies across transcriptome reconstruction methods.

The evaluation of the new rescue module within ML-filtered data provided valuable insights into the effectiveness of this approach, extending beyond the retrieval of FSM-supported reference transcripts. Notably, the strategy demonstrated its capacity to successfully reclaim lost reference transcripts and genes, augmenting transcriptome complexity and finding high-quality substitute transcripts for approximately half of the rescue candidate artifacts. Remarkably, a substantial proportion of the rescued transcripts were identified during the mapping process, underscoring the utility of this newly introduced step for identifying suitable replacement transcripts, as well as the significance of considering non-FSM artifacts. While some rescue targets identified during mapping were omitted from the final transcriptome to avoid introducing redundant information, this accounted for a minority, as most unincorporated targets were rejected in favor of a filter-passing counterpart with a higher likelihood of being a genuine isoform. The fact that the exclusion of rescue targets is rarely done on the basis of redundancy, together with the small average multiplicity rate of artifacts over targets, underscores the potential loss of unique information that comes with the removal of each artifact, emphasizing the need to incorporate a rescue step into transcriptome curation pipelines. Furthermore, the decision to validate targets using the ML filter before their inclusion in the transcriptome was effective in preventing the recovery of transcripts of inferior quality or non sample-specific. These findings offer compelling evidence of the SQANTI3 Rescue's proficiency in preserving transcriptome complexity while reinforcing filtering decisions to exclude unreliable isoforms. All in all, the innovations in long-read transcriptomics methods presented in this chapter situate SQANTI3 at the forefront of its field, bridging the gap between lrRNA-seq and the generation of high-quality, reliable transcriptomes that are suitable for biology-aware analysis and interpretation.

Lastly, we illustrated that the combination of bulk lrRNA-seq and scRNA-seq, paired with suitable quantification algorithms, unlocked isoform-level expression analysis in single cells, aligning with the expectations outlined in Chapter 3. The use of full-length scRNA-seq heightened isoform detection sensitivity, while the incorporation of long-read data, even from bulk samples originating in distinct yet related brain regions, enabled the extraction of highly specific cell type expression signals (section 4.3.3). This was further affirmed through our multi-group Differential Expression (DE) analysis, which revealed a substantial portion of quantified isoforms with cell type-specific expression profiles. The combined application of a zero downweighting strategy [189] and a boostrap-based DE analysis stands as an additional contribution to the field, particularly when complex expression signals need to be derived for downstream analysis. These insights, in conjunction with the novel methodologies introduced in Chapter 5, equip researchers with valuable tools to explore the role of post-transcriptional regulation in shaping cell type identity.

Chapter 5

A novel method to derive isoform co-usage networks from single-cell data

Chapter adapted from Arzalluz-Luque, Á., Salguero, P., Tarazona, S., Conesa, A. *acorde* unravels functionally interpretable networks of isoform co-usage from single cell data. *Nature Communications* **13**, 1828 (2022).

5.1 Introduction

Single-cell RNA-seq (scRNA-seq) has revolutionized transcriptomics analysis, especially as the development of technologies with increasingly high throughput has boosted the amount of biological diversity that can be captured in a sequencing experiment [38]. The technology has been extensively applied to the discovery of new cell types and the characterization of their transcriptional profiles [15, 16, 18, 197–199]. These studies rely on the low number of features required to recapitulate the cell type structure of the data [200], which situates cell type characterization efforts at the baseline of single-cell biology. Other biological phenomena, such as regulatory networks or dynamic processes, may however show a wider range of complexity in their transcriptional encoding. The application of scRNA-seq to some of these aspects has resulted in the development of powerful single-cell-specific methods, such as pseudotime [201, 202] and RNA-velocity [203] analyses, which have provided insight on cell differentiation and the mechanisms behind cell state transitions [199, 204–208].

Single-cell research is nevertheless far from realizing its full potential for the investigation of the deep layers of cell regulation. In particular, Alternative Splicing (AS) and isoform expression dynamics have remained a challenge to the field. This is intrinsically related to the inability of short-read scRNA-Seq methods to fully identify alternative isoforms, as discussed in chapter 3. Most computational methods and studies therefore leave isoform characterization aside [102, 104, 139, 209–212], with recent exceptions [77]. Meanwhile, long-read RNA sequencing (lrRNA-seq) technologies are emerging as an increasingly powerful alternative for single-cell isoform studies thanks to improvements in their sequencing depth constraints, which are fully described in chapter 3. Recent long-read studies have achieved promising advances, showing that cell type-specific isoform selection patterns can be found in both broad cell types as well as subtypes [35, 78, 145, 151]. In spite of this recent progress, single-cell lrRNA-Seq is yet to match the amount of isoform diversity captured by short-read scRNA-Seq.

Notwithstanding this challenging scenario, the analysis of AS in single cells has largely contributed to expand the field's understanding of cell identity and function. Recent studies have shown that splicing differences can be used to discriminate cell types with as much accuracy as when using gene expression [213]. Moreover, the integration of gene expression and AS analyses has been reported to unlock the discovery of previously undetected cell types and states [77, 214–216]. In spite of this, the mechanistic patterns underlying cell-level AS remain mostly unknown. A glaring example of this is the ongoing controversy regarding the ability of individual cells to express one or several isoforms. Over the years, successive studies have provided non-conclusive results, with evidence of bimodal splicing patterns [101, 102, 217] as well as concerns regarding the relationship between bimodal isoform detection and technical noise [132, 218]. Another pending question for the field is whether isoform expression programs involve co-expression relationships between transcript variants from different genes. So far, the application of lrRNA-Seq to single cells has served to unravel coordinated event choice patterns within isoforms of the same gene [84, 91], however, cross-gene isoform expression newtorks have not yet been investigated. In spite of the present research gap, a certain degree of codependency between genes regarding the selection of transcript variants from their isoform repertoire would be expected as a result of splicing regulation. All in all, the proven relevance of AS for cell-type identity, together with the high number of pending questions, demand innovative approaches to leverage the myriad of highly dimensional scRNA-Seq datasets and extract these complex signals.

In this chapter, we hypothesize that cross-gene isoform expression coordination arises as a consequence of AS regulation, and that it can be computationally detected in the form of isoform groups showing co-variation across cell types. To demonstrate this, we designed acorde [219], an end-to-end pipeline for the study of isoform co-expression networks (figure 5.1), and applied it to the analysis of two publicly available mouse neural datasets [122, 163]. First, bulk long-reads and single-cell Illumina sequencing were integrated to estimate isoform expression (discussed in chapter 4, section 4.2.9). Next, to unlock the limitations of extant correlation metrics in scRNA-Seq data [220], we developed a novel strategy to obtain noise-robust correlation estimates, followed by a semi-automated clustering approach to detect modules of co-expressed isoforms (sections 5.3.1 and 5.3.2 of the present chapter). We also defined and implemented Differential Isoform Usage (DIU) and co-Differential Isoform Usage (coDIU) analyses in order to leverage the multiple cell types contained in single-cell datasets (section 5.3.3). In addition, to couple coDIU with a biologically interpretable readout, transcripts and predicted proteins were functionally annotated (already described in chapter 4, section

4.2.3). Finally, using these annotations, a number of functional analyses were performed to unravel potential implications of isoform co-expression for cell identity and function, results for which will be discussed in section 5.3.4.

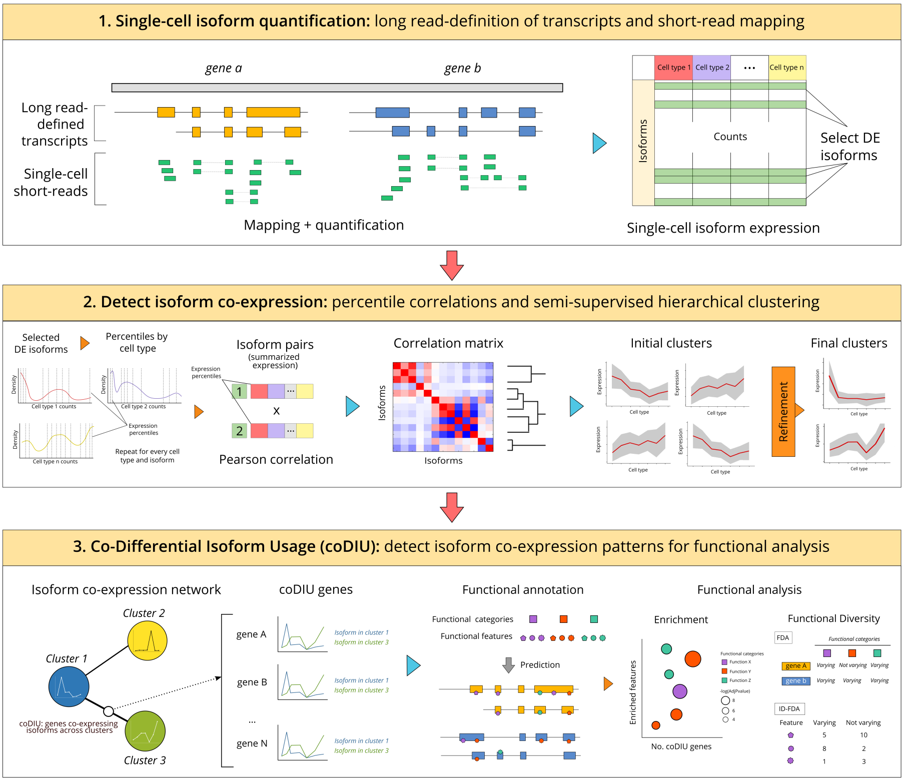


Figure 5.1: *acorde* workflow. The pipeline in this thesis, which includes the acorde method for detection of differential isoform usage (DIU) and co-usage (coDIU) across cell types, consists in three main analyses. First, single-cell short and bulk long-reads were integrated for transcriptome definition, followed by multi-group differential expression testing (described in chapter 4). Next, percentile correlations were computed to cluster isoforms with similar expression patterns across cell types. Finally, gene pairs were tested for co-differential isoform usage, detecting genes that form co-expression relationships for subsequent functional analysis.

5.2 Methods

5.2.1 Data and software availability

Accession codes for the single-cell short-read and bulk long-read datasets used in this analysis as well as a thorough description of their pre-processing have been provided in chapter 4, sections 4.2.1, 4.2.2 and 4.2.8.

The code used to perform the analyses in this manuscript has been implemented in the acorde R package, available at https://github.com/ConesaLab/acorde. Specifically, all analyses have been run using acorde v0.1.0 [219]. The isoform-level expression matrices used in this study are available as data objects under the `data` folder in the acorde R package repository.

5.2.2 Percentile correlation

In order to assess the similarity of isoform expression profiles across cells, a correlation measurement can be used by taking cells as observations. We propose here instead to first summarize the expression within a given cell type with percentiles and then compute the correlation using all cell types and their percentiles as observations, a method that we refer to as *percentile correlation*.

Percentile correlations rely on the assumption that cell-to-cell differences can be mostly attributed to transcriptional stochasticity or technical noise, and that these within-cell type differences have a smaller effect than between-cell type expression differences. However, expression estimates for transcripts within the same cell are biased in different degrees, mostly depending on their expression levels, with lower expression being generally accompanied by higher noise levels [108]. This modifies the extent to which isoforms are affected by noise in each cell and causes strong cell-level effects that prevent the detection of co-expression relationships using solely cell-level measurements. Instead, we set out to target changes in expression across cell groups. We therefore considered isoform expression levels in the different cell types as a range of possible values, defined by the cell-level measurements in the data. In this context, the expression value of an isoform in a cell is used a proxy to infer the underlying distribution of expression values in the cell type, where the shape and width of this distribution will depend on both biological and technical factors.

To translate this into a metric, the expression values of an isoform in each of the cell types are used to compute a number of percentiles (p). A default value of $p = 10$ was established to achieve a good balance between accuracy and computational burden in downstream analysis. As the minimum expression value (percentile 0) was also included, we obtained 11 values representing the expression range within a given cell type. As a result, each isoform possessed a new, recalculated expression vector where the percentile values computed in each cell type will replace cell-level expression estimates. This process was repeated for each isoform. Next, Pearson correlations were computed between every possible pair of isoforms using the `cor()` function in the in the R-base stats package [221], obtaining a percentile correlation matrix **R**. In this context, high correlations will appear if a pair of isoforms shows a similarly broad expression distribution in most cell types, as well as a similar amount of relative expression change between cell types.

5.2.3 Semi-automated isoform clustering

In order to obtain modules of tightly co-expressed isoforms, the hierarchical clustering algorithm [222] was combined with several rounds of cluster profile refinement, in order to automate the most intensive steps of clustering while also granting control over the level of aggregation and within-cluster similarity. Clustering and refinement steps can be combined and re-arranged to best capture co-expression patterns within the data, and their parameters can be defined by potential future users to provide maximum flexibility. Functions for clustering and refinement are implemented and documented in the acorde R package (https://github.com/ConesaLab/acorde). Code-level details on how these functions work and how were used to generate the results in this chapter are supplied in Appendix I. Therefore, this section will include a general description of the clustering and refinement strategies in the package and a brief guide of how they were combined to obtain isoform clusters.

Dynamic hierarchical clustering

The previously obtained correlation matrix ($\mathbf{R}$), where each element r_{ij} represents the Pearson's correlation coefficient between the percentiles of isoforms (i, j), was transformed into a distance metric to be used in the hierarchical clustering. As we aimed to cluster positively correlated isoforms given our biological hypothesis, negative correlation values were discarded by replacing them with zero values, and therefore defined the distance between any pair of isoforms i and j as in equation 5.1.

$$d_{ij} = \begin{Bmatrix} 1 - r_{ij} & if\ r_{ij} > 0 \\ 1 & if\ r_{ij} \leq 0 \end{Bmatrix} \tag{5.1}$$

Hierarchical cluster analysis was performed using the `hclust()` function in the R stats package [221] with the average linkage criterion, and obtained a dendrogram. To obtain clusters, we used the `cutreeHybrid()` function in the dynamicTreeCut R package [222] in order to find different thresholds for different branches of the dendrogram tree, instead of using a fixed threshold for the entire dendrogram. The following non-default parameters were provided to the `cutreeHybrid()` function: `deepSplit = 4`, `pamStage = FALSE`, `minClusterSize = 20`. Briefly, the `deepSplit` argument ranges between 0 and 4, and provides smaller clusters, more accurate clusters when set to high values. `pamStage`, on the other hand, determines whether a second stage of clustering using an algorithm similar to the Partition Around Medoids (PAM) method will be performed after searching the dendrogram for clusters (see Langfelder et al. [222]). As a result of this PAM-like step, no items are left unassigned to clusters, while setting `pamStage = FALSE` allows unclustered items. Finally, `minClusterSize` determines the minimum size of the produced clusters, and thus passing a higher value to this argument prevents the generation of too many clusters with a very small number of items.

This initial set of clusters is to be used as “hooks” to gather as much expression profile diversity from the data as possible. Importantly, even though our parametrization allows isoforms to remain unassigned to clusters, some isoforms may still show low similarity to their cluster’s profile. To be able to obtain profiles as consistent as possible for downstream refinement, a cluster quality control step was included in acorde to remove isoforms based on a minimum correlation threshold with the rest of the members. For our study, isoforms were moved to the unclustered group if they showed a correlation lower than 0.85 with 3 or more isoforms from their cluster. In this manner, only tightly-correlated groups of isoforms will remain clustered.

Expanding clusters with unassigned isoforms

To re-assign unclustered isoforms to clusters with which they show high correlation, acorde allows correlation-based cluster expansion. In this process, each cluster profile is summarized into an average representative transcript, hereby referred to as *metatranscript*. Metatranscripts are calculated as the mean of the percentile-based expression of all isoforms in the cluster. As a result, $11{\cdot}K$ (K being the number of cell types) mean-summarized percentile expression values are obtained, which can be understood as an approximation to the expression range shown by the isoforms from that cluster in each of the cell types. Next, correlations between metatranscripts and unclustered isoforms are computed and unclustered isoforms assigned if they show percentile correlation values above a specified threshold with at least one cluster. For the analyses in the present chapter, a correlation threshold of

0.9 was set, where the maximally correlated cluster was selected as the best match in case of ties.

Merging clusters by profile similarity

Prioritizing the reduction of within-cluster variability may lead to obtaining a large number of small, redundant clusters. To mitigate this effect while also preserving high correlations between cluster members, acorde can be used to merge clusters by profile similarity using the percentile correlations between their metatranscripts. To perform the analyses included in this chapter, hierarchical clustering was performed on the metatranscript correlation matrix via the `hclust()` function (stats R package [221]), subsequently creating clusters with the `cutree()` function (stats R package [221]) and a height cutoff of 0.1. Since this merging process may result in joining clusters with highly uncorrelated profiles, the cluster expansion process described in section 5.2.3 was next used for the re-assignment of isoforms from clusters flagged as inconsistent. Details and graphics representing the intermediate steps in the clustering process are available in Appendix I.

Recursive assignment of remaining unclustered isoforms

First, extant clusters were filtered again to maximize similarities between members of the same isoform group and generate reliable profiles for expansion. In this case, isoforms were returned to the unclustered group if they had percentile correlation lower than 0.7 with 10 or more isoforms of their cluster. Next, and following the cluster expansion process described above, percentile correlations between the isoforms to be assigned and cluster metatranscripts were computed. In this case, however, assignment was performed as a recursive process, in which (1) isoforms joined a cluster based on percentile correlation with its metatranscript, (2) metatranscripts were re-calculated for the newly expanded clusters and (3) assignment was performed again for the remaining unclustered isoforms. The percentile correlation thresholds was sequentially lowered from 0.9 to 0.8 and 0.7. Finally, any isoforms remaining unclustered at this point were assigned to the clusters with which they presented maximum correlation. In doing this, unclustered isoform groups were assigned in order, and highly correlated elements therefore contributed to strengthen within-cluster similarities before assigning more lowly correlated elements.

Finally, expanded clusters were merged again to remove remaining redundancies and generate larger clusters for subsequent co-Differential Isoform Usage (coDIU) detection. The strategy and parameters used were similar to those detailed in section 5.2.3, however, a complete description of this process is supplied in Appendix I.

5.2.4 Co-expression pattern simulation

To validate percentile correlations and our clustering strategy, we evaluated their performance on synthetic data, where co-expression relationships between simulated features need to be pre-defined as part of the data simulation process. However, there is, to the best of our knowledge, no currently available strategy to simulate single-cell data including modules of co-expressed features. We therefore designed our own simulation strategy by combining the SymSim R package [223] to adequately model single-cell RNA-Seq data, and a dedicated strategy to generate co-expression between SymSim simulated features.

First, we set the following parameters to the SimulateTrueCounts() function in SymSim in order to obtain a count matrix consisting in 1000 cells from 8 cell types and 8000 features, with sufficient feature-level variation between the different cell groups:

```
SimulateTrueCounts(ncells_total = 1000, min_popsize = 100,
                   i_minpop = 1, ngenes = 8000, nevf = 10,
                   n_de_evf = 9, evf_type = ‘‘discrete’’,
                   phyla = pbtree(n = 7, type = ‘‘discrete),
                   vary = ‘‘s’’, Sigma = 0.25,
                   gene_effect_prob = 0.5, bimod = 0.4,
                   prop_hge = 0.03, mean_hge = 5)
```

Next, we modeled technical effects on these true counts in order to obtain real, observed counts using the True2ObservedCounts() function in SymSim, with the following parameters:

```
True2ObservedCounts(true_counts$counts,
                    meta_cell = true_counts$cell_meta,
                    protocol = ‘‘nonUMI’’,
                    alpha_mean = 0.1, alpha_sd = 0.005,
                    lenslope = 0,
                    gene_len = rep(1000, nrow(true_counts$counts)),
                    depth_mean = 4e6, depth_sd = 1e4)
```

To create co-expression patterns, we then re-ranked expression values on a cell type-specific manner to define synthetic features, based on the expression profile of 15 pre-defined co-expression modules.

First, we drafted 15 different co-expression profiles reflecting three levels of expression complexity, that is, showing high expression or expression "peaks" in one, two, or three cell groups, respectively. To generate a count matrix reflecting these expression patterns, simulated counts were shuffled to create new, synthetic features. To achieve this, features in each cell group were first re-ranked by mean expression across cells in the group, breaking feature connectivity between the simulated cell types. Then, the top 1,400 features from each cell type were selected, together with the bottom 1,400 features. In this manner, high-expression and low-expression count vectors for each group were obtained, which were then combined to create synthetic features following the pre-designed cluster's co-expression pattern. For each cluster, 200 count vectors from top-expression features were assigned to peaking groups, and 200 count vectors form bottom-expression features to cell groups showing low expression. Of note, 1,400 features were selected in order to grant at least 7 different 200-feature groups could be generated for each cell type, thus minimizing the probability that the same expression vector is selected for two different patterns. Nevertheless, this random selection process was repeated 15 times to generate the different clusters. Co-expression simulation was implemented in the `simulate_coexpression()` function in the acorde R package, documenting its usage on a separate vignette that is available in Appendix I.

All in all, a simulated count matrix containing 1,000 cells from 8 cell types and 3,000 synthetic features was obtained, all of which belonged to one of the 15 simulated co-expression modules. Therefore, by breaking feature-level connectivity between cell types, we benefited from feature-specific properties at the cell type level, while re-creating cell type expression coordination patterns that the SymSim strategy was not able to generate. Finally, to ensure the quality of the simulated clusters, we filtered synthetic features if their Pearson correlation with the cluster's median profile was below 0.75 (see section 5.2.3).

5.2.5 Benchmarking of isoform correlation metrics for scRNA-seq data

Traditional correlation metrics have been shown to perform poorly when applied to scRNA-seq data, mainly given the increased noise and stochasticity levels in this data type. Recently, extensive benchmarks including single cell-tailored metrics have shed light on how to better select correlation metrics for single-cell data (see review by Skinnider et al. [220]). We therefore compared the performance of percentile correlations to a representative set of correlation metrics used in single-cell co-expression studies, namely classical Pearson and Spearman correlations, single-cell designed zero-inflated Kendall correlation [224], and proportionality metric rho (ρ) [225], in agreement with previous reports showing that proportionality metrics were among the best performing co-expression methods in single-cell data. To measure performance, we computed these five co-expression metrics for all the synthetic features in the previously-simulated dataset, generating five different distance matrices for clustering, and evaluated which metric best recapitulated the simulated co-expression modules when used in our clustering pipeline. Pearson and Spearman correlations were computed using the `cor()` function in the R-base stats package [221]. Zero-inflated Kendall correlation and rho (ρ) were computed using the `dismay()` function in the dismay R package [220].

To make our benchmarking comparable, we adapted our clustering pipeline to remove all non-automated steps and always generate a fixed number of clusters. First, hierarchical clustering was performed on each correlation matrix using `dynamicTreeCut()` [222] and the following non-default parameters to maximize granularity: `deepSplit = 4`, `pamStage = FALSE`, `minClusterSize = 10`. Of note, we skipped the quality filtering step based on intra-cluster correlations (see section 5.2.3) to avoid bias against metrics that tend to yield low values when applied to single-cell data. Since we intended to evaluate the number of features remaining unclustered using each metric, we additionally suppressed the unclustered isoform assignment step (section 5.2.3). Finally, the merging process was automated by using the traditional hierarchical clustering algorithm (implemented in the `hclust()` function in the R stats package [221]) to group clusters based on the inferred meta-transcripts that summarize the cluster's expression profile (section 5.2.3). Finally, we set the number of clusters to 15, i.e. the number of simulated co-expression modules.

In addition to the number of unclustered isoforms, we used the levels of internal correlation in the empirical clusters, i.e. those obtained by *de novo* clustering of simulated synthetic features, to evaluate the clustering. We did this by jointly considering all pairwise metric values for features within a cluster and measuring the percentage of metrics that are above a threshold value of 0.8. To assess how well empirical clusters recapitulated the co-expression simulation, we paired empirical with simulated clusters using the correlations between their mean cluster profiles. Simulated clusters were therefore paired with the empirical clustering showing maximum profile correlation. We next compared synthetic feature IDs assigned to the obtained and empirical clusters in each pair using the Jaccard Index (JI).

5.2.6 Differential Isoform Usage and co-Differential Isoform Usage across multiple groups

Defining Differential Isoform Usage across multiple groups

Grouping isoforms into different clusters allows detection of a number of expression patterns across the multiple cell types included in single-cell data. As previously described, we filtered DE isoforms to ensure that all transcripts had at least one other counterpart from the same gene that was also significantly DE. Intuitively, in order for Differential Isoform Usage (DIU) to occur, a gene must first have at least two DE isoforms. However, we only considered a gene to be positive for DIU if (at least) two isoforms were DE and were assigned to different clusters, indicating that two of the gene's isoforms show different expression patterns across groups (see figure 5.8.a). Ultimately, this can be interpreted as an indicator that isoform expression regulation is cell type-dependent in that gene.

Detecting co-splicing patterns across isoform clusters: co-Differential Isoform Usage

We define coordinated splicing patterns as a situation where post-transcriptional regulation, defined by isoform expression, can be detected independently of transcriptional regulation, i.e. gene-level expression. To detect splicing coordination, we defined co-Differential Isoform Usage (coDIU) as a pattern where a group of genes shows co-expression of their isoforms, but no co-expression can be detected when only gene expression is considered (see figure 5.8.b). In the context of our pipeline, a set of potentially coDIU genes will have at least two of their isoforms assigned to the same clusters, therefore showing detectable isoform-level co-expression, and suggesting coordinated splicing regulation in that group of genes. However, clustering allows expression pattern variability among members, and therefore some isoforms might be assigned to clusters that do not faithfully represent their expression profile, leading to detection of false-positive coDIU genes.

To identify groups of genes that constitute candidates for coDIU, we applied negative-binomial generalized linear regression models. Let G be a group of genes, each of them with I_g isoforms, where $g = 1, \cdots, |G|$. At least one of the isoforms of each gene g in G must belong to the same cluster c, where $c \in 1, \cdots, C$ and C is the total number of clusters. Let z be the expression vector obtained after concatenating the expression vectors y_i of each isoform i of every gene $g = 1, \cdots, |G|$. For the sake of simplicity, let us assume that $|G| = 2$, $I_g = 2\ \forall g$, and consequently $C = 2$. In this case, vector z will contain $4N$ elements, where N is the total number of cells in the data ($N = 241$ in our data) and will be the response variable in our regression model. We need to assess if z values follow the trend depicted in figure 5.8.b, that is, the average profile across cell types of the two isoforms in cluster 1 must be significantly different to the average profile of the two isoforms in cluster 2. In addition, the average profile of the two isoforms of gene 1 must not be different to the average profile of the two isoforms of gene 2. To identify groups of genes with these characteristics, we proposed to fit the regression model in equation 5.2 and select the group of tested genes as coDIU candidates when having a significant interaction between cluster and cell type effects, and a non-significant interaction between gene and cell type effects.

$$h(z) = \beta_0 + \beta_1 G_2 + \beta_2 C_2 + \sum_{k=2}^{K} \gamma_k T_k + \beta_3 G_2 C_2 + \sum_{k=2}^{K} \delta_k T_k G_2 + \sum_{k=2}^{K} \tau_k T_k C_2 + \epsilon, \tag{5.2}$$

where G_2 and C_2 are dummy variables indicating whether the expression value corresponds to gene or cluster 2 (value 1) or 1 (value 0), respectively, T_k is a dummy variable which takes value 1 when the corresponding cell is assigned to cell type k ($k = 1, \cdots, K$) and 0 otherwise, β_k, γ_k, δ_k and τ_k are the regression coefficients, ϵ represents the error term, and $h()$ is the link function of the GLM (natural logarithm in this case).

We fitted the GLM model with the `glm()` function in the R-base package [221], and the `negative.binomial()` function in the MASS R package [226], with $\theta = 10$. To test the significance level of the $cluster \times cell\,type$

and $gene \times cell\,type$ interactions, we calculated type-II analysis-of-variance (ANOVA) tables for the model using a likelihood-ratio χ^2 test, implemented in the `Anova()` function from the car R package [227], since we had an unbalanced design. P-values for each of the interactions were separately adjusted using the Benjamini & Hochberg correction. Gene pairs were considered positive for coDIU if FDR adjusted p-value$<$0.05 for the $cluster \times cell\,type$ interaction and FDR adjusted p-value$>$0.05 for the $gene \times cell\,type$ interaction. In other words, we required expression variance across cell types to be a function of the expression profile captured by the clustering, while imposing the additional limitation that aggregating expression by gene must make this effect undetectable. Given that all genes with clustered isoforms will form pairs with all potentially coDIU counterparts and be repeatedly tested, we considered genes to be positive for coDIU if they met the significance criteria in at least one of these pairwise tests.

5.2.7 Functional analyses

The analyses in this manuscript are based on a long read-defined transcriptome which, after careful quality control and curation of the isoform models, was further annotated using IsoAnnotLite (https://isoannot.tappas.org/isoannot-lite/) to include positionally-defined functional features in the annotation (see Chapter 4). Functional features are grouped in functional categories depending on the database from which the information was retrieved and on the biological functions performed by the features (comprehensive list in Chapter 4). In this manner, we gathered sufficient information to couple our co-expression analyses with a biological readout. The specific analysis strategies used to this end are detailed below.

Functional Enrichment Analysis

In order to understand the functional properties of AS-regulated and co-regulated genes, we set out to characterize DIU and coDIU genes using different functional enrichment analysis approaches. In this manner, we intended to gain insight on functional features and categories showing significant overrepresentation in each of these two gene lists, in comparison to different backgrounds, i.e. lists of genes to compare to in order to detect enrichment.

In the case of DIU genes, we calculated enrichment relative to genes with multiple DE isoforms in order to discriminate the functional properties of genes regulated by alternative splicing, as opposed to those lacking differential usage of their isoforms. We considered all annotated functional categories and features, and applied tappAS Functional Enrichment Analysis [121], which relies on the GOSeq R package [228]. Briefly, the method performs an overrepresentation Fisher's Exact test for each functional feature, considering the number of genes annotated with the feature in the tests and background lists. In addition, however, GOSeq accounts for the length bias in overrepresentation detection by downweighting the contribution of longer genes. tappAS next corrects for multiple testing within each functional category by the Benjamini-Hochberg method, allowing multiple functional databases to be included or excluded from the analysis without influencing the number of significant features after p-value adjustment. Significant enrichment for the different tests was defined using a threshold of FDR<0.05.

For coDIU genes, we designed a different strategy in order to improve the statistical power of our functional enrichment analysis, aiming to compare functional properties between splicing regulation (DIU) and co-regulation (coDIU). As stated above, DIU is best measured by using genes with DE isoforms as background. Intuitively, coDIU genes should then be characterized by comparing them to DIU genes. To accommodate these two test/background lists in a functional enrichment analysis without ignoring the overlap between the coDIU and DIU gene groups, we computed enrichment using a partially overlapping samples z-test via the `Prop.test()` function in the Partiallyoverlapping R package [229]. Specifically, we compared the proportion of coDIU genes containing each of the functional features (relative to DIU genes) with the proportion of DIU genes containing that same annotation (with respect to genes with DE isoforms). In other words, we tested whether the proportion of coDIU vs DIU genes including a given functional feature was significantly higher than that shown in the comparison between DIU and DE genes. We performed the analysis for features with more than 15 annotated genes, and subsequently corrected for multiple testing within functional categories using the Benjamini-Hochberg method. For GO terms, ontologies with more than 150 annotated genes were also removed to eliminate excessively broad -and potentially less meaningful- functions. Functional features were considered to be present in a significantly higher proportion in coDIU genes when FDR<0.05.

Annotations used in all functional analyses included Gene Ontology (GO) terms. The hierarchical structure of the Gene Ontology database can often result in multiple significantly enriched terms that refer to the same, or very similar, functions, components and processes. To enhance visualization and result interpretation of coDIU functional enrichment results, we used Revigo [230] to perform a semantic similarity analysis of all significant GO terms obtained in the partially overlapping samples test. We applied a dispensability (a measure of semantic similarity) threshold of 0.5 to assign GO terms to a cluster, and then selected a representative of each term cluster to be included in the visualization.

Functional Diversity Analysis

To obtain insight into the functional changes generated as a consequence of DIU and coDIU, we again used the tappAS tool for the functional analysis of alternative splicing [121]. In particular, we first applied tappAS' Functional Diversity Analysis (FDA) module (see main text Figure 5c). Briefly, FDA performs a within-gene comparison of all the isoforms included in the analysis, aiming to detect whether they present variation in the inclusion of a functional feature. In FDA, variation can be positional, i.e. one or more of the gene's isoforms present a change in the genomic coordinates defining the feature, or be defined by presence/absence, i.e. at least one of the isoforms lacks a feature that is present in the rest. As a result, FDA provides analyzed genes with a label for each of the feature categories included in the transcriptome's functional annotation file, flagging them as varying if at least one of the isoforms presents variation in a feature from that category, or not varying if no changes are detected. For more details on FDA, see the Methods section in de la Fuente et al. [121].

We run both positional and presence/absence FDA for three gene sets: (1) genes with multiple DE isoforms, (2) DIU genes and (3) coDIU genes. Next, for each of these gene sets, we computed the proportion of varying genes detected for each functional category. Varying proportions were calculated relative to the total number of genes including annotations from the category, instead of considering all genes in the set. In this manner, we avoided underestimating variation rates for categories that were less represented in the functional annotation file. In order to check whether any of these gene sets presented a significantly higher mean proportion of varying genes across categories, we performed a paired t-test for each combination of gene set pairs: DIU vs multiple DE, coDIU vs multiple DE, and coDIU vs DIU. In this analysis, we considered functional categories to be the individuals under evaluation, while the proportion of varying genes calculated for each category in the two tested sets constituted the paired observations. As a result, we obtained three p-values per FDA analysis type, i.e. presence/absence and positional variation.

To better understand the functional readout that can be obtained using the acorde pipeline, we analyzed a subset of the coDIU gene network, namely three clusters showing related isoform co-expression patterns: neuron-specific expression (cluster 1), oligodendrocyte-specific expression (cluster 14) and expression in both neural and oligodendrocyte cell types (cluster 4). To characterize functional variation among the clusters, we used positional/presence

FDA (see above) and ID-level FDA. ID-level FDA is also included in tappAS [121] and provides a within-feature summary of FDA results. In other words, ID-level FDA ultimately reports the number of varying and not varying genes detected for each feature ID included in a given functional category. In this case, varying status obeys a similar criterion to the one described above, i.e. genes in which at least one isoform shows differential inclusion/exclusion of the feature. Since each functional category may include several features, ID-level FDA provides a complementary view to that of FDA, allowing users to inspect which particular features are more frequently changing as a result of the category-level functional variation reported in FDA. For more details on ID-level FDA, see the Methods section in de la Fuente et al. [121].

5.2.8 Analysis of GABA-ergic neuron subtypes

To illustrate the applicability of the acorde approach, we retrieved additional single-cell RNA-Seq data from a second study by Tasic et al. [122] (accession codes provided in Chapter 4, section 4.2.1). Cell-level isoform expression estimates were obtained with Kallisto [87] using our long read-generated neural transcriptome (see chapter 4) and the mouse genome assembly version in Chapter 4, section 4.2.1. Cell clusters and the cell subtype labels assigned to them by authors in the original study were retrieved and used in the analysis. After quality filtering of cells (1.25e6<total counts<2.75e6) and lowly expressed isoforms (counts>0 in at least 25% of one cell type), we additionally removed isoforms that did not accumulate more than 10% of their gene's expression in at least one cell type. Cells belonging to cell types *Meis2* (n = 43) and *Serpinf1* (n = 22) were discarded to balance cell type abundances, as the remaining cell types had approximately 1000 cells each. Differentially Expressed (DE) isoforms were next computed by combining ZinBWAvE weights [189] and DE testing using the edgeR R package [190], as described in chapter 4, section 4.2.9. Isoforms with FDR$<$0.05 and fold-change$>$1.5 between at least one pair of cell types were selected for downstream analysis. After computing percentile correlations with $p = 10$, DE isoforms were clustered using dynamic hierarchical clustering with the following non-default parameters for the `cutreeHybrid()` function in the dynamicTreeCut R package [222]: `deepSplit = 4`, `pamStage = FALSE`, `minClusterSize = 2`, `cutHeight = 0.1`. Parameters were fine-tuned to generate a high number of clusters with as accurate a profile as possible. Unclustered isoforms were assigned to clusters using the percentile correlation with cluster metatranscripts and successively decreasing thresholds. Expanded clusters were then merged by dynamic clustering of their metatranscripts using the following non-default parameters for the `cutreeHybrid()` function: `pamStage = FALSE`, `cutHeight = 0.3`. Details on the clustering process can be found in section 5.2.3. Detection of DIU and coDIU genes was performed as described in section 5.2.6.

5.3 Results

5.3.1 Detecting cell type-dependent isoform co-expression

Co-expression signals in single-cell data are weak and have often resulted in a poor performance of traditional correlation and network inference methods

[220, 231]. Although data transformation approaches [232] and alternative metrics [220] have been proposed, these can be complex to apply and considerably less interpretable, respectively. Furthermore, most of these studies have only investigated gene-level co-expression [233], often ignoring the post-transcriptional regulatory landscape. To address these limitations, we implemented a percentile correlation strategy: a simple, scalable approach to overcome single-cell noise in isoform co-expression studies (figure 5.2.a).

For the purpose of consistency, the Tasic et al. [163] short-read, single-cell dataset used throughout chapter 4 was used to demonstrate percentile correlations and characterize their impact on isoform co-expression detection. Briefly, the curated, long read-defined transcriptome obtained in sections 4.2.2 and 4.2.2 was used for isoform-level quantification, details for which are supplied in section 4.2.8. After quality control and Differential Expression (DE) analysis, this dataset included 1,591 cells from 7 cell types and 8,424 isoforms from 4,223 genes (see Chapter 4, section 4.3.3). However, since two or more isoforms with differential cell type expression are required to form co-splicing relationships, transcripts with no other same-gene DE counterpart were removed, retaining 6,794 isoforms from 2,696 genes.

To design the percentile correlation strategy, a series of assumptions were made regarding the nature and extent of biological and technical signals in scRNA-seq data. First, cell-type identity was defined as a transcriptional state shared by multiple cells and generated as a product of context-specific gene expression. While cell types are arguably difficult to define as a discrete entity, a certain degree of homogeneity can be assumed among closely-

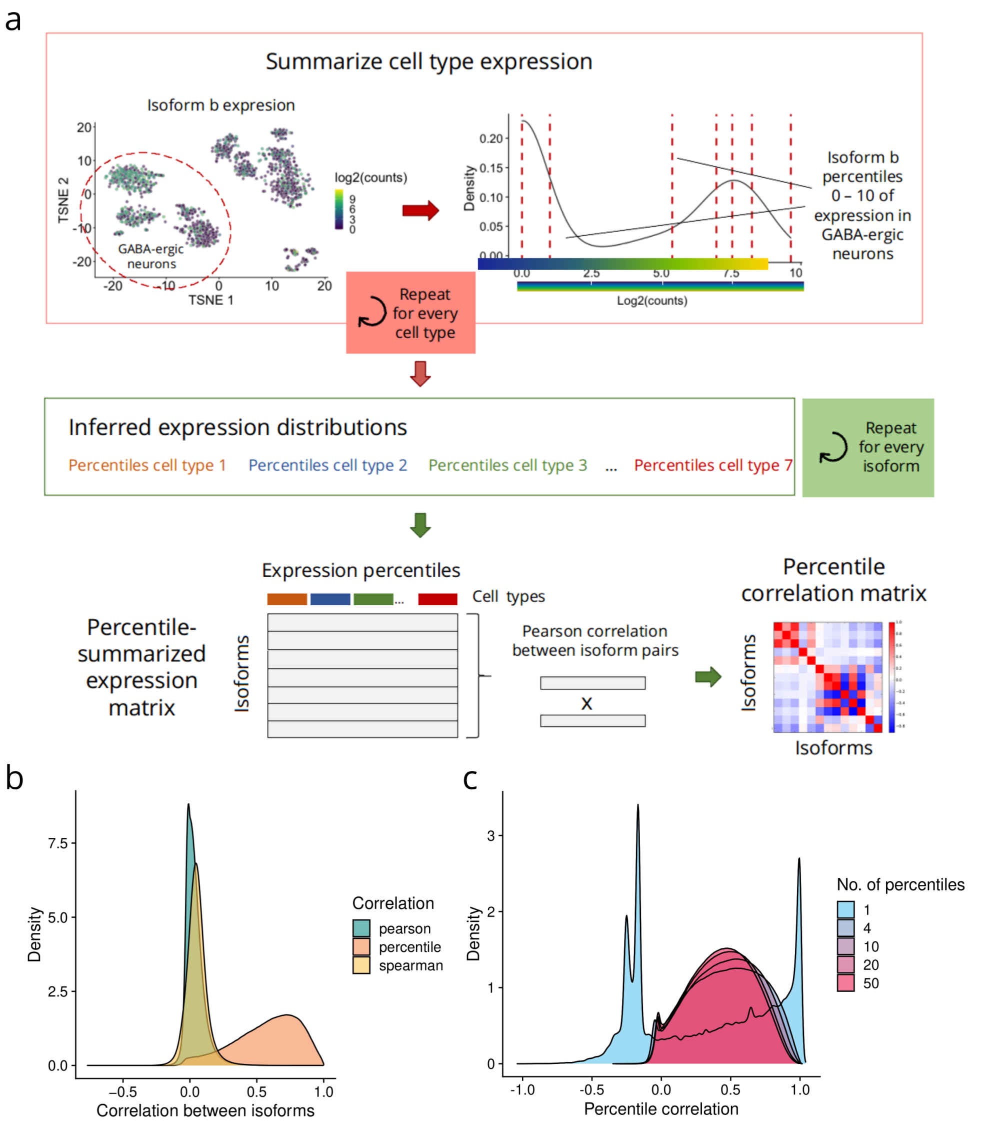


Figure 5.2: Percentile correlations. a Percentile correlation algorithm. For each isoform, cell type-level expression is summarized using percentiles (0–10) as a proxy of the isoform's expression distribution in each of the cell types. Then, Pearson correlations are computed using the percentile-summarized expression of all isoforms, obtaining a percentile correlation matrix. **b** Correlation density distributions. Pairwise isoform correlations were computed using Pearson, Spearman, and percentile (i.e. percentiles and Pearson) correlation. **c** Density distribution of pairwise isoform percentile correlations obtained using 1 (i.e. median), 4, 10, 20 and 50 percentiles.

related cells, and thus within-cell type stochasticity can be attributed to a combination of technical noise [108, 234] and biological mechanisms such as transcriptional bursting [235]. These effects can partially explain the heterogeneous expression patterns that are typically observed in single-cell data, generating high variance and sparsity across genes [234, 236, 237], particularly when expression is lower [108]. Moreover, the co-expression signal in the data can be masked as a result of these properties, yielding consistently low correlation values when traditional correlation metrics are applied (figure 5.2.b). To overcome this, single cells from the same cell-type were hereinafter treated as biological replicates, i.e. instances that represent the state of a discrete cell population but are differently affected by the aforementioned combination of technical and biological forces. In this context, the expression distribution of any given isoform across the selected population can be considered to be the signature of the isoform in that cell-type. Note that, given that different cell identity hierarchies and intermediate states often coexist, these definitions can be extrapolated to any cell population configuration (e.g. a higher level of granularity).

To translate these assumptions into a metric, the expression of each isoform within a given cell type was first summarized into an expression profile, where single-cell count values were replaced by 10 percentile values or deciles (see section 5.2.2). Intuitively, the reduced number of values captured the behavior of any given transcript in the cell type, as these were inferred based on cell-level observations. To grasp similarities between expression distributions across cell types, pairwise Pearson correlations were computed using percentile-summarized isoform expression, resulting in a more meaningful distribution of correlation values than obtained with traditional metrics (figure 5.2.b). Our co-expression metric therefore by-passes cell-level matching of individual observations, providing a correlation estimate that is both robust to the uncertainty of single cell expression and interpretable as a measure of expression similarity. Remarkably, changing percentile number did not have a noticeable effect on the resulting correlation values (figure 5.2.c). Nevertheless, using 1 percentile (median expression), substantially disrupted the correlation value distribution (figure 5.2.c), stressing the importance of selecting a sufficient number of percentiles to avoid over-summarizing isoform expression.

To detect modules of co-expressed isoforms, we used percentile correlations as a distance metric for hierarchical clustering, and designed a semi-automated cluster refinement approach to ensure maximal profile similarity within clustered modules (figure 5.3.a, described in detail in section 5.2.3). First, the dynamicTreeCut R package [222] was used to initialize clustering. The dynamic clustering algorithm enables the selection of adaptive thresholds for better detection of clusters within a dendrogram, after which 166 clusters were obtained. These were then re-clustered to mitigate the presence of highly similar (i.e. redundant) expression profiles (example in figure 5.3.b). To achieve this, cluster metatranscripts (defined in section 5.2.3) were computed and standard hierarchical clustering applied, generating 26 clusters. At this point, 2,381 isoforms remained unclustered, including isoforms from 3 groups that presented noisy expression profiles (figure 5.3.c). These tran-

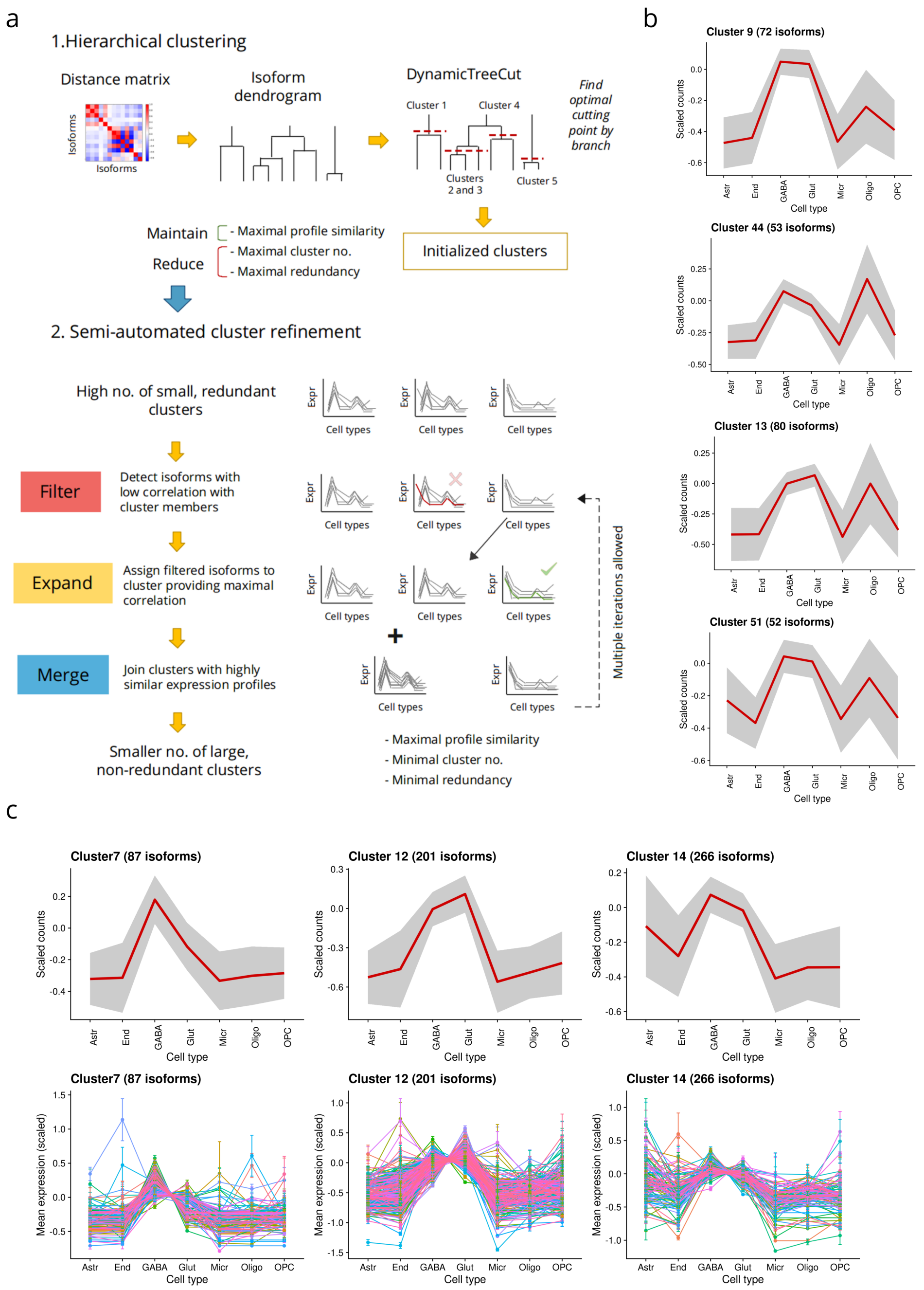

a
1.Hierarchical clustering
Distance matrix
Isoforms
Isoforms
Isoform dendrogram
DynamicTreeCut
Cluster 1
Cluster 4
Clusters 2 and 3
Cluster 5
Find optimal cutting point by branch
Maintain
Reduce
- Maximal profile similarity
- Maximal cluster no.
- Maximal redundancy
Initialized clusters
2. Semi-automated cluster refinement
High no. of small, redundant clusters
Filter
Detect isoforms with low correlation with cluster members
Expand
Assign filtered isoforms to cluster providing maximal correlation
Merge
Join clusters with highly similar expression profiles
Smaller no. of large, non-redundant clusters
Expr
Cell types
Multiple iterations allowed
- Maximal profile similarity
- Minimal cluster no.
- Minimal redundancy
b
Cluster 9 (72 isoforms)
Cluster 44 (53 isoforms)
Cluster 13 (80 isoforms)
Cluster 51 (52 isoforms)
Scaled counts
Cell type
Astr
End
GABA
Glut
Micr
Oligo
OPC
c
Cluster7 (87 isoforms)
Cluster 12 (201 isoforms)
Cluster 14 (266 isoforms)
Scaled counts
Mean expression (scaled)
Cell type
Astr
End
GABA
Glut
Micr
Oligo
OPC

Figure 5.3: Isoform clustering. **a** Clustering pipeline. The percentile correlation matrix is first used as a distance matrix for hierarchical clustering. After dynamic cluster generation, noisy clusters are refined by a three-step semi-automated process. **b** Example of four redundant clusters (i.e. clusters representing the same expression profiles) that were merged by profile similarity. **c** Example of noisy clusters, i.e. clusters grouping isoforms with highly dissimilar expression patterns across cell types, whose isoforms were re-assigned to other clusters. For the single-line plots (b and c), expression is first scaled by subtracting the transcript mean across all cells and dividing centered transcripts by their standard deviations. Cell-level mean expression is then computed for all transcripts and then aggregated as the global cell type mean, represented by the red line. Grey area corresponds to cell type mean ± standard deviation. For the multi-line plots (lower set of plots), each line corresponds to the cell-type mean expression values of one transcript and the error bar corresponds to ± standard deviation, that is, the standard deviation of transcript-level mean values in each cell type.

scripts were assigned by maximizing the similarities between cluster and isoform expression profiles, i.e. using the percentile correlation between isoform expression and cluster metatranscripts (see section 5.2.3). Finally, clusters were merged again to obtain completely unique profiles, reducing the number to 17 clusters. Of note, we spotted two pairs of clusters with clearly similar profiles that were not merged metatranscript clustering. To avoid detection of falsely coDIU genes in downstream analysis, these were merged manually. As a result, we generated a total of 15 distinct clusters (figure 5.4) containing all initially analyzed isoforms (6,794 in total) and representing diverse expression modalities across the 7 broad cell types. The range of cluster refinement strategies used in this section were implemented as functions in the acorde R package [219], and their application to the Tasic dataset to obtain the clusters in figure 5.4 is thoroughly described in Appendix I.

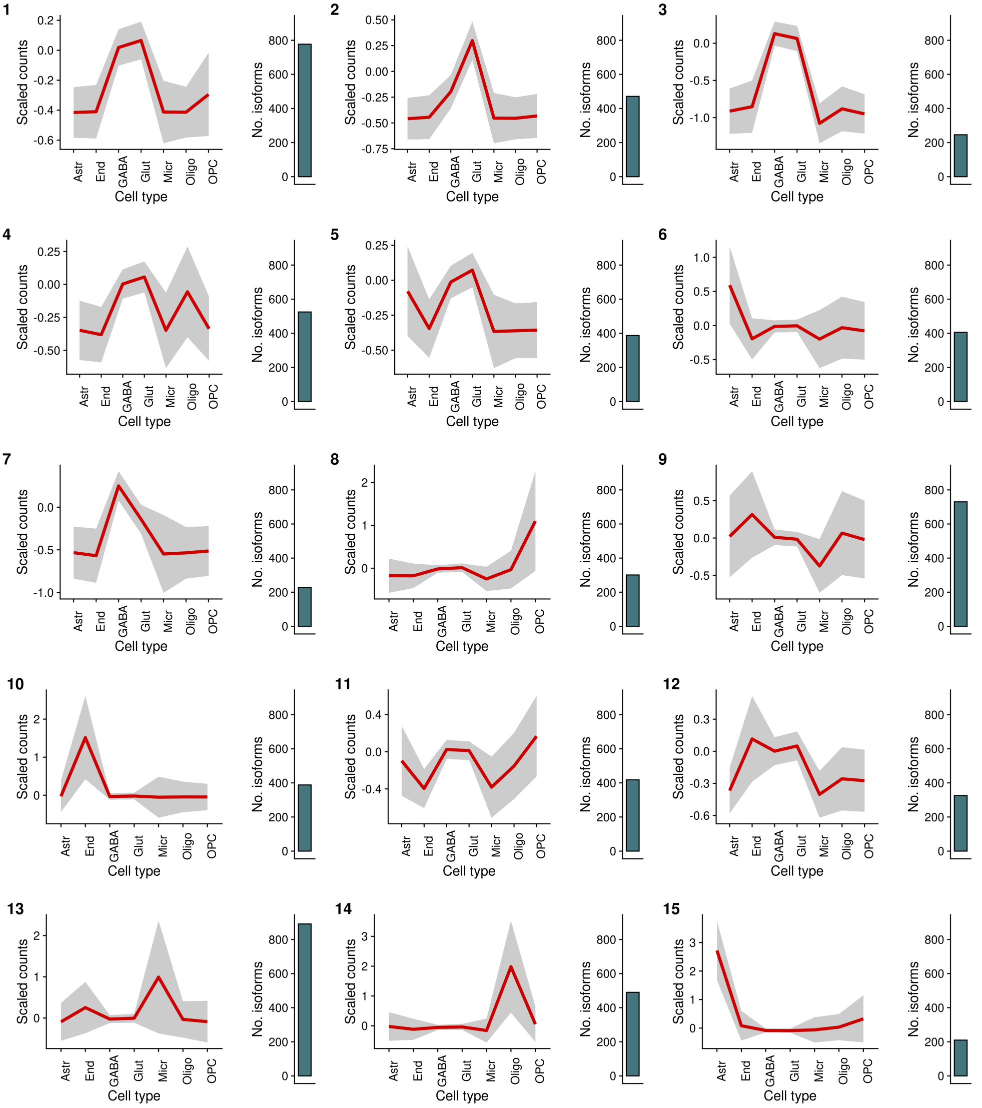


Figure 5.4: Clusters generated after applying the acorde clustering pipeline to the mouse neural dataset. Numbers 1-15 indicate cluster labels. For each labelled cluster, the summarized expression profile is showed on the left. Cell-level mean expression (scaled) is computed for all transcripts and then aggregated as the global cell type mean, represented by the red line. Gray area corresponds to cell type mean±standard deviation. On the right, the barplot indicates the absolute cluster size.

5.3.2 Validation of percentile correlations on simulated data

Building on studies reporting the poor performance of popular correlation metrics in single-cell data, authors have attempted the implementation of sparsity-aware measurements [224, 232] and reported the potential of other alternatives to compute similarity, such as proportionality metrics [220]. Here, we present an interpretable, scalable and biology-aware alternative for single-cell co-expression studies. However, to better understand the performance of percentile correlation in comparison to extant correlation metrics, it was compared to Pearson, Spearman and zero-inflated Kendall correlations [224] as well as to proportionality metric rho (ρ) [225] using simulated data.

Given that, at the time when this study was carried out, there were no scRNA-seq data simulators that could generate transcript co-expression patterns, a simulation strategy was designed (figure 5.5.a, section 5.2.4) in order to generate an appropriate validation framework for our metric. Briefly, we first applied SymSim [223] to simulate a single-cell RNA-seq dataset (8 cell types, 1,000 cells and 8,000 transcripts, 5.5.b) and used the simulated expression values to artificially create 3,000 synthetic transcripts showing 15 different expression profiles across the 8 cell types. In order to achieve this, simulated features were ranked by mean expression within each cell type, losing connectivity between features across cell types (figure 5.5.a, section 5.2.4). Then, a number of high and low-expression vectors were selected in each cell type to generate the desired patterns. As a result, the simulated dataset contained 15 clusters with distinct expression profiles (figure 5.5.c) while preserving the original cell type structure generated by SymSim (Figure 5.5.d). Among them, clusters 1 to 5, 6 to 10 and 11 to 15 included transcripts showing high expression in one, two and three cell types, respectively, gradually increasing the complexity of the simulated patterns (figure 5.5.c). After cluster filtering (section 5.2.4), 1,790 synthetic transcripts remained distributed across the 15 simulated clusters in groups ranging from 180 to 60 transcripts (figure 5.6.a).

In order to evaluate how well the 5 correlation methods recapitulated the simulated patterns, each of the metrics was computed for all synthetic transcript pairs in each simulated cluster (figure 5.6.b). Among them, percentile correlation consistently yielded the best proportion of high within-cluster correlations followed by ρ. However, rather counter-intuitively, ρ had only an average performance when low-complexity patterns were provided, with less than 20% output proportionality values $>$0.8 within clusters 1-5. Strikingly, zero-inflated Kendall correlation, a single cell-tailored metric, failed to recapitulate the simulated co-expression profiles and showed a considerably lower proportion of high correlations within the simulated clusters than Pearson and Spearman correlations. To better assess the ability of each metric to discriminate true from spurious co-expression, pairwise correlations for isoforms within (intra-cluster) and between (inter-cluster) clusters were compared. Even though results showed overall good separation between pairs from the same cluster, percentile correlation was the only metric to provide a complete lack of overlap between inter and intra-cluster correlation distributions (figure 5.6.c). As a result of this evaluation, we can confidently assume

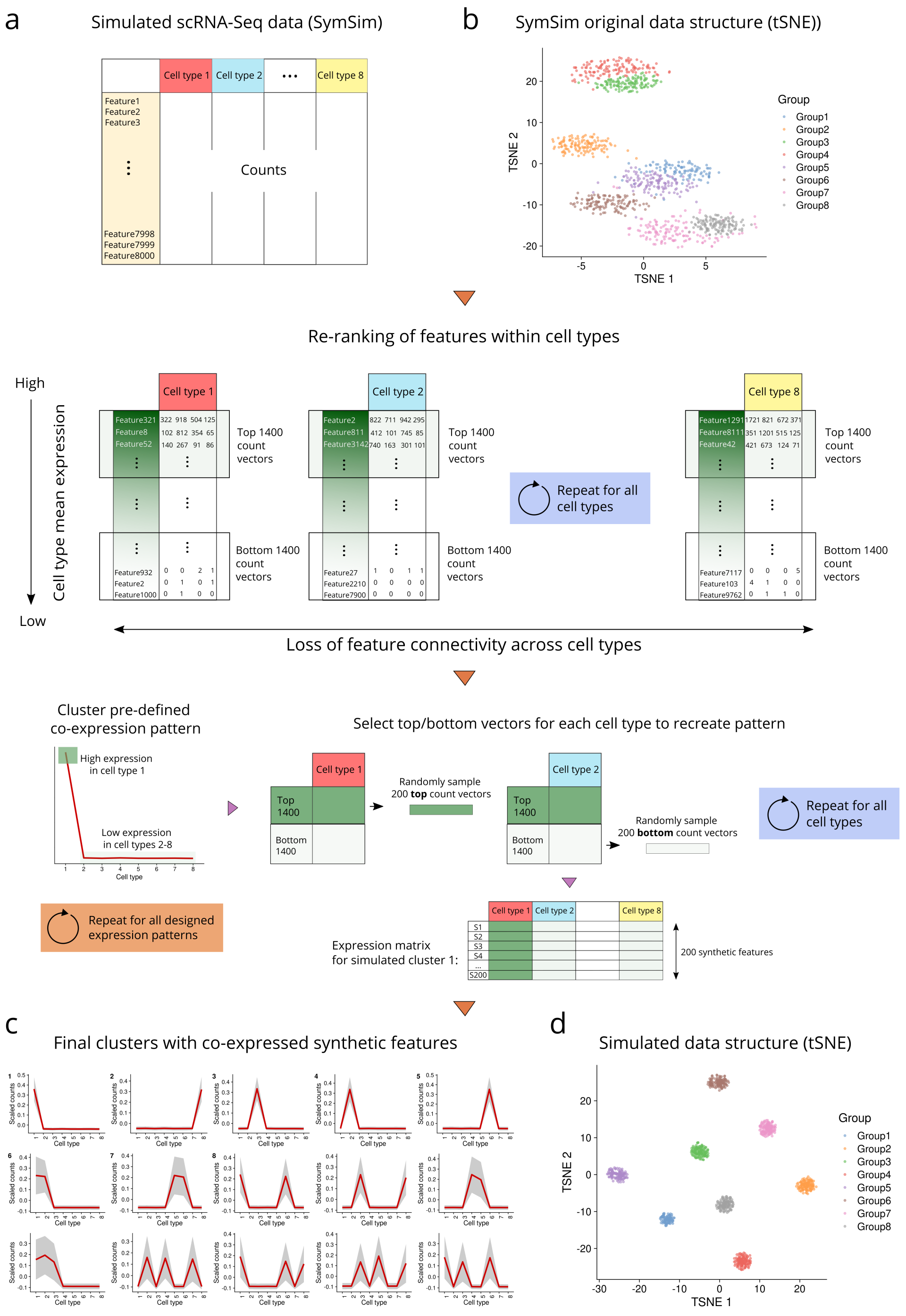

a
Simulated scRNA-Seq data (SymSim)
Cell type 1
Cell type 2
• • •
Cell type 8
Feature1
Feature2
Feature3
⋮
Feature7998
Feature7999
Feature8000
Counts
b
SymSim original data structure (tSNE))
TSNE 2
TSNE 1
Group
Group1
Group2
Group3
Group4
Group5
Group6
Group7
Group8
Re-ranking of features within cell types
High
Cell type mean expression
Low
Cell type 1
Feature321 322 918 504 125
Feature8 102 812 354 65
Feature52 140 267 91 86
Top 1400 count vectors
Feature932 0 0 2 1
Feature2 0 1 0 1
Feature1000 0 1 0 0
Bottom 1400 count vectors
Cell type 2
Feature2 822 711 942 295
Feature811 412 101 745 85
Feature3142 740 163 301 101
Top 1400 count vectors
Feature27 1 0 1 1
Feature2210 0 0 0 0
Feature7900 0 0 0 0
Bottom 1400 count vectors
Repeat for all cell types
Cell type 8
Feature1291 1721 821 672 371
Feature8111 351 1201 515 125
Feature42 421 673 124 71
Top 1400 count vectors
Feature7117 0 0 0 5
Feature103 4 1 0 0
Feature9762 0 1 1 0
Bottom 1400 count vectors
Loss of feature connectivity across cell types
Cluster pre-defined co-expression pattern
High expression in cell type 1
Low expression in cell types 2-8
Cell type
Repeat for all designed expression patterns
Select top/bottom vectors for each cell type to recreate pattern
Cell type 1
Top 1400
Bottom 1400
Randomly sample 200 top count vectors
Cell type 2
Top 1400
Bottom 1400
Randomly sample 200 bottom count vectors
Repeat for all cell types
Cell type 1
Cell type 2
Cell type 8
S1
S2
S3
S4
...
S200
Expression matrix for simulated cluster 1:
200 synthetic features
c
Final clusters with co-expressed synthetic features
1
2
3
4
5
6
7
8
Scaled counts
Cell type
d
Simulated data structure (tSNE)
TSNE 2
TSNE 1
Group
Group1
Group2
Group3
Group4
Group5
Group6
Group7
Group8

Figure 5.5: Simulation of single-cell co-expression patterns: workflow and results. **a** Co-expression simulation using SymSim-generated scRNA-seq data (shown in (b)). **b** Cell type structure of SymSim-simulated scRNA-Seq data depicted using t-SNE. **c** Expression profile of the 15 simulated clusters obtained after following the simulation workflow depicted in (a). Cell-level mean expression (scaled) is computed for all simulated transcripts in the cluster and then aggregated as the global cell type mean, represented by the red line. Grey area corresponds to cell type mean ± standard deviation. **d** Cell type structure after performing co-expression pattern simulation on the SymSim counts (t-SNE).

that percentile correlations are useful to detect co-variation patterns, yielding overall higher correlation values than all other considered metrics (figure 5.6.b) and correctly discriminating true correlated and uncorrelated transcript pairs (figure 5.6.c).

Next, we compared the ability of each co-expression metric to inform clustering and group transcripts with similar expression patterns. To achieve this, we run our clustering pipeline on the simulated isoforms using the 5 metrics as distance. To enable benchmarking, clustering was automated to generate a total number of 15 calculated clusters (see section 5.2.5). In order to evaluate which metric worked best to detect co-expressed transcript groups, we considered internal correlations between the transcripts in the calculated clusters. We observed that ρ and percentile-generated clusters, unlike the remaining co-expression metrics, presented consistently high levels of internal correlation (figure 5.7.a). Notably, the distribution of correlation values obtained using percentiles was the most robust among the five metrics (figure 5.7.b). We next assessed how well the clusters generated using each correlation metric (i.e. calculated clusters) recapitulated the simulated clusters. Calculated and simulated clusters were paired based on the similarities between their mean cluster profiles (section 5.2.5) and the Jaccard

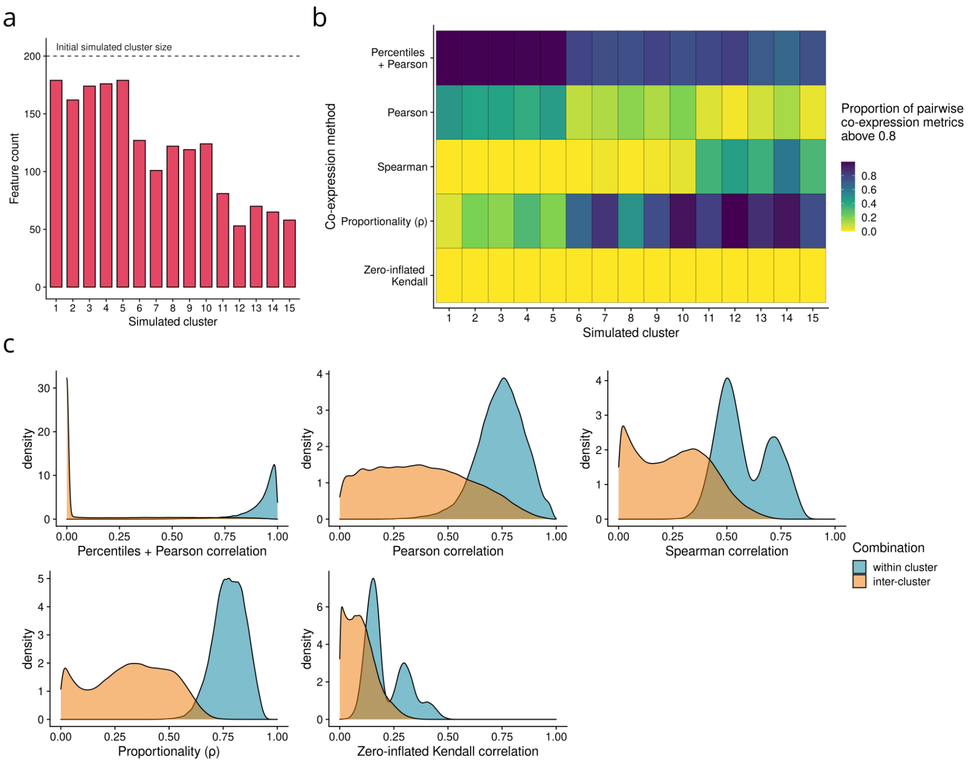


Figure 5.6: Simulated data characterization. **a** Number of transcripts in simulated clusters, calculated after filtering. **b** Heatmap representing the proportion of high pairwise co-expression values (>0.8) obtained within each simulated cluster. Darker colors indicate successful recapitulation of simulated co-expression relationships by the evaluated co-expression metrics. **c** Density distributions obtained when computing pairwise correlations between all possible within-cluster and between-cluster synthetic feature pairs, considering the simulated clusters.

index (JI) was computed for each simulated-calculated pair to measure the agreement in transcript assignment (figure 5.7.c). Interestingly, results were highly heterogeneous for most methods: even though a number of simulated co-expression groups were easily detected by most metrics, no method was able to fully recapitulate the simulated clusters, with ρ proportionality, Pearson and percentile correlations being the most accurate (figure 5.7.c).

Zero-inflated Kendall and Spearman correlations, on the other hand, showed consistently low agreement with the simulated transcript groups.

Finally, we considered the number of transcripts that remained unclustered (figure 5.7.d) before and after re-clustering unassigned transcripts (cluster expansion step, see section 5.2.3). Pearson correlation provided successful cluster assignment for practically all transcripts in the simulated dataset, especially when incorporating percentiles (Pearson: $\sim$10% unclustered before expansion, $\sim$1% after; percentile: $\sim$4% unclustered before expansion, 0% after), whilst the rest of metrics performed significantly worse, leaving 20-30% of transcripts unassigned even after cluster expansion, with proportionality ($\sim$30% unclustered before expansion, $\sim$25% after) being the less optimal. Altogether, even though ρ demonstrated good performance in many aspects of clustering, including intra-cluster correlation and agreement with the simulated clustering, it was outperformed by percentile correlation when globally considering all evaluated parameters (figure 5.7.e). In addition to the fact that ρ failed to control for unassigned transcripts, computing means and standard deviations of Jaccard indices across simulated-calculated pairs showed percentile and Pearson correlations as the most consistently accurate methods. All in all, our synthetic data evaluations showed that the percentile correlation approach performed well -and more consistently than ρ proportionality- in all the evaluated features, and visibly captured co-expression better than both traditional and zero inflation-aware correlation metrics.

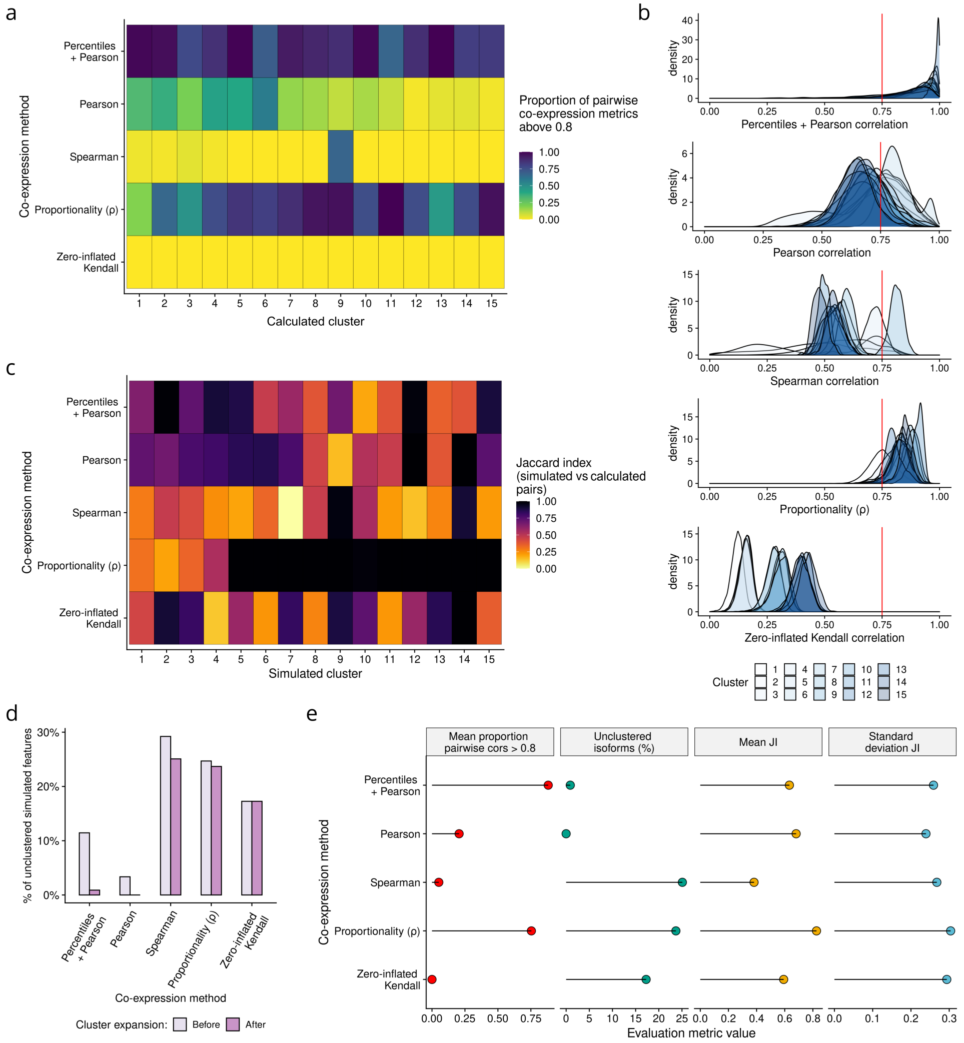


Figure 5.7: Comparative evaluation of percentile correlation on simulated clusters. **a** Proportion of co-expression values above 0.8 in each calculated cluster, obtained using the different evaluated methods as a distance metric for clustering. **b** Density distributions of pairwise correlation values computed between isoforms in the calculated clusters. **c** Jaccard index of simulated vs calculated clusters obtained with each evaluated co-expression method. Simulated clusters were paired with one calculated cluster based on mean profile similarity and synthetic transcript IDs included in each of the paired clusters compared. **d** Percentage of unclustered isoforms generated by each co-expression method. Results are shown before and after re-assigning unclustered isoforms by co-expression with the mean profile of extant clusters (i.e. cluster expansion). **e** Evaluation metric overview. Metrics are specified in the grid headers. x-axis shows values of the different metrics, y-axis displays evaluated co-expression methods.

5.3.3 Co-differential isoform usage analysis of single-cell isoform expression

Isoform clusters represent groups of alternative transcripts that are co-expressed at the cell type level. However, clustering results alone did not provide information on the iso-transcriptome properties associated with splicing regulation. To facilitate interpretation of the isoform clustering results, we first defined genes with Differential Isoform Usage (DIU) as those whose isoforms were assigned to different clusters (figure 5.8.a). Therefore, DIU genes will not only have two or more isoforms with significant changes in expression across cell types, but also simultaneously undergo cell type-dependent post-transcriptional regulation. In the present dataset, 23% of total analyzed genes (2,017 out of 8,814) and 75% of genes with clustered isoforms (total = 2,696) were positive for DIU, involving 5,278 clustered isoforms.

In order to extend this concept to the study of isoform co-expression patterns, we defined co-Differential Isoform Usage (coDIU) genes as those showing coordinated isoform usage across cell types. Specifically, two or more genes were considered to be coDIU if their isoforms had been assigned to the same clusters (figure 5.8.b, section 5.2.6). This resulted in the definition of an isoform co-splicing network, where nodes were clusters of correlated isoforms and edges represented coDIU genes (figure 5.9.a, 5.2.6). To ensure the reliability of the detected coDIU patterns, a generalized linear model was fitted for every pair of coDIU genes, subsequently selecting pairs that showed significant cluster-dependent expression across cell types and no significant changes in expression when only accounting for gene-level expression

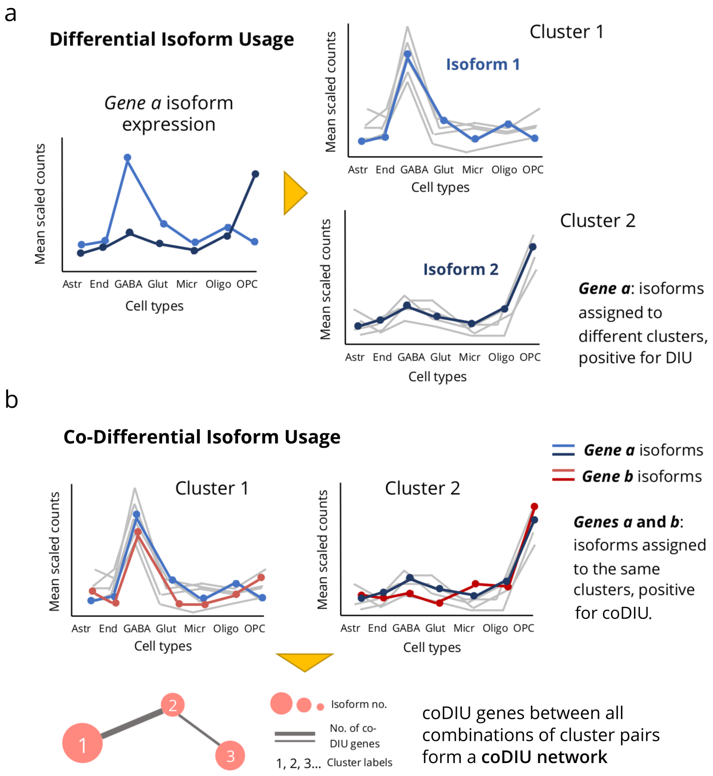

a
Differential Isoform Usage
Gene a isoform expression
Mean scaled counts
Astr End GABA Glut Micr Oligo OPC
Cell types
Cluster 1
Isoform 1
Cluster 2
Isoform 2
Gene a: isoforms assigned to different clusters, positive for DIU
b
Co-Differential Isoform Usage
Cluster 1
Cluster 2
Gene a isoforms
Gene b isoforms
Genes a* and *b: isoforms assigned to the same clusters, positive for coDIU.
1
2
3
Isoform no.
No. of co-DIU genes
1, 2, 3... Cluster labels
coDIU genes between all combinations of cluster pairs form a **coDIU network**

Figure 5.8: Differential Isoform Usage (DIU) and Co-Usage (coDIU). a Cluster-based definition of DIU across multiple cell types. DIU genes have at least two isoforms assigned to different clusters, indicating a differential isoform selection pattern across the different cell types. **b** Definition of co-Differential Isoform Usage (coDIU) using clusters. CoDIU genes have multiple isoforms assigned to the same clusters, establishing across-cell type co-expression relationships for at least two of their isoforms.

(see section 5.2.6). CoDIU genes must therefore present cell type-dependent co-expression of at least two isoforms, represented by cluster assignment matches, but should not be co-expressed when only gene expression is considered. Using this strategy, 1,784 genes with at least one significant coDIU partner (cluster×cell-type FDR<0.05, gene×cell-type FDR>0.05) were detected, involving 5,274 co-expressed isoforms. The number of coDIU genes sharing isoforms across each cluster pair was variable, although it rose up to >130 for highly connected clusters (figure 5.9.a).

The coDIU network was next interrogated to find patterns underlying the splicing coordination signal detected by the acorde pipeline. First, in order to measure whether coDIU generated strong or subtle variations in isoform selection across cell types, we investigated the association of coDIU to single or multiple cell type isoform switching events. Single isoform switching events involve clusters with patterns that are similar across all cell types except one, hence being more likely to yield high between-cluster correlations. To measure the strength of the isoform selection switches generated by coDIU, Pearson correlations between the average expression profiles of the clusters were computed (figure 5.9.b). A positive association between low switch complexity and high levels of coDIU would therefore translate into highly correlated cluster pairs being among the ones with the highest number of coDIU genes. Interestingly, the number of coDIU genes between isoform clusters was found to be linked to cluster size, as expected, but showed no direct relationship with the similarities between the expression profiles of the clusters (figure 5.9.b). The detection of coDIU genes with highly different isoform usage patterns hence suggested that coordinated isoform expression may be able to produce strong cell type-level shifts in isoform selection.

We next evaluated the cell type-level relationships in the isoform co-expression network, namely the occurrence of coDIU across all possible pairs of cell types in our data. To capture this effect, we selected all clusters representing high expression in a given cell type, successively pairing them with clusters that similarly represented each of the remaining cell types. Next, for any two cell types, the number of coDIU genes between all considered pairs of clusters was aggregated, obtaining an estimate of cross-cell type coDIU frequency. Although co-splicing could potentially occur between any combination of cell types, the results of this analysis showed that a high proportion of coDIU interactions were detected when the isoforms involved had high expression in one of the two neural cell types, i.e. GABA-ergic and glutamatergic neurons (figure 5.9.c). This was partially explained by the fact that some of the clusters with neuron expression were among the largest generated by the acorde pipeline (figure 5.9.a). However, a complementary explanation was that the central role of neurons in the tissue under study (i.e. primary visual cortex) might situate co-splicing at the core of neural function regulation, as well as the modulation of its interaction with glial cell types.

5.3.4 Functional analysis of isoforms within a coDIU network

We next set out to investigate the functional implications of our isoform co-expression network. Since, with a few exceptions [121, 192], splicing analysis tools rarely integrate functional information, we annotated the long

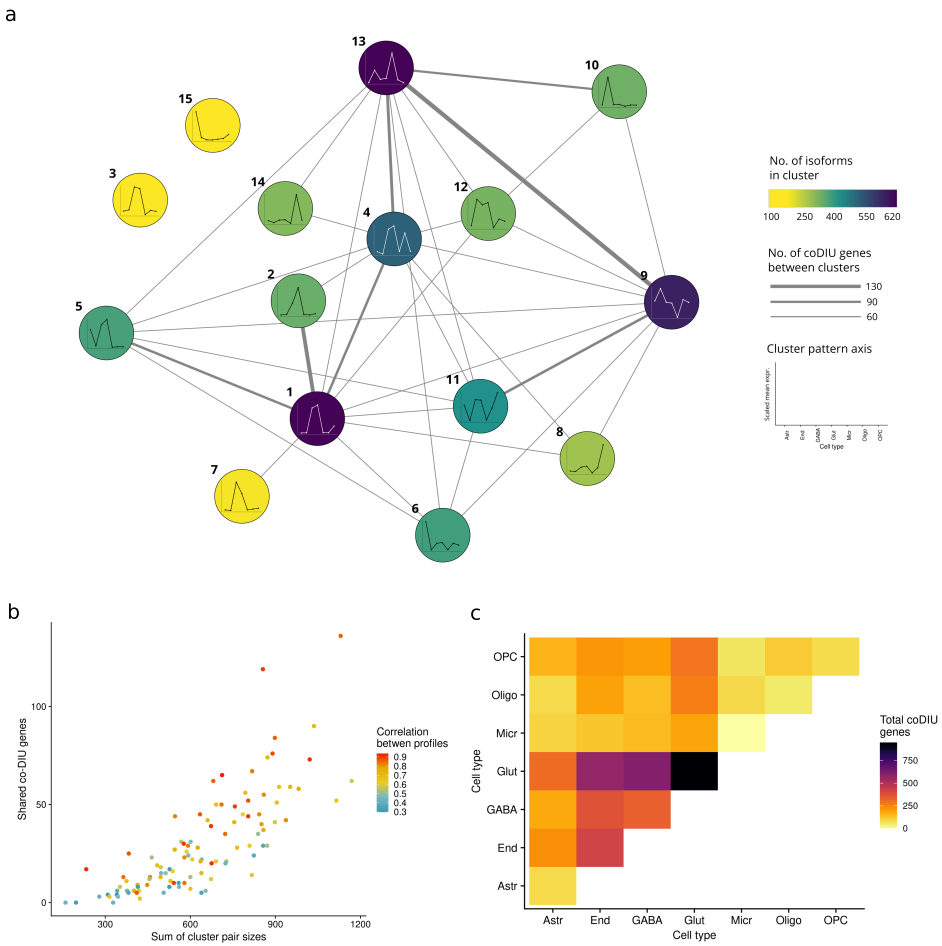


Figure 5.9: Characterization of CoDIU genes. a coDIU network. Nodes represent clusters and depict their mean expression profile across cell types. Node color represents cluster size, edge width represents number of coDIU genes detected between each pair of clusters. coDIU genes were considered if they had at least one significant isoform co-expression pattern with one other gene. **b** Evaluation of cluster profile similarity and size as a function of the number of coDIU genes detected. X-axis corresponds to the sum of isoforms in each possible pair of clusters. Y-axis contains the number of coDIU genes between the pair. Dot color represents the correlation between the mean expression profiles of each pair of clusters. **c** Cell type-level coDIU patterns. For each pair of cell types represented in x and y-axis, heatmap color corresponds to the total number of genes found to be co-DIU between them.

read-defined transcripts using IsoAnnotLite (https://isoannot.tappas.org/isoannot-lite/). The resulting functional annotation included both transcript and protein-level motifs, sites and domains, as well as non-positional, gene-level features such as Gene Ontology (GO) terms. A detailed description of the annotation process and a comprehensive list of functional categories and source databases is available in sections 4.2.3 and 4.3.2 of chapter 4.

First, we analyzed which biological processes and gene functions were potentially controlled by DIU (AS-regulated) and coDIU (co-regulated) mechanisms, that is, which gene functionalities were overrepresented in the DIU and coDIU sets. In order to discriminate the functional properties of AS-regulated genes from those showing no cell type specificity in isoform expression, we performed a functional enrichment test for DIU genes vs genes with DE isoforms, which were used as the background (figure 5.10, see section 5.2.7). Interestingly, DIU genes showed significant enrichment (FDR<0.05) in GO terms associated with gene expression regulation, including general mechanisms (*nucleic acid metabolic process*) and processes related to both DNA (*DNA metabolic process*) and RNA metabolism (*RNA metabolism*). In addition, genes annotated as participating in protein-complex mechanisms required for these processes (*protein-containing complex*) were found to be significantly overrepresented in the DIU group. Remarkably, functional enrichment revealed that DIU genes were enriched in binding sites for miR-412-3p (figure 5.10). Even though extant literature includes no functional roles for this miRNA in the brain, miR-412-3p has been found to interact with Mbnl1-AS16 [238], a long non-coding RNA that is also an antisense isoform of Mbnl1, which is an important splicing regulator in the neural context [239–241]. We thus speculate that differential inclusion of miR-412-3p binding sites in the isoforms of coDIU genes might be related to additional regulatory roles of this miRNA in neural cell types.

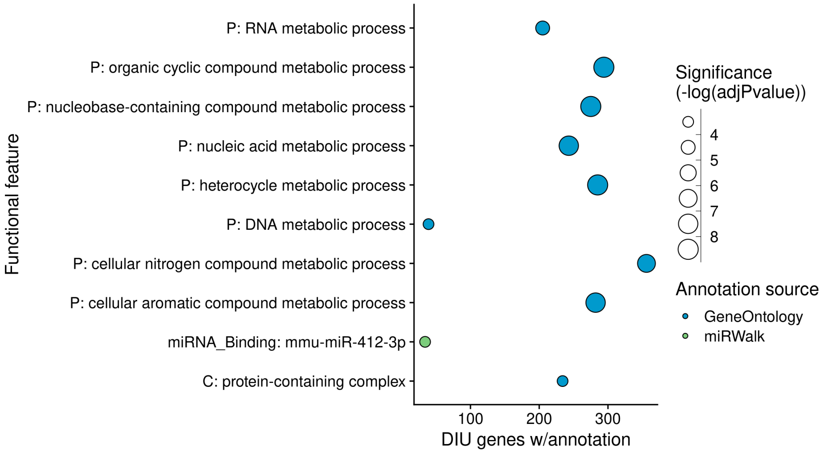


Figure 5.10: Functional enrichment analysis results for DIU genes vs genes with DE isoforms. x-axis indicates the total number of test genes (i.e. DIU) including the tested annotation feature. y-axis shows functional features. Dot color represents functional category (i.e. annotation source database), dot size represents the significance (-log(adjusted p-value)) obtained in the Fisher's Exact test.

Next, in order to investigate the cellular processes where coDIU could have a relevant regulatory role, we compared the proportion of coDIU and DIU genes annotated for each functional feature in the transcriptome using a partially-overlapping samples z test [229] (described in section 5.2.7). In total, 91 positional functional features and 59 GO terms were found to be annotated in a significantly higher proportion in coDIU vs DIU genes (FDR<0.05, Appendix II.2). Given the extensive list of significant features obtained, we focused on the most relevant set of features for visualization and interpretation purposes (figures 5.11.a-b). Full results, however, are available in Appendix II.2.

First, to remove redundant functionalities, GO terms were filtered by semantic similarity using Revigo [230] (see section 5.2.7, figure 5.11.a), resulting in 26 unique terms. Among them, and similarly to DIU genes, coDIU genes showed significant enrichment (FDR<0.05) in functionalities related to specific aspects of transcriptional regulation (*regulation of nitrogen compound metabolic process, regulation of transcription by RNA polymerase II*, figure 5.11.a, Appendix II.2). However, genes that were positive for coDIU were also significantly associated with signaling mechanisms (*protein kinase activity, protein phosphorylation, signal transduction*), membrane transport and cell-to-cell communication (*transmembrane signaling receptor activity, cell communication, regulation of response to stimulus*). While no synaptic terms were found to be significant in this analysis, the association of coDIU genes to some of these GO terms, i.e. *transmembrane signaling receptor activity* (GO:0004888) and *signal transduction* (GO:0007165), may point towards a connection between AS co-regulation, neural signaling pathways and, ultimately, the modulation of synaptic activity. These results therefore open up an avenue for in-depth characterization of the mechanistic role that isoform coordination plays within neuron-specific biological processes.

Remarkably, coDIU genes showed additional enrichment for post-transcriptional processes and functionalities such as RNA binding and translation. This result links genes participating in RNA metabolism with the coordination of AS, and suggests that the co-expression of alternative isoforms may contribute to the fine-tuning of post-transcriptional regulatory processes. Regarding positional functional features, coDIU genes presented a significantly higher proportion of miRNA binding, 3'UTR and upstream Open Reading Frame (uORF) motifs (FDR<0.05, figure 5.11.b), as well as several predicted protein elements such as post-translational modifications (PTMs), nuclear localization signals (NLS) and transmembrane domains. These motifs and domains exert a broad variety of roles, including interactions with regulators (e.g. 3'UTR motifs), mRNA turnover control (e.g. uORFs) and protein-specific localization (e.g. NLS, transmembrane), indicating the potential of coDIU to create functional synergies via the co-inclusion of specific domains and motifs in an isoform-specific manner.

To further explore the potential of domain-including (or excluding) isoform co-expression across neural cell types, a Functional Diversity Analysis (FDA, section 5.2.7) was next performed. FDA is part of the tappAS framework [121], and identifies functionally varying genes, i.e. genes expressing transcript variants with differences in the inclusion of functional features (see full description in section 5.2.7). FDA can be evaluated from a presence/absence standpoint (i.e. AS completely removes a feature), or by detecting variation in the transcriptomic positions defining the feature. Using both criteria, the diversity in transcript-level functional features between DIU, coDIU genes and genes with more than one DE isoform was evaluated for each of the functional categories provided by IsoAnnotLite (figure 5.12, Appendix II.3). Interestingly, the percentage of varying genes was shown to increase with isoform expression complexity, with coDIU resulting in the largest amount

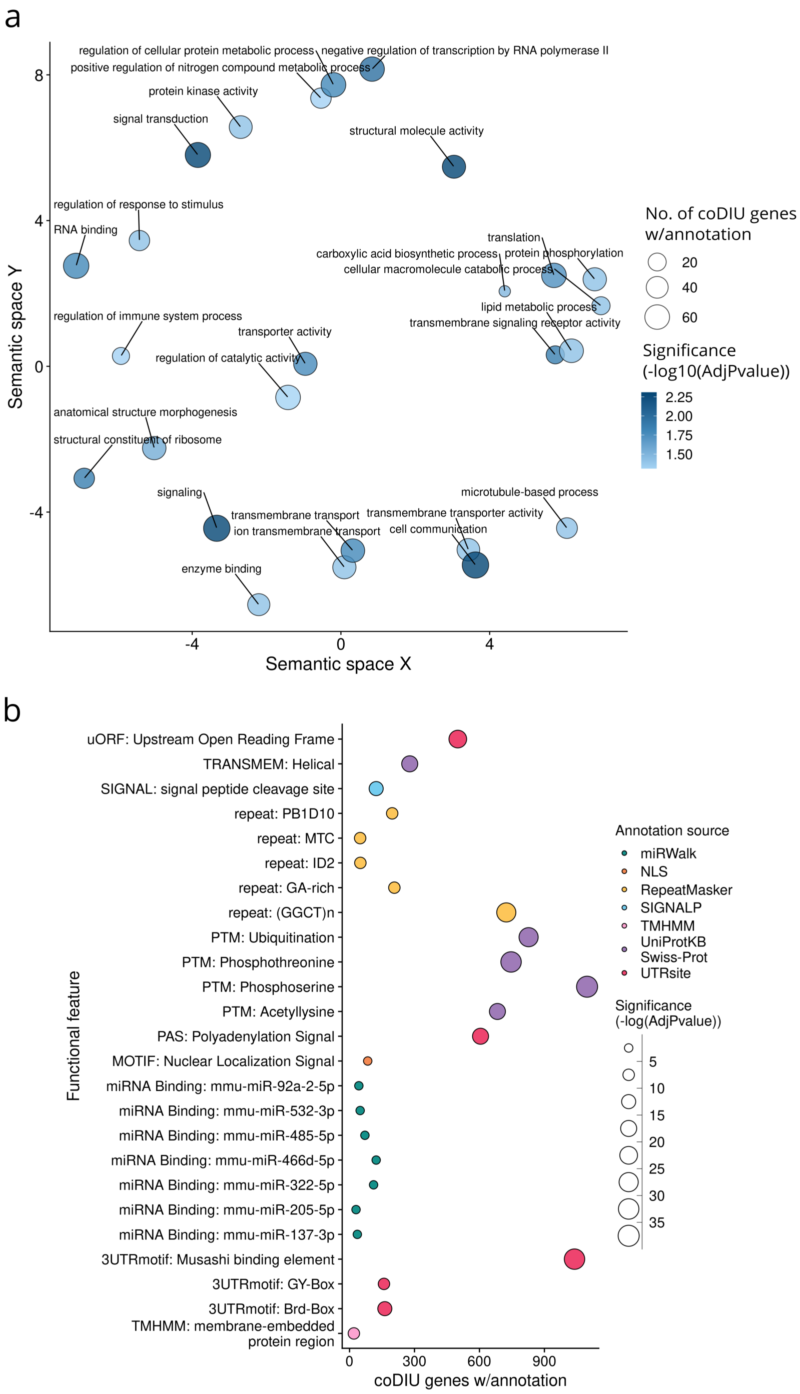

a
regulation of cellular protein metabolic process
negative regulation of transcription by RNA polymerase II
positive regulation of nitrogen compound metabolic process
protein kinase activity
signal transduction
structural molecule activity
regulation of response to stimulus
RNA binding
translation
carboxylic acid biosynthetic process
protein phosphorylation
cellular macromolecule catabolic process
lipid metabolic process
transmembrane signaling receptor activity
regulation of immune system process
transporter activity
regulation of catalytic activity
anatomical structure morphogenesis
structural constituent of ribosome
microtubule-based process
signaling
transmembrane transport
transmembrane transporter activity
ion transmembrane transport
cell communication
enzyme binding
Semantic space Y
8
4
0
-4
-4
0
4
Semantic space X
No. of coDIU genes w/annotation
20
40
60
Significance (-log10(AdjPvalue))
2.25
2.00
1.75
1.50
b
uORF: Upstream Open Reading Frame
TRANSMEM: Helical
SIGNAL: signal peptide cleavage site
repeat: PB1D10
repeat: MTC
repeat: ID2
repeat: GA-rich
repeat: (GGCT)n
PTM: Ubiquitination
PTM: Phosphothreonine
PTM: Phosphoserine
PTM: Acetyllysine
PAS: Polyadenylation Signal
MOTIF: Nuclear Localization Signal
miRNA Binding: mmu-miR-92a-2-5p
miRNA Binding: mmu-miR-532-3p
miRNA Binding: mmu-miR-485-5p
miRNA Binding: mmu-miR-466d-5p
miRNA Binding: mmu-miR-322-5p
miRNA Binding: mmu-miR-205-5p
miRNA Binding: mmu-miR-137-3p
3UTRmotif: Musashi binding element
3UTRmotif: GY-Box
3UTRmotif: Brd-Box
TMHMM: membrane-embedded protein region
Functional feature
0
300
600
900
coDIU genes w/annotation
Annotation source
miRWalk
NLS
RepeatMasker
SIGNALP
TMHMM
UniProtKB Swiss-Prot
UTRsite
Significance (-log(AdjPvalue))
5
10
15
20
25
30
35

Figure 5.11: Functional enrichment analysis results for coDIU vs DIU genes. **a** Gene Ontology (GO) Functional Enrichment results for co-Differential isoform Usage (coDIU) vs genes with Differential Isoform Usage (DIU) of isoforms. Only terms obtained after Revigo filtering are shown. x and y-axis indicate semantic similarity, as defined by Revigo. Dot size represents total coDIU genes annotated for each GO term. Dot color represents significance (-log10(adjusted p-value)). **b** Functional Enrichment of positional features for coDIU vs DIU genes. x-axis indicates the total number of coDIU genes including the tested annotation feature, y-axis shows functional features. Dot color represents the functional category and dot size represents -log(adjusted p-value).

of feature inclusion diversity in virtually all protein and transcript feature categories. Note that these results, including absolute values for all categories, are fully available at Appendix II.3. To better measure this effect, we compared the percentages of variation for all pairwise combinations of the three gene sets (paired samples t-test, see section 5.2.7) and confirmed the observed trend, regardless of the variability criteria employed (i.e. position/p-resence). In particular, even though nearly all comparisons were significant, coDIU resulted in the most significant increase in feature variation (coDIU vs DE isoform genes p-value: presence = 1.14e-03, position = 8.92e-04; coDIU vs DIU genes p-value: presence = NS, position = 2.04e-03). This result seems to indicate that alternative isoforms engaging in co-expression relationships tend to alter their functional properties significantly more often than other transcripts, thereby coupling alternative splicing and isoform co-expression with changes in the functional potential of the resulting transcripts and proteins.

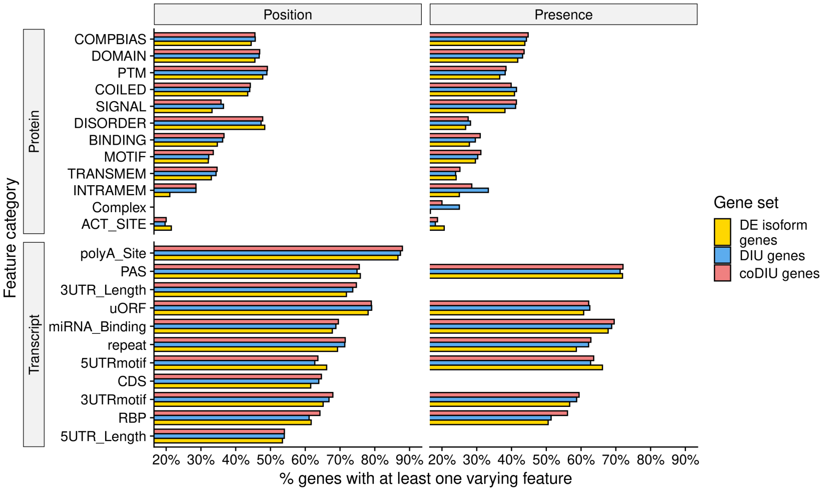


Figure 5.12: Functional Diversity Analysis (FDA) for DE isoform, DIU and coDIU genes. y-axis shows transcript and protein functional categories. x-axis shows the percentage of genes including at least one feature annotation from each of the categories that are detected as functionally varying. Both FDA criteria are shown (position: left grid column, presence: right grid column). CoDIU genes show the largest level of functional variation.

5.3.5 Functional insights on neuron-oligodendrocyte isoform co-expression patterns

To further understand the relationship between cell-type identity, isoform co-expression and the functional properties of coDIU genes, we searched the coDIU network for cluster groups representing biologically-related isoform switches between neural cell types. Namely, we focused on a set of 118 coDIU genes (figure 5.13.a) containing isoforms with higher relative expression in oligodendrocytes (cluster 14), neurons (GABA and Glutamatergic neuron cell types, cluster 1) or both (cluster 4) and analyzed isoform-associated functional variability using FDA (figure 5.13.b). For this set of alternative isoforms, 3'UTR length showed the highest variation rate among annotated transcript-level functional categories (varying in 70% genes, figure 5.13.b). Moreover, we noticed that these changes followed a clear cell type-specific pattern, with the majority of coDIU genes showing higher relative expression of their longest 3'UTR isoforms in neurons (figure 5.13.c) and neural-specific isoforms generally expressing longer 3'UTRs than their oligodendrocyte-expressed counterparts (figure 5.13.d). This neuron-specific pattern of of 3'UTR co-elongation is consistent with the available literature [242, 243], including studies that outline the general role of UTR-based regulation in non-proliferating cells [244].

To verify the regulatory role of UTR elongation, we inspected several related functional categories (repeat regions, miRNA binding, and 3'UTR motifs), using ID-level FDA (see section 5.2.7, figure 5.13.e) to identify functional features associated to UTR length differences between neurons and oligodendrocytes. Regarding the presence of miRNA binding sites, in spite high varying rates, no specific miRNA motif was shared by more than 10% of genes (maximum of 12 out of 118 genes for miR-495-3p), while some repeats, such as (GT)n, were present in 25% of coDIU genes across the three clusters with varying rates >50%. Nevertheless, in the case of 3'UTR motifs, we found that Musashi binding motifs presented inclusion changes in 60% of annotated coDIU genes (figure 5.13.e). The Musashi protein is known to be a neural RNA-binding protein that participates in translation control, regulating cell fate and cell cycle [245, 246]. In line with this, the coDIU network included several genes in which 3'UTR elongation led to neuron-specific co-inclusion of Mushashi binding elements, including kinase-encoding genes

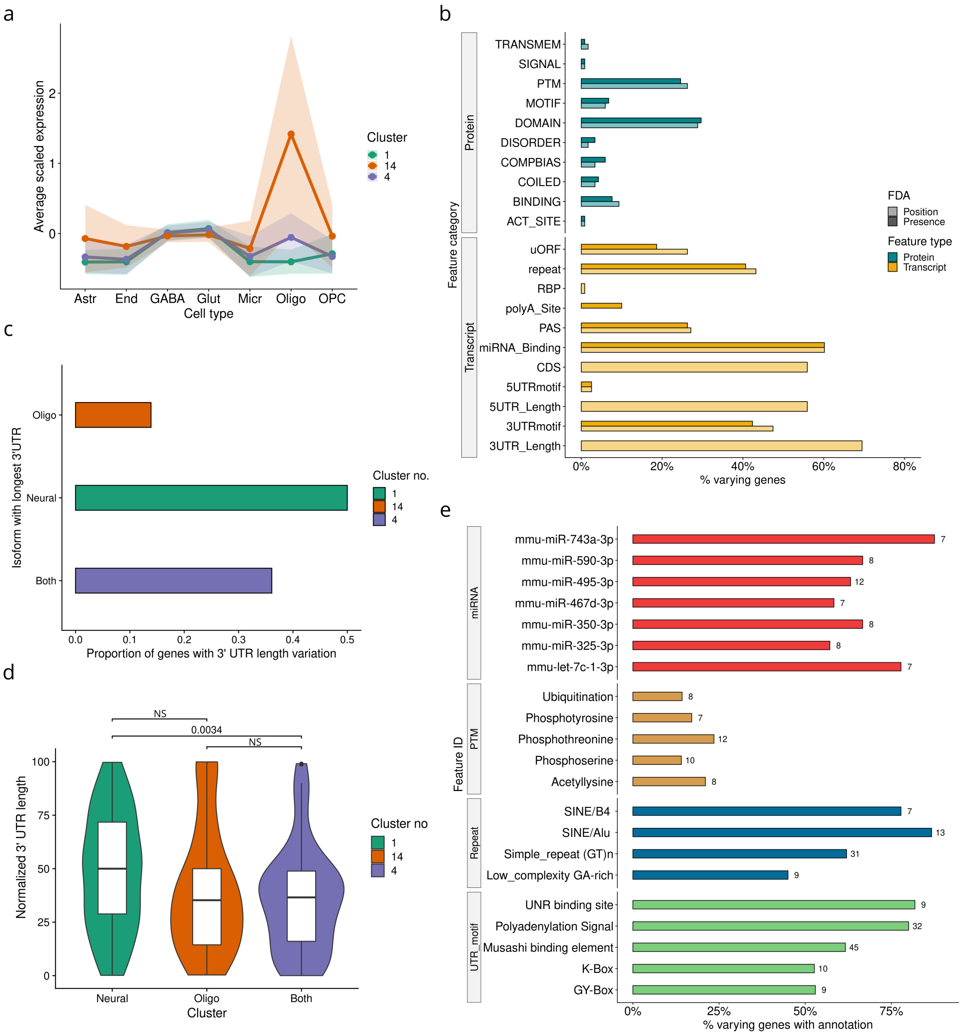

a
Average scaled expression
2
1
0
Astr
End
GABA
Glut
Micr
Oligo
OPC
Cell type
Cluster
1
14
4
b
Feature category
Protein
TRANSMEM
SIGNAL
PTM
MOTIF
DOMAIN
DISORDER
COMPBIAS
COILED
BINDING
ACT_SITE
Transcript
uORF
repeat
RBP
polyA_Site
PAS
miRNA_Binding
CDS
5UTRmotif
5UTR_Length
3UTRmotif
3UTR_Length
0%
20%
40%
60%
80%
% varying genes
FDA
Position
Presence
Feature type
Protein
Transcript
c
Isoform with longest 3'UTR
Oligo
Neural
Both
0.0
0.1
0.2
0.3
0.4
0.5
Proportion of genes with 3' UTR length variation
Cluster no.
1
14
4
d
NS
0.0034
NS
Normalized 3' UTR length
100
75
50
25
0
Neural
Oligo
Both
Cluster
Cluster no
1
14
4
e
Feature ID
miRNA
mmu-miR-743a-3p
7
mmu-miR-590-3p
8
mmu-miR-495-3p
12
mmu-miR-467d-3p
7
mmu-miR-350-3p
8
mmu-miR-325-3p
8
mmu-let-7c-1-3p
7
PTM
Ubiquitination
8
Phosphotyrosine
7
Phosphothreonine
12
Phosphoserine
10
Acetyllysine
8
Repeat
SINE/B4
7
SINE/Alu
13
Simple_repeat (GT)n
31
Low_complexity GA-rich
9
UTR_motif
UNR binding site
9
Polyadenylation Signal
32
Musashi binding element
45
K-Box
10
GY-Box
9
0%
25%
50%
75%
% varying genes with annotation

Figure 5.13: Functional analysis of the 118 coDIU genes detected across neural-oligodendrocyte clusters. a Cell type expression patterns of clusters selected for downstream functional analysis: neural (cluster 1, green), oligodendrocyte (cluster 14, orange) or shared (cluster 4, purple). For each cluster, cell-level mean expression (scaled) is computed for all transcripts, and cell means are aggregated to obtain a global cell type mean, represented by the lines. Colored areas correspond to cell type mean±standard deviation. **b** Functional Diversity Analysis (FDA) results. y-axis: functional annotation categories. x-axis: percentage of genes including at least one varying functional feature. **c** Proportion of coDIU genes with 3'UTR variation across isoforms that have their longest 3'UTR isoform in each of the three analyzed clusters. **d** Violin and boxplots of normalized 3'UTR lengths for isoforms in each of the three neural-oligodendrocyte clusters (n = 177 transcript isoforms from coDIU genes with 3'UTR variability). Normalized lengths are computed by dividing the 3'UTR length of each individual isoform by the sum of 3'UTR lengths of all same-gene isoforms. Violin plots indicate density distributions. Significance levels for the comparison of the three groups are indicated above the corresponding braces (p-value, Wilcoxon's test, two-sided). NS: not significant (p-value>0.05). **e** ID-level FDA results for features in highly-varying functional categories. y-axis shows functional feature categories as boxes and individual features in the axis text. x-axis contains the percentage of genes containing the feature that show functional variation across isoforms. Percentages are shown for the most frequently annotated features in each category. Total coDIU genes with feature are indicated by the bar label.

Ppip5k1 (diphosphoinositol pentakisphosphate kinase 1, figure 5.14.a) and *Prkcz* (protein kinase C zeta type, figure 5.14.b). These results align with the previously shown enrichment of signaling and translation-related genes within the coDIU network (figure 5.11.a) and hints that co-expression of Musashi-binding isoforms may generate 3' UTR binding-mediated changes in the translation of proteins participating in different signaling pathways.

Importantly, the majority of neuron-oligodendrocyte coDIU genes also presented frequent coding region variation (i.e. CDS, figure 5.13.b), revealing that coordinated isoform usage can modify both transcript and protein functional properties. In particular, protein domains (PFAM) and post-translational modifications (PTMs) presented high variation rates (varying in 30% of genes containing the feature, figure 5.13.b) and thus constituted the categories with the most cell type-dependent functional variation. While

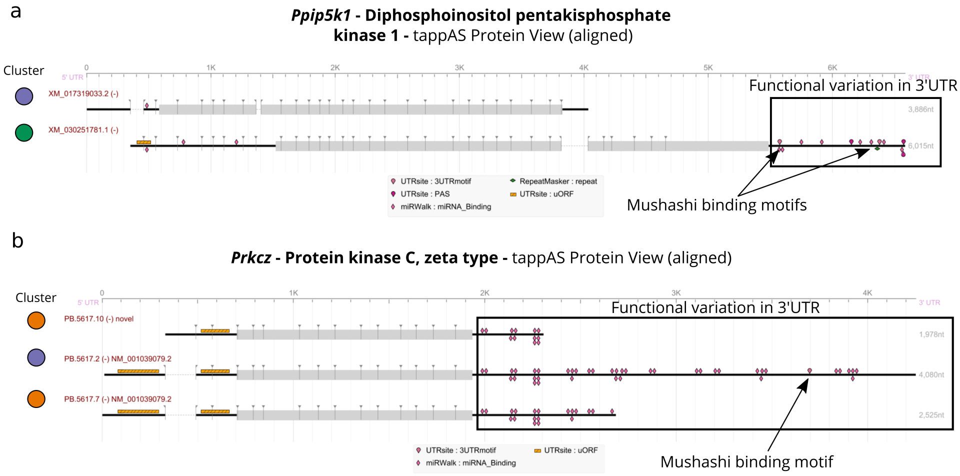


Figure 5.14: tappAS view of transcript functional annotation for **a** *Ppip5k1* and **b** *Prkcz* isoforms. Cluster assignments for each isoform are indicated by dot color (green: GABA-glutamatergic neuron cluster, orange: oligodendrocyte cluster, purple: expression in both oligodendrocytes and neurons).

ID-FDA reported no specific PFAM domains shared among the analyzed gene set (1%, maximum of 2 out of 118), up to 10% of them presented inclusion variation in similar PTMs (12 out of 118) with medium to high variation rates for phosphorylation, acetylation and ubiquitination (figure 5.13.e). However, synergies between PTM and domain inclusion changes could still result in differential functional activities at the protein level. As an example, we found two genes involved in different aspects of RNA metabolism, *Lrif1* (ligand-dependent nuclear receptor interacting factor, figure 5.15.a) and *Stau2* (Staufen RNA binding protein homolog, figure 5.15.b), both of which present cell type-level domain inclusion associated to coordinated changes in the expression of alternative protein isoforms. In humans, *Lrif1* has been shown to interact with a number of nuclear receptors, including retinoic acid receptors, to suppress the ligand-mediated transcriptional activator role of these proteins [247]. This interaction occurs at the N-terminal end, which presents differential inclusion of several protein motifs and domains among *Lrif1* isoforms. Specifically, two *Lrif1* isoforms that are depleted in oligodendrocytes present inclusion of a coiled/disordered region as well as binding and phosphorylation sites (figure 5.15.a), which may be connected to a specific transcriptional regulation role in mouse neuron cells. Also regarding the neural-related functionality of these genes, the rat homolog of *Stau2* is known to have a role in mRNA transport from the nucleus to neuron dendrites [248]. In our system, *Stau2* isoforms upregulated in oligodendrocytes show a C-terminal Staufen domain that is not included in neural-specific isoforms (figure 5.15.b) and is responsible for Staufen dimerization in humans [249]. On the other hand, one of the neuron-expressed isoforms includes an extra N-terminal RNA binding domain. Differences among *Stau2* isoforms may be connected to a dual role for this protein in neurons and glia in which enhanced RNA binding activity is required in neurons, while Staufen dimers may be more likely to form in oligodendrocytes. However, further analyses and validation experiments are required to confirm these findings and reveal additional coordination of domain inclusion changes for genes involved in differential RNA processing between neurons and glial types.

5.3.6 Analysis of coDIU patterns in GABA-ergic neuron subtypes

In order to showcase the applicability of the acorde pipeline and test its performance under high-granularity conditions, we additionally analyzed isoform

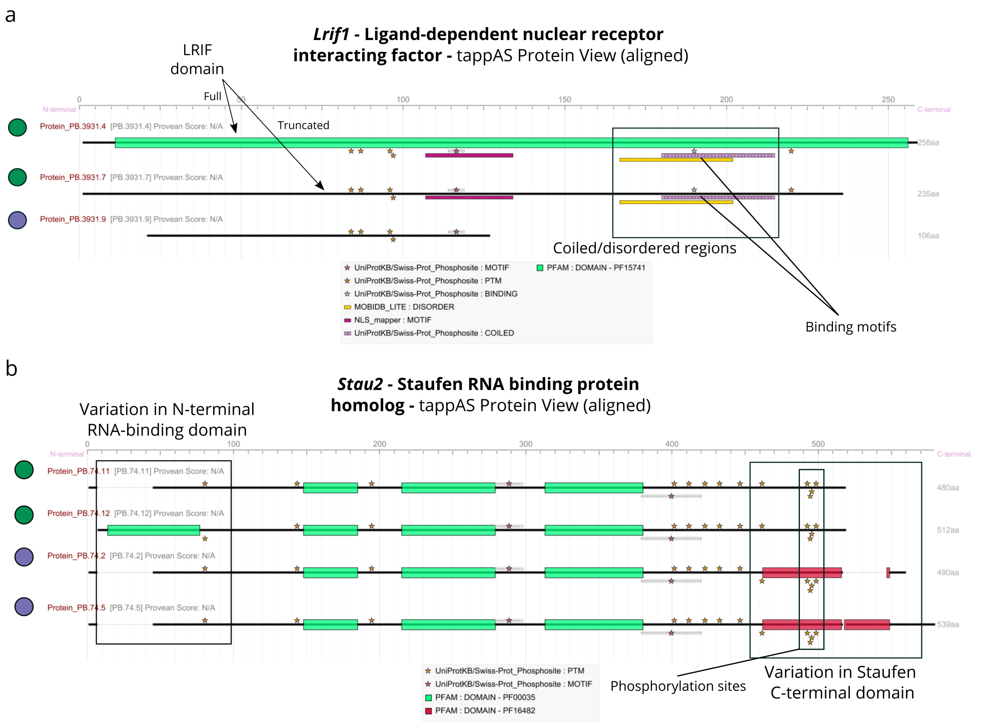

a
Lrif1 - Ligand-dependent nuclear receptor interacting factor - tappAS Protein View (aligned)
LRIF domain
Full
Truncated
N-terminal
C-terminal
0
50
100
150
200
250
Protein_PB.3931.4 [PB.3931.4] Provean Score: N/A
Protein_PB.3931.7 [PB.3931.7] Provean Score: N/A
Protein_PB.3931.9 [PB.3931.9] Provean Score: N/A
258aa
235aa
106aa
Coiled/disordered regions
Binding motifs
UniProtKB/Swiss-Prot_Phosphosite : MOTIF
UniProtKB/Swiss-Prot_Phosphosite : PTM
UniProtKB/Swiss-Prot_Phosphosite : BINDING
MOBIDB_LITE : DISORDER
NLS_mapper : MOTIF
UniProtKB/Swiss-Prot_Phosphosite : COILED
PFAM : DOMAIN - PF15741
b
Stau2 - Staufen RNA binding protein homolog - tappAS Protein View (aligned)
Variation in N-terminal RNA-binding domain
N-terminal
C-terminal
0
100
200
300
400
500
Protein_PB.74.11 [PB.74.11] Provean Score: N/A
Protein_PB.74.12 [PB.74.12] Provean Score: N/A
Protein_PB.74.2 [PB.74.2] Provean Score: N/A
Protein_PB.74.5 [PB.74.5] Provean Score: N/A
480aa
512aa
490aa
539aa
UniProtKB/Swiss-Prot_Phosphosite : PTM
UniProtKB/Swiss-Prot_Phosphosite : MOTIF
PFAM : DOMAIN - PF00035
PFAM : DOMAIN - PF16482
Phosphorylation sites
Variation in Staufen C-terminal domain

Figure 5.15: tappAS view of protein functional annotation for **a** *Lrif1* and **b** *Stau2* isoforms. Cluster assignments for each isoform are indicated by dot color (green: GABA-glutamatergic neuron cluster, orange: oligodendrocyte cluster, purple: expression in both oligodendrocytes and neurons).

expression in the primary visual cortex among 5 GABA-ergic neuron subtypes (Lamp5, Pvalb, Sncg, Sst, Vip; 4,921 total cells post-QC) defined in a more recent study by Tasic et al. [122] (hereby referred to as Tasic 2018 dataset). Remarkably, quantification of isoform expression using the previously-defined long read transcriptome (see chapter 4, section 4.2.2) resulted in no large transcriptomic differences among the subtypes, as revealed by UMAP [250] dimension reduction (figure 5.16.a), with Pvalb neurons showing the largest differences with the rest of the cell types.

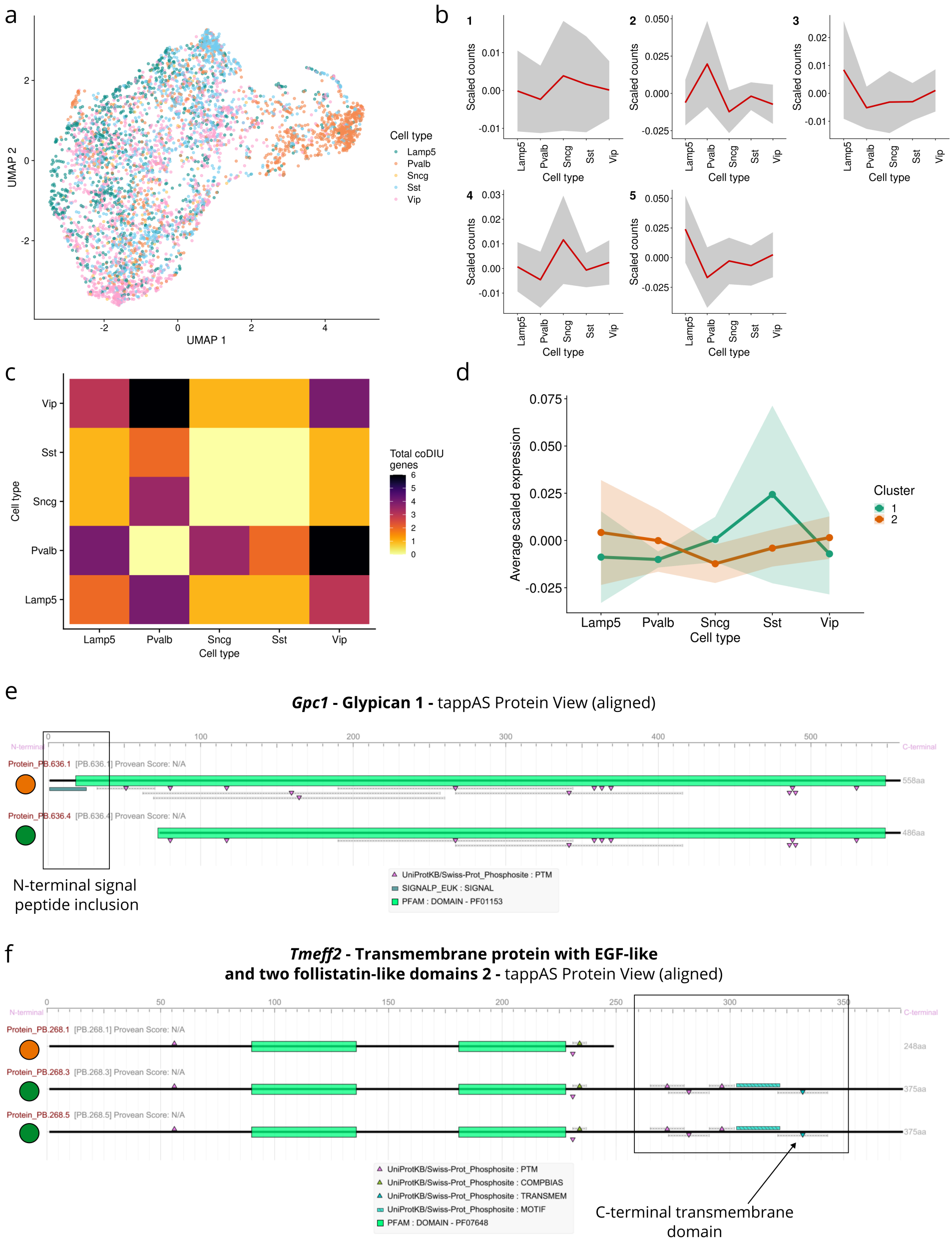

a
UMAP 2
UMAP 1
Cell type
Lamp5
Pvalb
Sncg
Sst
Vip
b
1
2
3
4
5
Scaled counts
Cell type
Lamp5
Pvalb
Sncg
Sst
Vip
c
Cell type
Total coDIU genes
d
Average scaled expression
Cluster
1
2
e
Gpc1 - Glypican 1 - tappAS Protein View (aligned)
N-terminal
C-terminal
Protein_PB.636.1 [PB.636.1] Provean Score: N/A
558aa
Protein_PB.636.4 [PB.636.4] Provean Score: N/A
486aa
N-terminal signal peptide inclusion
UniProtKB/Swiss-Prot_Phosphosite : PTM
SIGNALP_EUK : SIGNAL
PFAM : DOMAIN - PF01153
f
Tmeff2 - Transmembrane protein with EGF-like and two follistatin-like domains 2 - tappAS Protein View (aligned)
N-terminal
C-terminal
Protein_PB.268.1 [PB.268.1] Provean Score: N/A
248aa
Protein_PB.268.3 [PB.268.3] Provean Score: N/A
375aa
Protein_PB.268.5 [PB.268.5] Provean Score: N/A
375aa
UniProtKB/Swiss-Prot_Phosphosite : PTM
UniProtKB/Swiss-Prot_Phosphosite : COMPBIAS
UniProtKB/Swiss-Prot_Phosphosite : TRANSMEM
UniProtKB/Swiss-Prot_Phosphosite : MOTIF
PFAM : DOMAIN - PF07648
C-terminal transmembrane domain

Figure 5.16: Analysis of coDIU in GABA-ergic neuron subtypes. **a** UMAP visualization of GABA-ergic neuron subtypes in the Tasic 2018 dataset. **b** Clustering results. Isoform expression was averaged at the cell-level and aggregated by computing the mean in each cell type. x-axis shows scaled expression (counts), y-axis shows cell subtype labels. Grey area corresponds to cell type mean ± standard deviation. **c** Cell type-level coDIU patterns. For each pair of neuron subtypes in the x and y-axis, heatmap color corresponds to the total number of coDIU genes found between them. Total: 16 coDIU genes. **d** Average expression profile of isoforms from coDIU genes in clusters 1 (7 isoforms) and 2 (13 isoforms), as described in (b). tappAS protein view of **e** Gpc1 and **f** Tmeff2 genes.

Accordingly, DE analysis between the five groups only returned 568 significantly DE isoforms (FDR<0.05, fold-change>1.5, see section 5.2.8), which correspond to ~4% of all isoforms included in the analysis (13,870 isoforms remaining post-QC). Using percentile correlations, DE isoforms were grouped in 171 small clusters, which were then merged into 5 distinct expression profiles (figure 5.16.b) using dynamic hierarchical clustering (see section 5.2.8). The largest among the 5 clusters contained those isoforms with higher relative expression in Pvalb neurons (cluster 2, 184 isoforms, figure 5.16.b), in agreement with the global data structure (figure 5.16.a).

The DIU and coDIU relationships encoded by these modules of co-expressed isoform were next investigated. 22 DIU genes were found among the 5 clusters, 16 of which presented significant coDIU relationships with at least one other gene, involving 36 isoforms in total. In line with the clustering and UMAP results (figures 5.16.a-b), most isoforms participating in coDIU relationships presented higher relative expression in Pvalb neurons (figure 5.16.c). We therefore decided to explore the functional features that varied among coDIU genes that had isoforms in the Pvalb expression cluster. Remarkably, two of the detected coDIU genes were membrane-associated proteoglycans

Gpc1 and *Tmeff2*, which have been described to have neuronal function in previous studies [251, 252]. These genes presented coordinated switches in isoform expression in the Pvalb (cluster 1) and Scng (cluster2) subtypes (figure 5.16.d) as well as cell type-specific changes in the functional properties of their isoforms. Namely, *Gpc1* presented Pvalb-specific increase in expression of a signal peptide-including isoform (figure 5.16.e). Meanwhile, for *Tmeff2*, Pvalb-expressed isoforms were missing the C-terminal end, which included the transmembrane domain of the protein (figure 5.16.f). Even though these -and other similar results shown in this manuscript- would require further validation, they serve to illustrate the potential of acorde to uncover candidates for functionally-relevant isoform co-expression relationships.

Of note, the low number of clusters and DIU/coDIU genes detected by the acorde pipeline in this dataset, explained by the high isoform expression homogeneity among GABA cell subtypes, precluded the generation of more comprehensive functional results. Even so, we strongly believe that, all in all, these analyses demonstrate that acorde can be used to study isoform co-expression even in untoward scenarios, making a case for its usability in datasets with less well-defined cell types or lower signal-to-noise ratio.

5.4 Discussion

Alternative Splicing (AS) is known to be a tightly-regulated process in which splicing factors interact to create cell type-specific isoform expression patterns [253]. The transcriptome-level consequences of AS regulation have been studied in different ways, including the detection of within-isoform coordination of alternative sites [84, 91] and the generation of gene-isoform networks to uncover regulatory relationships [254–257]. More recently, single-cell transcriptomics applications have been used to unravel cell type-specific isoform expression patterns [77, 137]. When combined with spatial data, these studies have provided powerful insight on the regional specificity of AS [77, 78]. However, these investigations have largely focused on gene-level detection of Differential Isoform Usage (DIU), with no attempt to find patterns of isoform expression changes across genes. As a result, the extent to which AS regulation creates co-expression patterns among alternative isoforms from different genes has not yet been fully addressed. Previous research tackling isoform co-expression has either focused on specific event types, such as alternative 3' exons [107], or solely on the identification of functionally-relevant alternative isoforms in different biological contexts [258, 259]. In this work, we developed a pipeline (acorde, https://github.com/ConesaLab/acorde) that not only generates isoform co-expression networks and detects genes with DIU, but can also discover modules of genes that jointly change their isoforms across multiple cell types, i.e. genes with co-Differential Isoform Usage (coDIU).

First, we developed percentile correlations, a metric designed to overcome single-cell noise and sparsity and provide high-confidence estimates of isoform-to-isoform co-expression. By summarizing cell-level expression estimates into an expression value distribution, the metric draws on cell type-level patterns and avoids relying on cell-to-cell comparisons. Here, we show that percentile-summarized Pearson correlations outperform both classic and single-cell specific correlation strategies such as zero-inflated Kendall correlation [224]. Specifically, our metric can better discriminate true correlated and uncorrelated isoform pairs and capture simulated co-expression clusters with higher accuracy than the evaluated methods. In addition, results obtained using percentile correlations were comparable -even superior in some aspects- to those yielded by proportionality methods, which were recently proposed as one of the best alternatives to measure co-expression in single-cell data [220]. Specifically, percentile correlation matched the sensitivity of ρ proportionality when detecting co-expression, while ρ clustering results were only slightly more accurate. Nevertheless, ρ-generated clusters left a large proportion of transcripts unassigned, in contrast with the near-complete assignments obtained when combining of percentiles and Pearson correlation, which explains this minor gap in accuracy and highlights the consistency of our metric.

The comparison presented in this chapter also constitutes, to the best of our knowledge, the first isoform-centric evaluation of single-cell co-expression metrics. The recent, comprehensive review by Skinnider et al. [220] focused solely on gene-level correlation and, moreover, did not evaluate the impact of correlation metric selection on feature clustering results. The present study successfully accounts for this, while additionally providing a pioneer data simulation strategy to incorporate co-expression patterns into synthetic single-cell data. Considering these innovative aspects, we believe that the evaluation approach developed for the present study could be extended to benchmark other scRNA-Seq isoform or gene clustering methods.

This work additionally adapts the concept of DIU, originally defined for bulk transcriptomics [121], to the multi-group structure of scRNA-Seq data. Specifically, in acorde, genes that have isoforms assigned to different co-expression modules are considered to be undergoing DIU. To ensure the robustness of this strategy, isoforms are previously required to be significantly differentially expressed in at least one cell type (see chapter 4). The power of this interpretation of DIU lays on the fact that no pairwise cell type comparisons are required, enabling the construction of a global map of isoform selection and splicing changes across multiple cell types. This constitutes an improvement in comparison to some previously released methods, which have been designed to interrogate alternatively spliced genes for only two cell types at a time [78, 260]. Conversely, studies featuring multi-cell type strategies for DIU testing have not included software implementations for the community [151], hindering usability, or only allow testing of pairwise isoform relationships [137]. Recently, a study by Morabito et al. [261] introduced and applied correlation-based method (hdWGCNA) to generate single-cell isoform co-expression networks. In this study, authors used a definition of DIU that was similar to the one established in our previous manuscript [219], which supports the soundness of our approach. Last, but not least, the present study is the first to extend DIU testing to encompass cross-gene isoform usage changes (i.e. coDIU), paving the way towards an analysis of AS that considers isoform interactions beyond single genes.

The usage of a long read-defined, functionally annotated transcriptome enabled us to quantify full-length isoforms and obtain a biological readout from the isoform network. In the context of the two datasets that were analyzed [122, 163], coDIU genes were found to be enriched in the same biological functions, a number of which were unique in comparison to genes solely reported as DIU. Inter-gene isoform co-expression therefore appeared to impact a subset of DIU genes sharing specific functions. This suggested that coDIU could contribute to some cellular processes by supplying an additional layer of complexity, operating as a fine-tuning mechanism. Our analyses also revealed that isoforms from coDIU genes encompassed higher functional diversity than those belonging to DIU genes. In other words, genes that jointly change their expressed isoforms also generate a larger amount of changes in the differential inclusion of the functional features they encode. Importantly, these changes were found to impact the same types of features for certain groups of coDIU genes, giving rise to simultaneous changes in functional properties among isoforms. These functional synergies between alternatively spliced genes can be thought of as a result of AS and post-transcriptional regulation mechanisms, and may contribute to modulate key cell-level processes and encode cell-type identity. While these insights need to be subject to further experimental validation, they serve to illustrate the hypothesis-generating power of our pipeline and, moreover, state the importance of shifting the field's perspective of isoform expression from a gene-isolated process towards a network of interconnected isoforms and genes.

Chapter 6

Conclusions

The conclusions of this thesis are reported according to the goals that were set in Chapter 2:

1. Evaluate the potential of single-cell RNAseq data for the study of alternative splicing and alternative isoform expression

- We defined a set of four requirements for the study of isoforms using single-cell data, which were consistent with the properties of alternative isoform expression. Given the lack of benchmark and review studies at the time, this set the first reference for future studies in the field.

- These requirements were situated in context, evaluating to what extent each of them was met in published single-cell isoform studies. This provided the first isoform-oriented literature review study in the field.

- The computational methods that were used in said studies were additionally reviewed, and an evaluation of their suitability made in the light of our set of requirements.

- Data-associated limitations brought out by the study were reproduced using computational simulations, anticipating potential pitfalls that could be found during future analyses. These analyses added perspective to the limitations of UMI-based methods for isoform discrimination and illustrated the isoform detection issues associated with shallow single-cell long-read sequencing.

- Even though no library preparation technology was able to provide the conditions for accurate isoform computational analysis, full-length methods (i.e. Smart-seq2) provided the best balance.

- In spite of being designed for bulk RNA-seq, the most suitable set of methods for isoform-level quantification were those based on the Expectation Maximization (EM) algorithm.
- This insight successfully informed pipeline design and analysis decisions made during Chapters 4 and 5.

2. Design a single-cell RNAseq data processing pipeline that is compatible with downstream isoform-level analyses

- A combination of bulk long-read an single-cell short-read RNA-seq data was successfully used to achieve a balance between adequate isoform reconstruction and sensitive cell-level quantification, respectively.

- During long-read transcript reconstruction, both extant (i.e. the SQANTI ML filter) and innovative strategies (i.e. data-driven filtering of unsupported isoforms) for transcriptome curation were integrated, obtaining a high-quality set of neural isoforms.

- The developed transcriptome curation strategies were refined and implemented as novel modules in the transcriptome quality control software SQANTI3. This includes an improved version of the original SQANTI filter and a novel rescue module.

- Our comprehensive evaluation of automated and manual transcript filtering, especially in the context of machine learning-based filters like those in SQANTI and SQANTI3, has significantly contributed to enhancing the roadmap for transcriptome curation. This work provides valuable guidance for establishing filtering rules and offers insights into the decision-making processes of random forest models.

- Regarding the application of the SQANTI3 ML filter, this work illustrates the importance of defining a reliable true positive set, as well as of the integration of multiple validation data sources for each aspect of transcript structural variability, preferably from the same biological samples.

- The evaluation of the new SQANTI Rescue module in the context of ML-filtered data demonstrated its effectiveness in reclaiming lost reference transcripts and genes, enriching transcriptome complexity by providing high-quality replacement transcripts for numerous rescue candidates. This ultimately unveiled the importance of considering non-FSM artifacts and validated the value of including a rescue step in transcriptome curation pipelines.
- We leveraged functional annotation transference using the IsoAnnotLite tool to allow coupling of expression analyses and biological interpretation.
- We enabled multi-group differential expression analysis for single-cell data, precluding pairwise comparisons and detecting transcript expression changes across several cell types.

3. Develop a novel analysis method to leverage single-cell RNA-seq data to gain valuable insight into isoform biology.

- We developed percentile correlations, a novel metric that successfully overcomes single-cell noise and captures transcript co-expression across cell types.

- Using a semi-supervised clustering strategy, we revealed that there are modules of co-expressed isoforms that can be detected in single-cell data, revealing a layer of regulation that takes part in the transcriptomic identity of cell types and is independent of gene expression.

- We extended the definition of Differential Isoform Usage (DIU) to a multi-cell type design, and developed the concept of co-Differential Isoform Usage (coDIU), facilitating the interpretation of the isoform cluster network.

- CoDIU genes were found to have a comprehensive, yet distinct functional signature, showing enrichment in cell processes that were different to those found in the analysis of DIU genes.

- Our results show that co-expression of isoform across cell types is coupled with coordinated functional changes in transcript and protein properties, with coDIU genes showing more frequent and structurally/functionally similar changes in the inclusion/exclusion of transcript and protein motifs and domains.

- We released *acorde*, an R package for the detection of isoform co-usage networks in single-cell RNA-seq data in which all code and functionalities developed for the fulfillment of this aim are made available for users in a reproducible manner.

References

1. Mortazavi, A., Williams, B. A., McCue, K., Schaeffer, L. & Wold, B. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature Methods* **5,** 621–628 (2008).
2. Wang, Z., Gerstein, M. & Snyder, M. RNA-Seq: a revolutionary tool for transcriptomics. *Nature Reviews Genetics* **10,** 57–63 (2009).
3. Core, L. J., Waterfall, J. J. & Lis, J. T. Nascent RNA Sequencing Reveals Widespread Pausing and Divergent Initiation at Human Promoters. *Science* **322,** 1845–1848 (2008).
4. Batut, P. & Gingeras, T. R. RAMPAGE: promoter activity profiling by paired-end sequencing of 5'-complete cDNAs. *Current protocols in molecular biology* **104,** Unit 25B.11 (Nov. 2013).
5. Pelechano, V., Wei, W., Jakob, P. & Steinmetz, L. M. Genome-wide identification of transcript start and end sites by transcript isoform sequencing. *Nature protocols* **9,** 1740–59 (2014).
6. Ingolia, N. T., Ghaemmaghami, S., Newman, J. R. S. & Weissman, J. S. Genome-Wide Analysis in Vivo of Translation with Nucleotide Resolution Using Ribosome Profiling. *Science* **324,** 218–223 (2009).
7. Licatalosi, D. D. *et al.* HITS-CLIP yields genome-wide insights into brain alternative RNA processing. *Nature* **456,** 464–469 (2008).
8. Cloonan, N. *et al.* Stem cell transcriptome profiling via massive-scale mRNA sequencing. *Nature Methods* **5,** 613–619 (2008).

9. Pan, Q., Shai, O., Lee, L. J., Frey, B. J. & Blencowe, B. J. Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing. *Nature Genetics* **40,** 1413–1415 (2008).

10. Wang, E. T. *et al.* Alternative isoform regulation in human tissue transcriptomes. *Nature* **456,** 470–476 (2008).

11. Tang, F. *et al.* mRNA-Seq whole-transcriptome analysis of a single cell. *Nature Methods* **6,** 377–382 (2009).

12. Llorens-Bobadilla, E. *et al.* Single-Cell Transcriptomics Reveals a Population of Dormant Neural Stem Cells that Become Activated upon Brain Injury. *Cell Stem Cell* **17,** 329–340 (2015).

13. Buettner, F. *et al.* Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nature Biotechnology* **33,** 155–160 (2015).

14. Macosko, E. Z. *et al.* Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell* **161,** 1202–1214 (2015).

15. Zeisel, a. *et al.* Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347,** 1138–42 (2015).

16. Chen, R., Wu, X., Jiang, L. & Zhang, Y. Single-Cell RNA-Seq Reveals Hypothalamic Cell Diversity. *Cell Reports* **18,** 3227–3241 (2017).

17. Tang, F. *et al.* Tracing the derivation of embryonic stem cells from the inner cell mass by single-cell RNA-seq analysis. *Cell Stem Cell* **6,** 468–478 (2010).

18. Treutlein, B. *et al.* Reconstructing lineage hierarchies of the distal lung epithelium using single-cell RNA-seq. *Nature* **509,** 371–375 (2014).

19. DeLaughter, D. M. *et al.* Single-Cell Resolution of Temporal Gene Expression during Heart Development. *Developmental Cell* **39,** 480–490 (2016).

20. Chu, L.-F. *et al.* Single-cell RNA-seq reveals novel regulators of human embryonic stem cell differentiation to definitive endoderm. *Genome Biology* **17,** 173 (2016).

21. Yao, Z. *et al.* A Single-Cell Roadmap of Lineage Bifurcation in Human ESC Models of Embryonic Brain Development. *Cell Stem Cell* **20,** 120–134 (2017).

22. Tirosh, I. *et al.* Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq. *Science* **352,** 189–196 (2016).

23. Tirosh, I. *et al.* Single-cell RNA-seq supports a developmental hierarchy in human oligodendroglioma. *Nature* **539,** 309–313 (2016).

24. Zeb, Q., Wang, C., Shafiq, S. & Liu, L. *Chapter 6 - An Overview of Single-Cell Isolation Techniques* (eds Barh, D. & Azevedo, V.) 101–135. ISBN: 978-0-12-814919-5 (Academic Press, Jan. 2019).

25. Ziegenhain, C. *et al.* Comparative Analysis of Single-Cell RNA Sequencing Methods. *Molecular Cell* **65,** 631–643.e4 (2017).

26. Schmid, K. T. *et al.* scPower accelerates and optimizes the design of multi-sample single cell transcriptomic studies. *Nature Communications* **12,** 6625 (Nov. 2021).

27. Kolodziejczyk, A. A., Kim, J. K., Svensson, V., Marioni, J. C. & Teichmann, S. A. The Technology and Biology of Single-Cell RNA Sequencing. *Molecular Cell* **58,** 610–620 (2015).

28. Ziegenhain, C., Vieth, B., Parekh, S., Hellmann, I. & Enard, W. Quantitative single-cell transcriptomics. *Briefings in Functional Genomics,* 1–13 (May 2018).

29. Stegle, O., Teichmann, S. A. & Marioni, J. C. Computational and analytical challenges in single-cell transcriptomics. *Nature Reviews Genetics* **16,** 133–145 (2015).

30. Parekh, S., Ziegenhain, C., Vieth, B., Enard, W. & Hellmann, I. The impact of amplification on differential expression analyses by RNA-seq. *Scientific Reports* **6,** 25533 (2016).

31. Islam, S. *et al.* Quantitative single-cell RNA-seq with unique molecular identifiers. *Nature Methods* **11,** 163–166 (2013).

32. Klein, A. M. *et al.* Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell* **161,** 1187–1201 (2015).

33. Picelli, S. *et al.* Tn5 transposase and tagmentation procedures for massively scaled sequencing projects. *Genome Research* **24,** 2033–2040 (2014).

34. Picelli, S. *et al.* Full-length RNA-seq from single cells using Smart-seq2. *Nature Protocols* **9,** 171–181 (Jan. 2, 2014).

35. Gupta, I. *et al.* Single-cell isoform RNA sequencing characterizes isoforms in thousands of cerebellar cells. *Nature Biotechnology* **36,** 1197–1202 (Dec. 1, 2018).

36. Singh, M. *et al.* High-throughput targeted long-read single cell sequencing reveals the clonal and transcriptional landscape of lymphocytes. *Nature Communications* **10,** 3120 (July 2019).

37. Arzalluz-Luque, Á. & Conesa, A. Single-cell RNAseq for the study of isoforms—how is that possible? *Genome Biology* **19,** 110 (Dec. 10, 2018).

38. Svensson, V., Vento-Tormo, R. & Teichmann, S. A. Exponential scaling of single-cell RNA-seq in the past decade. *Nature Protocols* **13,** 599–604 (Apr. 1, 2018).

39. Kharchenko, P. V., Silberstein, L. & Scadden, D. T. Bayesian approach to single-cell differential expression analysis. *Nature Methods* **11,** 740–742 (July 2014).

40. Zhu, Y., Machleder, E., Chenchik, a., Li, R. & Siebert, P. Reverse transcriptase template switching: A SMART (TM) approach for full-length cDNA library construction. *Biotechniques* **30,** 892–897 (2001).

41. Phipson, B., Zappia, L. & Oshlack, A. Gene length and detection bias in single cell RNA sequencing protocols. *F1000Research* (Apr. 2017).

42. Steijger, T. *et al.* Assessment of transcript reconstruction methods for RNA-seq. *Nature Methods* **10,** 1177–1184 (2013).

43. Engström, P. G. *et al.* Systematic evaluation of spliced alignment programs for RNA-seq data. *Nature Methods* **10,** 1185–1191 (Dec. 2013).

44. Byrne, A., Cole, C., Volden, R. & Vollmers, C. Realizing the potential of full-length transcriptome sequencing. *Philosophical Transactions of the Royal Society B: Biological Sciences* **374,** 20190097 (Oct. 2019).

45. Oikonomopoulos, S. *et al.* Methodologies for Transcript Profiling Using Long-Read Technologies. *Frontiers in Genetics* **11** (2020).

46. Eid, J. *et al.* Real-Time DNA Sequencing from Single Polymerase Molecules. *Science* **323,** 133–138 (Jan. 2009).

47. Rhoads, A. & Au, K. F. PacBio Sequencing and Its Applications. *Genomics, Proteomics & Bioinformatics. SI: Metagenomics of Marine Environments* **13,** 278–289 (Oct. 2015).

48. Jain, M. *et al.* Improved data analysis for the MinION nanopore sequencer. *Nature Methods* **12,** 351–356 (2015).

49. Jain, M., Olsen, H. E., Paten, B. & Akeson, M. The Oxford Nanopore MinION: delivery of nanopore sequencing to the genomics community. *Genome Biology* **17,** 239 (Nov. 2016).

50. Tardaguila, M. *et al.* SQANTI: extensive characterization of long-read transcript sequences for quality control in full-length transcriptome identification and quantification. *Genome Research* **28,** 396–411 (Mar. 9, 2018).

51. Jaworski, E. & Routh, A. Parallel ClickSeq and Nanopore sequencing elucidates the rapid evolution of defective-interfering RNAs in Flock House virus. *PLoS Pathogens* **13** (2017).

52. Tilgner, H., Grubert, F., Sharon, D. & Snyder, M. P. Defining a personal, allele-specific, and single-molecule long-read transcriptome. *Proceedings of the National Academy of Sciences* **111,** 9869–9874 (July 2014).

53. Byrne, A. *et al.* Nanopore long-read RNAseq reveals widespread transcriptional variation among the surface receptors of individual B cells. *Nature Communications* **8,** 16027 (Dec. 19, 2017).

54. Karlsson, K. & Linnarsson, S. Single-cell mRNA isoform diversity in the mouse brain. *BMC Genomics* **18,** 126 (2017).

55. Lebrigand, K., Magnone, V., Barbry, P. & Waldmann, R. High throughput error corrected Nanopore single cell transcriptome sequencing. *Nature Communications* **11,** 4025 (Aug. 2020).

56. Ebrahimi, G. *et al.* Fast and accurate matching of cellular barcodes across short-reads and long-reads of single-cell RNA-seq experiments. *iScience* **25,** 104530 (July 2022).

57. You, Y. *et al.* Identification of cell barcodes from long-read single-cell RNA-seq with BLAZE. *Genome Biology* **24,** 66 (Apr. 2023).

58. Bacher, R. & Kendziorski, C. Design and computational analysis of single-cell RNA-sequencing experiments. *Genome Biology* **17,** 63 (2016).

59. Grün, D. & Van Oudenaarden, A. Design and Analysis of Single-Cell Sequencing Experiments. *Cell* **163,** 799–810 (2015).

60. Hicks, S. C., Townes, F. W., Teng, M. & Irizarry, R. A. Missing data and technical variability in single-cell RNA-sequencing experiments. *Biostatistics* **19,** 562–578 (Oct. 2018).

61. Lähnemann, D. *et al.* Eleven grand challenges in single-cell data science. *Genome Biology* **21,** 31 (Feb. 2020).

62. Wu, Y. & Zhang, K. Tools for the analysis of high-dimensional single-cell RNA sequencing data. *Nature Reviews Nephrology* **16,** 408–421 (July 2020).

63. Luecken, M. D. & Theis, F. J. Current best practices in single-cell RNA-seq analysis: a tutorial. *Molecular Systems Biology* **15,** e8746 (June 2019).

64. Nilsen, T. W. & Graveley, B. R. Expansion of the eukaryotic proteome by alternative splicing. *Nature* **463,** 457–463 (2010).

65. Kalsotra, A. & Cooper, T. A. Functional consequences of developmentally regulated alternative splicing. *Nature Reviews Genetics* **12,** 715–729 (2011).

66. Kim, E., Magen, A. & Ast, G. Different levels of alternative splicing among eukaryotes. *Nucleic Acids Research* **35,** 125–131 (Jan. 2007).

67. Schad, E., Tompa, P. & Hegyi, H. The relationship between proteome size, structural disorder and organism complexity. *Genome Biology* **12,** R120 (Dec. 2011).

68. Yang, P., Wang, D. & Kang, L. Alternative splicing level related to intron size and organism complexity. *BMC Genomics* **22,** 853 (Nov. 2021).

69. Black, D. L. Mechanisms of alternative pre-messenger RNA splicing. *Annual review of biochemistry* **72,** 291–336 (2003).

70. Matera, A. G. & Wang, Z. A day in the life of the spliceosome. *Nature Reviews Molecular Cell Biology* **15,** 108–121 (Feb. 2014).

71. Chen, M. & Manley, J. L. Mechanisms of alternative splicing regulation: insights from molecular and genomics approaches. *Nature Reviews Molecular Cell Biology* **10,** 741–754 (Nov. 2009).

72. Di Giammartino, D. C., Nishida, K. & Manley, J. L. Mechanisms and Consequences of Alternative Polyadenylation. *Molecular Cell* **43,** 853–866 (Sept. 2011).

73. Tian, B. & Manley, J. L. Alternative polyadenylation of mRNA precursors. *Nature Reviews Molecular Cell Biology* **18,** 18–30 (2016).

74. Smith, C. W. & Valcárcel, J. Alternative pre-mRNA splicing: the logic of combinatorial control. *Trends in Biochemical Sciences* **25,** 381–388 (Aug. 2000).

75. Matlin, A. J., Clark, F. & Smith, C. W. J. Understanding alternative splicing: towards a cellular code. *Nature Reviews Molecular Cell Biology* **6,** 386–398 (May 2005).

76. Gallego-Paez, L. M. *et al.* Alternative splicing: the pledge, the turn, and the prestige: The key role of alternative splicing in human biological systems. *Human Genetics,* 1–28 (2017).

77. Booeshaghi, A. S. *et al.* Isoform cell-type specificity in the mouse primary motor cortex. *Nature* **598,** 195–199 (Oct. 2021).

78. Joglekar, A. *et al.* A spatially resolved brain region- and cell type-specific isoform atlas of the postnatal mouse brain. *Nature Communications* **12,** 463 (Dec. 19, 2021).

79. Martin, J. A. & Wang, Z. Next-generation transcriptome assembly. *Nature Reviews Genetics* **12,** 671–682 (Oct. 2011).

80. Dobin, A. *et al.* STAR: Ultrafast universal RNA-seq aligner. *Bioinformatics* **29,** 15–21 (2013).

81. Katz, Y., Wang, E. T., Airoldi, E. M. & Burge, C. B. Analysis and design of RNA sequencing experiments for identifying isoform regulation. *Nature Methods* **7,** 1009–1015 (2010).

82. Shen, S. *et al.* rMATS: robust and flexible detection of differential alternative splicing from replicate RNA-Seq data. *Proceedings of the National Academy of Sciences of the United States of America* **111,** E5593–601 (Dec. 23, 2014).

83. Trincado, J. L. *et al.* SUPPA2: Fast, accurate, and uncertainty-aware differential splicing analysis across multiple conditions. *Genome Biology* (2018).

84. Tilgner, H. *et al.* Comprehensive transcriptome analysis using synthetic long-read sequencing reveals molecular co-association of distant splicing events. *Nature Biotechnology* **33,** 736–742 (July 18, 2015).

85. Li, B. & Dewey, C. N. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC Bioinformatics* **12,** 323 (Dec. 4, 2011).

86. Patro, R., Mount, S. M. & Kingsford, C. Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms. *Nature Biotechnology* **32,** 462–464 (May 2014).

87. Bray, N. L., Pimentel, H., Melsted, P. & Pachter, L. Near-optimal probabilistic RNA-seq quantification. *Nature Biotechnology* **34,** 525–527 (May 2016).

88. Patro, R., Duggal, G., Love, M. I., Irizarry, R. A. & Kingsford, C. Salmon provides fast and bias-aware quantification of transcript expression. *Nature Methods* **14,** 417–419 (Apr. 2017).

89. Tilgner, H. *et al.* Accurate Identification and Analysis of Human mRNA Isoforms Using Deep Long Read Sequencing. *G3 Genes|Genomes|Genetics* **3,** 387–397 (Mar. 2013).

90. Sharon, D., Tilgner, H., Grubert, F. & Snyder, M. A single-molecule long-read survey of the human transcriptome. *Nature Biotechnology* **31,** 1009–1014 (Nov. 2013).

91. Tilgner, H. *et al.* Microfluidic isoform sequencing shows widespread splicing coordination in the human transcriptome. *Genome Research* **28,** 231–242 (Feb. 1, 2018).

92. Raj, B. & Blencowe, B. J. Alternative Splicing in the Mammalian Nervous System: Recent Insights into Mechanisms and Functional Roles. *Neuron* (2015).

93. Martinez, N. M. & Lynch, K. W. Control of alternative splicing in immune responses: Many regulators, many predictions, much still to learn. *Immunological Reviews* **253,** 216–236 (2013).

94. Teichroeb, J. H., Kim, J. & Betts, D. H. The role of telomeres and telomerase reverse transcriptase isoforms in pluripotency induction and maintenance. *RNA Biology* **13,** 707–719 (2016).

95. Irimia, M. & Blencowe, B. J. Alternative splicing: Decoding an expansive regulatory layer. *Current Opinion in Cell Biology* **24,** 323–332 (2012).

96. Braunschweig, U., Gueroussov, S., Plocik, A. M., Graveley, B. R. & Blencowe, B. J. Dynamic integration of splicing within gene regulatory pathways. *Cell* (2013).

97. Paronetto, M. P., Passacantilli, I. & Sette, C. Alternative splicing and cell survival: from tissue homeostasis to disease. *Cell Death and Differentiation* **23,** 1919–1929 (2016).

98. Marinov, G. K. *et al.* From single-cell to cell-pool transcriptomes: Stochasticity in gene expression and RNA splicing. *Genome Research* **24,** 496–510 (2014).

99. Ramsköld, D. *et al.* Full-length mRNA-Seq from single-cell levels of RNA and individual circulating tumor cells. *Nature Biotechnology* **30,** 777–782 (2012).

100. Zhang, J., Kuo, C. C. J. & Chen, L. WemIQ: An accurate and robust isoform quantification method for RNA-seq data. *Bioinformatics* **31,** 878–885 (2015).

101. Shalek, A. K. *et al.* Single-cell transcriptomics reveals bimodality in expression and splicing in immune cells. *Nature* **498,** 236–240 (2013).

102. Song, Y. *et al.* Single-Cell Alternative Splicing Analysis with Expedition Reveals Splicing Dynamics during Neuron Differentiation. *Molecular Cell* **67,** 148–161.e5 (2017).

103. Welch, J. D., Hu, Y. & Prins, J. F. Robust detection of alternative splicing in a population of single cells. *Nucleic Acids Research* **44,** e73 (2016).

104. Huang, Y. & Sanguinetti, G. BRIE: transcriptome-wide splicing quantification in single cells. *Genome Biology* **18,** 123 (2017).

105. Velten, L. *et al.* Single-cell polyadenylation site mapping reveals 3' isoform choice variability. *Molecular Systems Biology* **11,** 812–812 (2015).

106. Karlsson, K., Lönnerberg, P. & Linnarsson, S. Alternative TSSs are co-regulated in single cells in the mouse brain. *Molecular systems biology* **13,** 930 (2017).

107. Yap, K. & Makeyev, E. V. Functional impact of splice isoform diversity in individual cells. *Biochemical Society Transactions* **44,** 1079–1085 (2016).

108. Brennecke, P. *et al.* Accounting for technical noise in single-cell RNA-seq experiments. *Nature Methods* **10,** 1093–1095 (2013).

109. Wu, A. R. *et al.* Quantitative assessment of single-cell RNA-sequencing methods. *Nature Methods* **11,** 41–46 (2013).

110. Gonzàlez-Porta, M., Frankish, A., Rung, J., Harrow, J. & Brazma, A. Transcriptome analysis of human tissues and cell lines reveals one dominant transcript per gene. *Genome Biology* **14,** R70 (July 1, 2013).

111. Zheng, G. X. Y. *et al.* Massively parallel digital transcriptional profiling of single cells. *Nature Communications* **8,** 14049 (Jan. 16, 2017).

112. Houseley, J. & Tollervey, D. Apparent non-canonical trans-splicing is generated by reverse transcriptase in vitro. *PLoS ONE* **5** (2010).

113. Nam, D. K. *et al.* Oligo(dT) primer generates a high frequency of truncated cDNAs through internal poly(A) priming during reverse transcription. *Proceedings of the National Academy of Sciences of the United States of America* **99,** 6152–6 (2002).

114. Cocquet, J., Chong, A., Zhang, G. & Veitia, R. A. Reverse transcriptase template switching and false alternative transcripts. *Genomics* **88,** 127–131 (2006).

115. Pervouchine, D. D., Knowles, D. G. & Guigó, R. Intron-centric estimation of alternative splicing from RNA-seq data. *Bioinformatics* **29,** 273–274 (2013).

116. Scialdone, A. *et al.* Resolving early mesoderm diversification through single-cell expression profiling. *Nature* **535,** 289–293 (2016).

117. Vallejos, C. A., Risso, D., Scialdone, A., Dudoit, S. & Marioni, J. C. Normalizing single-cell RNA sequencing data: Challenges and opportunities. *Nature Methods* **14,** 565–571 (2017).

118. Lun, A. T. L., Calero-Nieto, F. J., Haim-Vilmovsky, L., Göttgens, B. & Marioni, J. C. Assessing the reliability of spike-in normalization for analyses of single-cell RNA sequencing data. *Genome Research* **27,** 1795–1806 (Jan. 2017).

119. Frazee, A. C., Jaffe, A. E., Langmead, B. & Leek, J. T. Polyester: Simulating RNA-seq datasets with differential transcript expression. *Bioinformatics* **31,** 2778–2784 (2015).

120. Gordon, S. P. *et al.* Widespread Polycistronic Transcripts in Fungi Revealed by Single-Molecule mRNA Sequencing. *PLOS ONE* **10,** 1–15 (July 2015).

121. De la Fuente, L. *et al.* tappAS: a comprehensive computational framework for the analysis of the functional impact of differential splicing. *Genome Biology* **21,** 119 (Dec. 18, 2020).

122. Tasic, B. *et al.* Shared and distinct transcriptomic cell types across neocortical areas. *Nature* **563,** 72–78 (Nov. 2018).

123. Vento-Tormo, R. *et al.* Single-cell reconstruction of the early maternal–fetal interface in humans. *Nature* **563,** 347–353 (Nov. 2018).

124. Schaum, N. *et al.* Single-cell transcriptomics of 20 mouse organs creates a Tabula Muris. *Nature* **562,** 367–372 (Oct. 2018).

125. Almanzar, N. *et al.* A single-cell transcriptomic atlas characterizes ageing tissues in the mouse. *Nature* **583,** 590–595 (July 2020).

126. Yao, Z. *et al.* A transcriptomic and epigenomic cell atlas of the mouse primary motor cortex. *Nature* **598,** 103–110 (Oct. 2021).

127. The Tabula Sapiens Consortium. The Tabula Sapiens: A multiple-organ, single-cell transcriptomic atlas of humans. *Science* **376,** eabl4896 (May 2022).

128. Mamanova, L. *et al.* High-throughput full-length single-cell RNA-seq automation. *Nature Protocols* **16,** 2886–2915 (June 2021).

129. Jaeger, B. N. *et al.* Miniaturization of Smart-seq2 for Single-Cell and Single-Nucleus RNA Sequencing. *STAR Protocols* **1,** 100081 (Sept. 2020).

130. Ando, Y., Kwon, A. T.-J. & Shin, J. W. An era of single-cell genomics consortia. *Experimental & Molecular Medicine* **52,** 1409–1418 (Sept. 2020).

131. Hagemann-Jensen, M. *et al.* Single-cell RNA counting at allele and isoform resolution using Smart-seq3. *Nature Biotechnology* **38,** 708–714 (June 2020).

132. Westoby, J., Artemov, P., Hemberg, M. & Ferguson-Smith, A. Obstacles to detecting isoforms using full-length scRNA-seq data. *Genome Biology* **21,** 74 (Dec. 23, 2020).

133. Westoby, J., Herrera, M. S., Ferguson-Smith, A. C. & Hemberg, M. Simulation-based benchmarking of isoform quantification in single-cell RNA-seq. *Genome Biology* **19,** 191 (Dec. 2018).

134. Hu, Y., Wang, K. & Li, M. Detecting differential alternative splicing events in scRNA-seq with or without Unique Molecular Identifiers. *PLOS Computational Biology* **16,** e1007925 (June 2020).

135. Dehghannasiri, R., Olivieri, J. E., Damljanovic, A. & Salzman, J. Specific splice junction detection in single cells with SICILIAN. *Genome Biology* **22,** 219 (Aug. 2021).

136. Wen, W. X., Mead, A. J. & Thongjuea, S. VALERIE: Visual-based inspection of alternative splicing events at single-cell resolution. *PLOS Computational Biology* **16,** e1008195 (Sept. 2020).

137. Vu, T. N. *et al.* Isoform-level gene expression patterns in single-cell RNA-sequencing data. *Bioinformatics doi* **10,** 1–9 (March 2018).

138. Wen, W. X., Mead, A. J. & Thongjuea, S. MARVEL: an integrated alternative splicing analysis platform for single-cell RNA sequencing data. *Nucleic Acids Research* **51,** e29 (Mar. 2023).

139. Patrick, R. *et al.* Sierra: Discovery of differential transcript usage from polyA-captured single-cell RNA-seq data. *Genome Biology* **21,** 1–27 (July 8, 2020).

140. Olivieri, J. E., Dehghannasiri, R. & Salzman, J. The SpliZ generalizes 'percent spliced in' to reveal regulated splicing at single-cell resolution. *Nature Methods* **19,** 307–310 (Mar. 2022).

141. Tekath, T. & Dugas, M. Differential transcript usage analysis of bulk and single-cell RNA-seq data with DTUrtle. *Bioinformatics* **37** (ed Boeva, V.) 3781–3787 (Nov. 2021).

142. Gilis, J., Vitting-Seerup, K., Van den Berge, K. & Clement, L. satuRn: Scalable analysis of differential transcript usage for bulk and single-cell RNA-sequencing applications [version 2; peer review: 2 approved]. *F1000Research* **10** (2022).

143. Nowicka, M. & Robinson, M. D. *DRIMSeq: a Dirichlet-multinomial framework for multivariate count outcomes in genomics* tech. rep. 5:1356 (F1000Research, Dec. 2016).

144. Van den Berge, K., Soneson, C., Robinson, M. D. & Clement, L. stageR: a general stage-wise method for controlling the gene-level false discovery rate in differential expression and differential transcript usage. *Genome Biology* **18,** 151 (Aug. 2017).

145. Volden, R. *et al.* Improving nanopore read accuracy with the R2C2 method enables the sequencing of highly multiplexed full-length single-cell cDNA. *Proceedings of the National Academy of Sciences of the United States of America* **115,** 9726–9731 (2018).

146. Philpott, M. *et al.* Nanopore sequencing of single-cell transcriptomes with scCOLOR-seq. *Nature Biotechnology* **39,** 1517–1520 (Dec. 2021).

147. Mikheenko, A., Prjibelski, A. D., Joglekar, A. & Tilgner, H. U. Sequencing of individual barcoded cDNAs using Pacific Biosciences and Oxford Nanopore Technologies reveals platform-specific error patterns. *Genome Research* **32,** 726–737 (Apr. 2022).

148. You, Y., Clark, M. B. & Shim, H. NanoSplicer: accurate identification of splice junctions using Oxford Nanopore sequencing. *Bioinformatics* **38,** 3741–3748 (Aug. 2022).

149. Volden, R. & Vollmers, C. Single-cell isoform analysis in human immune cells. *Genome Biology* **23,** 47 (Dec. 2022).

150. Rebboah, E. *et al.* Mapping and modeling the genomic basis of differential RNA isoform expression at single-cell resolution with LR-Split-seq. *Genome Biology* **22,** 286 (Dec. 2021).

151. Tian, L. *et al.* Comprehensive characterization of single-cell full-length isoforms in human and mouse with long-read sequencing. *Genome Biology* **22,** 310 (Nov. 2021).

152. Boileau, E. *et al.* Full-Length Spatial Transcriptomics Reveals the Unexplored Isoform Diversity of the Myocardium Post-MI. *Frontiers in Genetics* **13,** 912572 (July 2022).

153. Hardwick, S. A. *et al.* Single-nuclei isoform RNA sequencing unlocks barcoded exon connectivity in frozen brain tissue. *Nature Biotechnology* **40,** 1082–1092 (July 2022).

154. Al'Khafaji, A. M. *et al.* High-throughput RNA isoform sequencing using programmed cDNA concatenation. *Nature Biotechnology,* 1–5 (June 2023).

155. Shi, Z.-X. *et al.* High-throughput and high-accuracy single-cell RNA isoform analysis using PacBio circular consensus sequencing. *Nature Communications* **14,** 2631 (May 2023).

156. Prjibelski, A. D. *et al.* Accurate isoform discovery with IsoQuant using long reads. *Nature Biotechnology,* 1–4 (Jan. 2023).

157. Soneson, C., Matthes, K. L., Nowicka, M., Law, C. W. & Robinson, M. D. Isoform prefiltering improves performance of count-based methods for analysis of differential transcript usage. *Genome Biology* **17,** 12 (Dec. 26, 2016).

158. Amarasinghe, S. L. *et al.* Opportunities and challenges in long-read sequencing data analysis. *Genome Biology* **21,** 30 (Feb. 2020).

159. Wyman, D. *et al. A technology-agnostic long-read analysis pipeline for transcriptome discovery and quantification* June 18, 2019.

160. Tang, A. D. *et al.* Full-length transcript characterization of SF3B1 mutation in chronic lymphocytic leukemia reveals downregulation of retained introns. *Nature Communications* **11,** 1438 (Mar. 2020).

161. Kuo, R. I. *et al.* Illuminating the dark side of the human transcriptome with long read transcript sequencing. *BMC Genomics* **21,** 1–22 (Dec. 1, 2020).

162. Pardo-Palacios, F. J. *et al.* SQANTI3: curation of long-read transcriptomes for accurate identification of known and novel isoforms. *bioRxiv* (2023).

163. Tasic, B. *et al.* Adult mouse cortical cell taxonomy revealed by single cell transcriptomics. *Nature Neuroscience* **19,** 335–346 (2016).

164. Forrest, A. R. *et al.* A promoter-level mammalian expression atlas. *Nature* **507,** 462–470 (Jan. 1, 2014).

165. Abugessaisa, I. *et al.* FANTOM5 CAGE profiles of human and mouse reprocessed for GRCh38 and GRCm38 genome assemblies. *Scientific Data* **4,** 170107 (Aug. 2017).

166. Tian, B., Hu, J., Zhang, H. & Lutz, C. S. A large-scale analysis of mRNA polyadenylation of human and mouse genes. *Nucleic Acids Research* **33,** 201–212 (Jan. 1, 2005).

167. Wang, R., Zheng, D., Yehia, G. & Tian, B. A compendium of conserved cleavage and polyadenylation events in mammalian genes. *Genome Research* **28,** 1427–1441 (Oct. 1, 2018).

168. De la Fuente, L. *Development of a bioinformatics approach for the functional analysis of alternative splicing.* PhD thesis (Universitat Politècnica de València, 2019).

169. Lupas, A., Van Dyke, M. & Stock, J. Predicting coiled coils from protein sequences. *Science* (1991).

170. Giurgiu, M. *et al.* CORUM: the comprehensive resource of mammalian protein complexes-2019. *Nucleic Acids Research* **47,** D559–D563 (Jan. 2019).

171. Consortium, T. G. O. *et al.* Gene Ontology: tool for the unification of biology. *Nature genetics* **25,** 25 (May 2000).

172. Gene Ontology Consortium. The Gene Ontology resource: enriching a GOld mine. *Nucleic Acids Research* **49,** D325–D334 (Jan. 2021).

173. Dweep, H. & Gretz, N. miRWalk2.0: a comprehensive atlas of microRNA-target interactions. *Nature Methods* **12,** 697–697 (2015).

174. Necci, M., Piovesan, D., Dosztanyi, Z. & Tosatto, S. C. MobiDB-lite: Fast and highly specific consensus prediction of intrinsic disorder in proteins. *Bioinformatics* **33,** 1402–1404 (2017).

175. Kosugi, S., Hasebe, M., Tomita, M. & Yanagawa, H. Systematic identification of cell cycle-dependent yeast nucleocytoplasmic shuttling proteins by prediction of composite motifs. *Proceedings of the National Academy of Sciences* **106,** 10171–10176 (2009).

176. Zhang, Z. *et al.* Noisy splicing, more than expression regulation, explains why some exons are subject to nonsense-mediated mRNA decay. *BMC Biology* **7,** 23 (2009).

177. Quevillon, E. *et al.* InterProScan: protein domains identifier. *Nucleic Acids Research* **33,** W116–W120 (July 1, 2005).

178. Blum, M. *et al.* The InterPro protein families and domains database: 20 years on. *Nucleic Acids Research* **49,** D344–D354 (Jan. 2021).

179. Tempel, S. Using and understanding repeatMasker. *Methods in Molecular Biology,* 29–51 (2012).

180. Biswas, A. & Brown, C. M. Scan for Motifs: a webserver for the analysis of post-transcriptional regulatory elements in the 3 untranslated regions (3 UTRs) of mRNAs. *BMC Bioinformatics* **15,** 174 (June 2014).

181. Petersen, T. N., Brunak, S., von Heijne, G. & Nielsen, H. SignalP 4.0: discriminating signal peptides from transmembrane regions. *Nature Methods* **8,** 785–786 (Oct. 29, 2011).

182. Krogh, A., Larsson, B., Von Heijne, G. & Sonnhammer, E. L. Predicting transmembrane protein topology with a hidden Markov model: Application to complete genomes. *Journal of Molecular Biology* **305,** 567–580 (2001).

183. UniProt Consortium, T. UniProt: the universal protein knowledgebase. *Nucleic Acids Research* **46,** 2699 (Mar. 2018).

184. Hornbeck, P. V. *et al.* PhosphoSitePlus, 2014: Mutations, PTMs and recalibrations. *Nucleic Acids Research* **43,** D512–D520 (D1 2015).

185. Grillo, G. *et al.* UTRdb and UTRsite (RELEASE 2010): a collection of sequences and regulatory motifs of the untranslated regions of eukaryotic mRNAs. *Nucleic Acids Research* **38,** D75–D80 (Jan. 2010).

186. Moll, P., Ante, M., Seitz, A. & Reda, T. QuantSeq 3 mRNA sequencing for RNA quantification. *Nature Methods* **11** (Dec. 2014).

187. Li, H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* **34,** 3094–3100 (May 2018).

188. Tarazona, S. *et al.* Data quality aware analysis of differential expression in RNA-seq with NOISeq R/Bioc package. *Nucleic Acids Research* **43,** gkv711 (July 16, 2015).

189. Van den Berge, K. *et al.* Observation weights unlock bulk RNA-seq tools for zero inflation and single-cell applications. *Genome Biology* **19,** 24 (Dec. 26, 2018).

190. Robinson, M. D., McCarthy, D. J. & Smyth, G. K. edgeR: A Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* **26,** 139–140 (Nov. 11, 2009).

191. Love, M. I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology* **15,** 550 (Dec. 5, 2014).

192. Vitting-Seerup, K. & Sandelin, A. IsoformSwitchAnalyzeR: analysis of changes in genome-wide patterns of alternative splicing and its functional consequences. *Bioinformatics (Oxford, England)* **35,** 4469–4471 (Nov. 1, 2019).

193. Takahashi, H., Lassmann, T., Murata, M. & Carninci, P. 5 end–centered expression profiling using cap-analysis gene expression and next-generation sequencing. *Nature Protocols* **7,** 542–561 (Mar. 2012).

194. Method of the Year 2022: long-read sequencing. *Nature Methods* **20,** 1–1 (Jan. 2023).

195. Dong, X. *et al.* Benchmarking long-read RNA-sequencing analysis tools using in silico mixtures. *Nature Methods,* 1–12 (Oct. 2023).

196. Pardo-Palacios, F. J. *et al.* Systematic assessment of long-read RNA-seq methods for transcript identification and quantification. *bioRxiv* (July 2023).

197. Wu, Y. E., Pan, L., Zuo, Y., Li, X. & Hong, W. Detecting Activated Cell Populations Using Single-Cell RNA-Seq. *Neuron* **96,** 313–329.e6 (Oct. 11, 2017).

198. Villani, A.-C. *et al.* Single-cell RNA-seq reveals new types of human blood dendritic cells, monocytes, and progenitors. *Science* **356,** eaah4573 (Apr. 21, 2017).

199. Zhong, S. *et al.* A single-cell RNA-seq survey of the developmental landscape of the human prefrontal cortex. *Nature* **555,** 524–528 (Mar. 22, 2018).

200. Crow, M. & Gillis, J. Co-expression in Single-Cell Analysis: Saving Grace or Original Sin? *Trends in Genetics* **34,** 823–831 (Nov. 1, 2018).

201. Trapnell, C. *et al.* The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature Biotechnology* **32,** 381–386 (Apr. 23, 2014).

202. Saelens, W., Cannoodt, R., Todorov, H. & Saeys, Y. A comparison of single-cell trajectory inference methods. *Nature Biotechnology* **37,** 547–554 (May 1, 2019).

203. La Manno, G. *et al.* RNA velocity of single cells. *Nature* **560,** 494–498 (Aug. 23, 2018).

204. Jia, G. *et al.* Single cell RNA-seq and ATAC-seq analysis of cardiac progenitor cell transition states and lineage settlement. *Nature Communications* **9,** 4877 (Dec. 19, 2018).

205. Su, X. *et al.* Single-cell RNA-Seq analysis reveals dynamic trajectories during mouse liver development. *BMC Genomics* **18,** 946 (Dec. 4, 2017).

206. Guo, F. *et al.* Single-cell multi-omics sequencing of mouse early embryos and embryonic stem cells. *Cell Research* **27,** 967–988 (2017).

207. Le, J. *et al.* Single-Cell RNA-Seq Mapping of Human Thymopoiesis Reveals Lineage Specification Trajectories and a Commitment Spectrum in T Cell Development. *Immunity* **52,** 1105–1118.e9 (June 16, 2020).

208. Jerber, J. *et al.* Population-scale single-cell RNA-seq profiling across dopaminergic neuron differentiation. *Nature Genetics* **53,** 304–312 (Mar. 4, 2021).

209. Qiu, X. *et al.* Single-cell mRNA quantification and differential analysis with Census. *Nature Methods* **14,** 309–315 (2017).

210. Ntranos, V., Yi, L., Melsted, P. & Pachter, L. A discriminative learning approach to differential expression analysis for single-cell RNA-seq. *Nature Methods* **16,** 163–166 (Feb. 1, 2019).

211. Wu, X., Liu, T., Ye, C., Ye, W. & Ji, G. scAPAtrap: identification and quantification of alternative polyadenylation sites from single-cell RNA-seq data. *Briefings in Bioinformatics* **2020,** 1–15 (Nov. 3, 2020).

212. Hu, Y., Wang, K. & Li, M. Detecting differential alternative splicing events in scRNA-seq with or without Unique Molecular Identifiers. *PLoS Computational Biology* **16** (ed Ma, J.) e1007925 (June 5, 2020).

213. Feng, H. *et al.* Complexity and graded regulation of neuronal cell-type–specific alternative splicing revealed by single-cell RNA sequencing. *Proceedings of the National Academy of Sciences* **118,** e2013056118 (Mar. 9, 2021).

214. Yi, L., Pimentel, H., Bray, N. L. & Pachter, L. Gene-level differential analysis at transcript-level resolution. *Genome Biology* **19,** 53 (Dec. 12, 2018).

215. Soneson, C. & Robinson, M. D. Bias, robustness and scalability in single-cell differential expression analysis. *Nature Methods* **15,** 255–261 (2018).

216. Becht, E., Zhao, E., Amezquita, R. & Gottardo, R. Aggregating transcript-level analyses for single-cell differential gene expression. *Nature Methods* **17,** 583–585 (June 1, 2020).

217. Liu, W. & Zhang, X. Single-cell alternative splicing analysis reveals dominance of single transcript variant. *Genomics* **112,** 2418–2425 (May 1, 2020).

218. Buen Abad Najar, C. F., Yosef, N. & Lareau, L. F. Coverage-dependent bias creates the appearance of binary splicing in single cells. *eLife* **9,** 1–23 (June 29, 2020).

219. Arzalluz-Luque, A., Salguero, P., Tarazona, S. & Conesa, A. acorde unravels functionally interpretable networks of isoform co-usage from single cell data. *Nature Communications* **13,** 1828 (Apr. 2022).

220. Skinnider, M. A., Squair, J. W. & Foster, L. J. Evaluating measures of association for single-cell transcriptomics. *Nature Methods* **16,** 381–386 (May 8, 2019).

221. R Core Team. *R: A language and environment for statistical computing.* Vienna, Austria, 2021.

222. Langfelder, P., Zhang, B. & Horvath, S. Defining clusters from a hierarchical cluster tree: The Dynamic Tree Cut package for R. *Bioinformatics* **24,** 719–720 (Mar. 1, 2008).

223. Zhang, X., Xu, C. & Yosef, N. Simulating multiple faceted variability in single cell RNA sequencing. *Nature Communications* **10,** 2611 (Dec. 13, 2019).

224. Pimentel, R. S., Niewiadomska-Bugaj, M. & Wang, J.-C. Association of zero-inflated continuous variables. *Statistics & Probability Letters* **96,** 61–67 (Jan. 1, 2015).

225. Lovell, D., Pawlowsky-Glahn, V., Egozcue, J. J., Marguerat, S. & Bähler, J. Proportionality: A Valid Alternative to Correlation for Relative Data. *PLOS Computational Biology* **11,** e1004075 (Mar. 2015).

226. Venables, W. & Ripley, B. *Modern Applied Statistics with S* (Springer, New York, 2002).

227. Fox, J. & Weisberg, S. *An R Companion to Applied Regression* (Sage, Thousand Oaks (CA), 2019).

228. Young, M. D., Wakefield, M. J., Smyth, G. K. & Oshlack, A. Gene ontology analysis for RNA-seq: accounting for selection bias. *Genome Biology* **11,** R14 (Feb. 4, 2010).

229. Derrick, B., White, P. & Toher, D. Parametric and non-parametric tests for the comparison of two samples which both include paired and unpaired observations. *Journal of Modern Applied Statistical Methods* **18,** 2–23 (Mar. 9, 2019).

230. Supek, F., Bošnjak, M., Škunca, N. & Šmuc, T. REVIGO Summarizes and Visualizes Long Lists of Gene Ontology Terms. *PLoS ONE* **6** (ed Gibas, C.) e21800 (July 18, 2011).

231. Chen, S. & Mar, J. C. Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data. *BMC Bioinformatics* **19,** 232 (Dec. 19, 2018).

232. Iacono, G., Massoni-Badosa, R. & Heyn, H. Single-cell transcriptomics unveils gene regulatory network plasticity. *Genome Biology* **20,** 110 (Dec. 4, 2019).

233. Pratapa, A., Jalihal, A. P., Law, J. N., Bharadwaj, A. & Murali, T. M. Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nature Methods* **17,** 147–154 (Feb. 1, 2020).

234. Grün, D., Kester, L. & van Oudenaarden, A. Validation of noise models for single-cell transcriptomics. *Nature Methods* **11,** 637–640 (2014).

235. Raj, A. & van Oudenaarden, A. Nature, Nurture, or Chance: Stochastic Gene Expression and Its Consequences. *Cell* **135,** 216–226 (Oct. 17, 2008).

236. Kim, J. K., Kolodziejczyk, A. A., Illicic, T., Teichmann, S. A. & Marioni, J. C. Characterizing noise structure in single-cell RNA-seq distinguishes genuine from technical stochastic allelic expression. *Nature Communications* **6,** 8687 (May 2015).

237. Vallejos, C. A., Richardson, S. & Marioni, J. C. Beyond comparisons of means: understanding changes in gene expression at the single-cell level. *Genome Biology* **17,** 70 (2016).

238. Zhu, K., Wang, Y., Liu, L., Li, S. & Yu, W. Long non-coding RNA MBNL1-AS1 regulates proliferation, migration, and invasion of cancer stem cells in colon cancer by interacting with MYL9 via sponging microRNA-412-3p. *Clinics and Research in Hepatology and Gastroenterology* **44,** 101–114 (Feb. 1, 2020).

239. Lee, K.-Y., Chang, H.-C., Seah, C. & Lee, L.-J. Deprivation of Muscleblind-Like Proteins Causes Deficits in Cortical Neuron Distribution and Morphological Changes in Dendritic Spines and Postsynaptic Densities. *Frontiers in Neuroanatomy* **13,** 75 (2019).

240. Wang, P.-Y., Chang, K.-T., Lin, Y.-M., Kuo, T.-Y. & Wang, G.-S. Ubiquitination of MBNL1 Is Required for Its Cytoplasmic Localization and Function in Promoting Neurite Outgrowth. *Cell Reports* **22,** 2294–2306 (Feb. 27, 2018).

241. Sta Maria, N. S. *et al.* Mbnl1 and Mbnl2 regulate brain structural integrity in mice. *Communications Biology* **4,** 1342 (Nov. 30, 2021).

242. Bray, N. The power of 3 UTRs. *Nature Reviews Neuroscience* **19,** 319 (June 1, 2018).

243. Bae, B. & Miura, P. Emerging roles for 3' UTRs in neurons. *International Journal of Molecular Sciences* **21,** 3413 (May 2, 2020).

244. Sandberg, R., Neilson, J. R., Sarma, A., Sharp, P. A. & Burge, C. B. Proliferating Cells Express mRNAs with Shortened 3' Untranslated Regions and Fewer MicroRNA Target Sites. *Science* **320,** 1643–1647 (June 2008).

245. MacNicol, M. C., Cragle, C. E. & MacNicol, A. M. Context-dependent regulation of Musashi-mediated mRNA translation and cell cycle regulation. *Cell Cycle* **10,** 39–44 (Jan. 1, 2011).

246. Okano, H., Imai, T. & Okabe, M. Musashi: a translational regulator of cell fate. *Journal of Cell Science* **115,** 1355–1359 (Apr. 1, 2002).

247. Li, H. J., Haque, Z. K., Chen, A. & Mendelsohn, M. RIF-1, a novel nuclear receptor corepressor that associates with the nuclear matrix. *Journal of Cellular Biochemistry* **102,** 1021–1035 (2007).

248. Tang, S. J., Meulemans, D., Vazquez, L., Colaco, N. & Schuman, E. A role for a rat homolog of staufen in the transport of RNA to neuronal dendrites. *Neuron* **32,** 463–475 (Nov. 8, 2001).

249. Gleghorn, M. L., Gong, C., Kielkopf, C. L. & Maquat, L. E. Staufen1 dimerizes through a conserved motif and a degenerate dsRNA-binding domain to promote mRNA decay. *Nature Structural & Molecular Biology* **20,** 515–524 (Apr. 2013).

250. McInnes, L., Healy, J. & Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv* (Sept. 17, 2020).

251. Masood, M., Grimm, S., El-Bahrawy, M. & Yagüe, E. TMEFF2: A Transmembrane Proteoglycan with Multifaceted Actions in Cancer and Disease. *Cancers* **12,** E3862 (Dec. 21, 2020).

252. Jen, Y.-H. L., Musacchio, M. & Lander, A. D. Glypican-1 controls brain size through regulation of fibroblast growth factor signaling in early neurogenesis. *Neural Development* **4,** 33 (Sept. 4, 2009).

253. Fu, X. D. & Ares, M. Context-dependent control of alternative splicing by RNA-binding proteins. *Nature Reviews Genetics* **15,** 689–701 (Oct. 11, 2014).

254. Saha, A. *et al.* Co-expression networks reveal the tissue-specific regulation of transcription and splicing. *Genome Research* **27,** 1843–1858 (Nov. 1, 2017).

255. Aghamirzaie, D., Collakova, E., Li, S. & Grene, R. CoSpliceNet: a framework for co-splicing network inference from transcriptomics data. *BMC Genomics* **17,** 845 (Dec. 28, 2016).

256. Zhang, P., Southey, B. R. & Rodriguez-Zas, S. L. *Co-expression networks uncover regulation of splicing and transcription markers of disease* in *EPiC Series in Computing* **70** (2020), 119–128.

257. Chau, K. K. *et al.* Full-length isoform transcriptome of the developing human brain provides further insights into autism. *Cell Reports* **36,** 109631 (Aug. 2021).

258. Ma, J. *et al.* Comprehensive expression-based isoform biomarkers predictive of drug responses based on isoform co-expression networks and clinical data. *Genomics* **112,** 647–658 (Jan. 1, 2020).

259. Ma, J.-Q. *et al.* Differential Alternative Splicing Genes and Isoform Regulation Networks of Rapeseed (Brassica napus L.) Infected with Sclerotinia sclerotiorum. *Genes* **11,** 784 (July 13, 2020).

260. Benegas, G., Fischer, J. & Song, Y. S. Robust and annotation-free analysis of alternative splicing across diverse cell types in mice. *eLife* **11** (eds Eyras, E. & Manley, J. L.) e73520 (Mar. 2022).

261. Morabito, S., Reese, F., Rahimzadeh, N., Miyoshi, E. & Swarup, V. hdWGCNA identifies co-expression networks in high-dimensional transcriptomics data. *Cell Reports Methods* **3,** 100498 (June 2023).

Appendix I

Documentation for the acorde R package

I.1 Introduction

This appendix includes the vignettes that document the different functionalities included in the acorde R package (see Chapter 5), which are available at https://github.com/ConesaLab/acorde. The first vignette, *Step-by-step guide to the acorde pipeline*, constitutes the main user manual for the software, including how to install the package, set up dependencies and prepare input data, as well as how to run each of the steps of the pipeline. Importantly, a section has also been included explaining how to use the output of acorde to generate tappAS-compatible inputs for functional analysis. The second vignette, *Simulating co-expression in pre-simulated scRNA-seq data with acorde*, explains how to run the simulation strategy outlined in Chapter 5, section 5.2.4. Besides ensuring the usability of the software, the information contained in these documents may also be used to reproduce the results shown in said chapter of the present thesis.

Step-by-step guide to the acorde pipeline

Ángeles Arzalluz-Luque [1]

[1]Institute for Integrative Systems Biology (I2SysBio), Valencia, Spain

Last updated: 15 junio, 2023

Package

acorde 0.1.0

Contents

1 Introduction

The **acorde** R package contains the necessary functions to reproduce the pipeline in this paper, a study by *Arzalluz-Luque et al.* in which we analyze networks of isoform co-usage using single-cell RNA-seq data (scRNA-seq).

The pipeline includes three basic analysis blocks:

1. **Single-cell isoform quantification and filtering**. First, bulk long read data is used to generate tissue-specific transcript models. Short-read scRNA-seq data is then used for isoform quantification, and isoforms are filtered according to their **Differential Expression** (DE) status across multiple cell types.
2. **Detection of isoform co-expression**. *acorde* includes the implementation of percentile correlations, a novel strategy to obtain noise-robust correlation estimates from scRNA-Seq data, and a semi-automated clustering approach to detect modules of co-expressed isoforms acorss cell types.
3. **Differential and co-Differential Isoform Usage analysis**. DIU and co-DIU analysis are designed to leverage the multiple cell types contained in single-cell datasets, and enable the detection of genes that show isoform expression coordination. To couple these analysis with a biologically interpretable readout, we incorporate functional annotations onto isoform models, and use tappAS for functional analysis.

Since both the long read-transcriptome definition procedure and the functional analyses in [1] are based on external tools, the present R package does **not** incorporate neither of these two analysis steps. Instead, acorde contains the necessary functions and documentation to obtain a set of DIU and co-DIU genes using an single-cell, isoform-level expression matrix as input.

In addition, we provide all the necessary instructions to reproduce the figures and additional analyses included in Arzalluz-Luque et al. [1], and provide the isoform expression matrix employed during the study as internal data in the package.

2 Installation

Acorde can be installed from GitHub using `devtools`:

```
install.packages("devtools")
devtools::install_github("ConesaLab/acorde", build_vignettes = TRUE)
```

3 Getting ready

To run the analyses in this vignette, you'll first need to load `acorde`:

```
# load acorde
library(acorde)

# load auxiliary packages
suppressPackageStartupMessages({
  library(dplyr)
  library(tibble)
  library(purrr)
  library(furrr)
  library(ggplot2)
  library(SingleCellExperiment)
})
```

In addition, we'll require some additional packages for data handling and formatting. Most of them are signaled as `acorde` dependencies, so they will already be installed in your system.

To generate plots, we make use of the `cowplot` R package and the cowplot theme. After install:

```
install.packages("cowplot")
```

. . . you can load and set the theme of your R session as follows:

```
library(cowplot)
theme_set(theme_cowplot())
```

4 Input data

The acorde pipeline requires a **single-cell isoform expression** matrix as input. Single-cell isoform counts should be provided in the form of a `data.frame` or `tibble` object including isoforms as rows and cells as columns. Isoform identifiers can be supplied as `rownames()` or as an additional identifier column, as required by tibble.

To generate an isoform-level single-cell expression matrix, we first processed long read bulk data from ENCODE (provided by Wyman et al. [2]) to build a mouse neural transcriptome, and then used publicly-available scRNA-seq data by Tasic et al. [3] to quantify the expression of long read-defined isoforms in mouse neural cell types. Details to this process can be found in our manuscript (see Supplementary Note and Methods).

If you wish to reproduce the analyses in Arzalluz-Luque et al. [1]), you can load the `tasic` object to use our isoform-quantified dataset:

```
# load Tasic dataset
data("tasic")

# load Tasic metadata
data("metadata")

# use metadata to create a cell - cell type identity table
id_table <- metadata %>%
    select(run, cell_type) %>%
    dplyr::rename(cell = "run")
```

These contain two `tibble` objects. After quality control (see Methods in Arzalluz-Luque et al. [1]), the `tasic` tibble contains expression data for **16240 isoforms** and **1591 cells** belonging to 7 neural cell types:

Cell type	Number of cells
Astrocyte	43
Endothelial Cell	29
GABA-ergic Neuron	729
Glutamatergic Neuron	711
Microglia	21
Oligodendrocyte	37
Oligodendrocyte Precursor Cell	21

```r
# display format of tasic
tasic[1:6, 1:8]
#> # A tibble: 6 x 8
#>   transcript   SRR2138606 SRR2138607 SRR2138609 SRR2138610 SRR2138612 SRR2138614
#>   <chr>             <dbl>      <dbl>      <dbl>      <dbl>      <dbl>      <dbl>
#> 1 NM_00100200~      0           2.21    7.74          46.5       12.6       0
#> 2 XM_00653431~      0           0         0             0        36.8       0.878
#> 3 NM_00103330~      0          27.8    1195.         693.        16.0      28.6
#> 4 NM_00134651~    324.          0         0             0          0        0
#> 5 NM_00104010~      0          14.7       0.378      54.1          0       23.1
#> 6 XM_01732159~      0.362       0         0        2013.       343.      270.
#> # i 1 more variable: SRR2138615 <dbl>

# number of cells and isoforms
dim(tasic)
#> [1] 16240  1592
```

Metadata is contained in the `metadata` tibble. This table was generated using Tasic et al. supplementary files, which were used to parse cell type labels for single-cell IDs (i.e. sequencing run IDs, included in the `run` column), among other information:

```r
# show information contained in metadata
metadata %>% colnames
#>  [1] "run"            "GEO_accession"  "passes_QC"      "input_material"
#>  [5] "mouse_line"     "dissection"     "animal_ID"      "sample_name"
#>  [9] "cell_type"      "subtype"        "cell_type.mod"

# display cell type labels
unique(metadata$cell_type)
#> [1] "Glutamatergic Neuron"           "GABA-ergic Neuron"
#> [3] "Endothelial Cell"               "Astrocyte"
#> [5] "Microglia"                      "Oligodendrocyte"
#> [7] "Oligodendrocyte Precursor Cell"
```

See `?tasic` and `?metadata` for details.

5 Isoform Differential Expression across multiple cell types

To select isoforms with robust co-variation across the 7 neural cell types, we first applied **multi-group Differential Expression analysis**, which will detect isoforms that are differentially expressed (DE) in at least one cell type.

To achieve this, we combined the zero-weighting strategy in the zinbwave R package with bulk-designed DE methods DESeq2 and edgeR. Both tools were set to detect DE across multiple groups. The incorporation of weights to these analyses and the correct application of both tools to scRNA-seq data was done following the Differential Expression section in the zinbwave vignette.

Acorde provides the `cell_type_DE()` function, which constitutes a wrapper to these two methodologies. This function takes a `SingleCellExperiment` object as input:

```r
# convert tibble to count matrix
count.matrix <- column_to_rownames(tasic, "transcript") %>%
  as.matrix
# round estimated counts from RSEM to generate integer values
count.matrix <- count.matrix %>% round

# create SingleCellExperiment object with Tasic data
sce <- SingleCellExperiment(assays = list(counts = count.matrix,
                                          logcounts = log2(count.matrix + 1)),
                            colData = metadata)
```

By default, `cell_type_DE()` automatically calculates and stores zinbwave weights in the `weights` slot of the `SingleCellExperiment` object. Alternatively, you may set `cell_type_DE(compute_weights = FALSE)` and run `zinbwave()` yourself (note that computing weights is a computationally costly step, so we'll use the `BiocParallel` R package to parallelize the process):

```r
# load biocParallel
library(BiocParallel)

# compute weights
library(zinbwave)

sce <- zinbwave(sce, observationalWeights = TRUE,
                BPPARAM = MulticoreParam(6))
```

Now we are ready to run isoform-level Differential Expression analysis. Set `cell_type_DE(method = "both")` to be able to compare multi-group DE results for both edgeR and DESeq2, or choose either `"edgeR"` or `"DESeq2"` to run just one DE analysis.

```r
# run DE analysis using both DESeq2 and edgeR
de_results <- cell_type_DE(sce, AdjPvalue = 0.05,
                           mode = "both",
                           compute_weights = FALSE)
```

In the acorde manuscript, a downsampling strategy was used to balance cell type abundances between neural and glial cell types. Briefly, 50 runs of random sampling were performed to select 45 GABA and 45 glutamatergic neuron cells, and both edgeR and DESeq2 were run to obtain DE isoforms using each of the 50 downsampled versions of the data. Next, isoforms detected to be significantly DE by *at least one of the methods* in >50% of the runs were considered to be DE.

We hereby provide the **DE consensus set** that was used in the acorde paper to fully ensure the reproducibility of our results. From here on out, this vignette will display the results of using this DE isoform set:

```
# load consensus set of DE isoforms
data("consensus_DE_set")

# filter expression matrix
tasic_de <- tasic %>%
  filter(transcript %in% consensus_DE_set$transcript)
```

5.1 Other isoform filtering criteria

In addition to DE filtering, several functions are provided in acorde to control for other expression-related biases and remove isoforms prior to downstream analysis (or during quality control):

- `detect_sparse()` flags isoforms that have high proportion of zeros in all cell types. Isoforms must have non-zero expression in a proportion higher than the supplied threshold in at least one cell type.
- `detect_minor_isoforms()` flags isoforms with low gene-relative expression across cell types. Isoforms that represent a small proportion of the total gene's expression will be flagged as minor.
- `detect_low_expression()` flags isoforms with low mean/median counts across cell types.

For our manuscript, we removed minor isoforms using a 10% gene-relative expression threshold:

```
# detect minor isoforms
minor <- detect_minor_isoforms(tasic_de, id_table = id_table,
                               gene_tr_table = gene_tr_ID,
                               gene_expr_proportion = 0.1,
                               isoform_col = "transcript")

head(minor)
#> # A tibble: 6 x 2
#>   transcript     minor_isoform
#>   <chr>          <lgl>
#> 1 NM_001001980.3 FALSE
#> 2 NM_001004367.4 FALSE
#> 3 NM_001012623.2 FALSE
#> 4 NM_001012625.2 FALSE
#> 5 NM_001015507.2 FALSE
#> 6 NM_001017426.2 FALSE

# summary of minor isoforms detected
table(minor$minor_isoform)
#>
#> FALSE  TRUE
#>  8306   969

# get isoforms that were not flagged as minor
excl_minor <- filter(minor, minor_isoform == FALSE)

# apply filter
```

```
tasic_sp <- tasic_de %>%
  filter(transcript %in% excl_minor$transcript)
```

Next, isoforms were removed if they belonged to genes with a single DE isoform since, in practice, no differential splicing can occur for single-isoform genes (or for genes for which only one isoform presents expression variation across cell types). Isoform - gene correspondence is provided by acorde in the `gene_tr_ID` data object.

```
# load and display gene-isoform table
data("gene_tr_ID")
head(gene_tr_ID)
#> # A tibble: 6 x 2
#>   transcript     gene
#>   <chr>          <chr>
#> 1 NM_001012623.2 Rims1
#> 2 NM_001012625.2 Rims1
#> 3 NM_001083121.2 Enah
#> 4 NM_001199003.1 Rgs7
#> 5 NM_001311166.1 Edem3
#> 6 NM_001347195.1 Rgs7

# remove transcripts from single-isoform genes
gspliced <- tasic_sp %>%
  select(transcript) %>%
  left_join(gene_tr_ID, by = "transcript") %>%
  group_by(gene) %>%
  filter(n() > 1)

tasic_sp <- tasic_sp %>%
  filter(transcript %in% gspliced$transcript)
```

As a summary, here's a comparison of the number of isoforms remaining in our expression matrix after performing these filtering steps:

Object	Content	Isoform_no
tasic	All isoforms	16240
tasic_de	DE isoforms	9275
tasic_sp	DE isoforms from multi-isoform genes	6794

6 Computing isoform co-expression using percentile correlations

To detect isoform co-expression across the 7 neural cell types in the Tasic dataset, we will apply **percentile correlations**. Percentile correlations, as described in [1]), are a metric designed to overcome cell-to-cell effects that generate noise and mask the co-expression signal in the data, yielding low correlation values when using traditional correlation metrics.

Instead, percentile correlations are based on a percentile-summarizing strategy in which cells of the same cell-type are used to estimate a cell type-specific expression distribution for each isoform and **cell-level counts are replaced by percentile values**. Then, Pearson correlations between isoforms are computed using this percentile-summarized expression. More details can be found in Arzalluz-Luque et al. [1]).

Acorde includes the `percentile_cor()` function to compute percentile correlations, which takes a cell - cell-type correspondence table (`id_table` defined above) and the expression matrix as input, and generates an isoform-to-isoform correlation matrix:

```
# compute percentile correlations
cors <- percentile_cor(tasic_sp,
                       id_table = id_table,
                       percentile_no = 10,
                       isoform_col = "transcript")

cors[1:4, 1:4]
#>                  NM_001033304.2 NM_001346518.1 NM_001040106.2 XM_017321597.2
#> NM_001033304.2      1.0000000      0.8659368      0.5790838      0.8680737
#> NM_001346518.1      0.8659368      1.0000000      0.7007998      0.9458788
#> NM_001040106.2      0.5790838      0.7007998      1.0000000      0.7615355
#> XM_017321597.2      0.8680737      0.9458788      0.7615355      1.0000000
```

By default, `percentile_cor()` summarizes isoform expression into 10 percentile values (deciles) per cell type, although users may supply any number between 4 (quantiles) and 100 (percentiles). If a tibble is supplied, users will need to specify the column name in which isoform identifiers are provided in order for `percentile_cor()` to successfully return isoform IDs as column and row names in the correlation matrix.

7 Semi-supervised isoform clustering

The correlation matrix generated by `percentile_cor()` can be used to detect groups of isoforms with similar expression patterns across and within cell types, given that percentile correlation captures not only the similarities in expression patterns among the cell types, but also the agreement in expression "behavior" of the isoforms in each of the cell types.

Acorde includes a series of functions for clustering and cluster refinement that, when combined, provide a flexible framework to obtain modules of co-expressed isoforms. Initial clustering is based on the `cutreeHybrid()` function from the dynamicTreeCut package. **dynamicTreeCut** is a hierarchical clustering algorithm based on the selection of optimal cut heights for different branches of the dendrogram, instead of applying the same fixed threshold to separate elements into clusters.

7.1 Initial dynamic clustering with `cluster_isoforms()`

We will first run the `cluster_isoforms()` wrapper function, which takes a correlation matrix, generates the necessary inptus and runs `cutreeHybrid` under the hood:

```
clusters <- cluster_isoforms(cors, deepSplit = 4, pamStage = FALSE,
                             minClusterSize = 20)
#> Inferring dendrogram via hclust()...
#> Creating clusters dynamically via cutreeHybrid()...
#>  ..cutHeight not given, setting it to 0.551  ===>  99% of the (truncated) height range in dendro.
#>  ..done.

# show number of clusters
length(clusters)
#> [1] 166
```

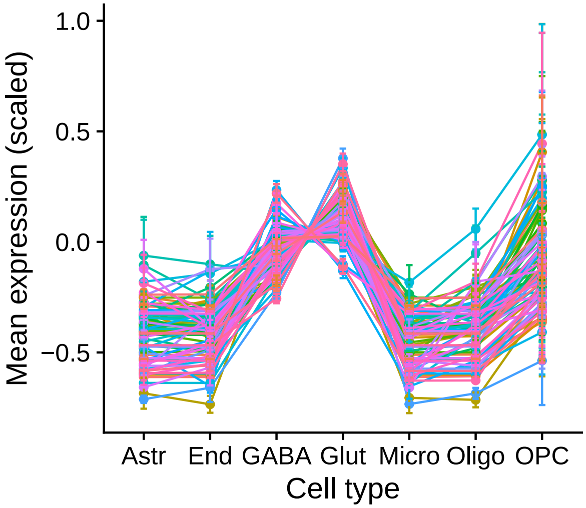
Briefly, `deepSplit` ranges between 0 and 4, and provides smaller, more accurate clusters when set to high values. Setting `pamStage = FALSE` allows return of unassigned items, which are placed on the first element of the `clusters` list. Finally, `minClusterSize` determines the minimum size of the produced clusters.

In our study, we set these parameters in order to maximize the similarity between isoforms assigned to the same cluster, regardless of the high number of clusters obtained. This configuration was selected because acorde includes a series of steps in the pipeline to **refine and merge** some of these clusters. These are designed to improve the expression signal while minimizing redundancies in the expression profile that they represent. However, if users want to run clustering using their own parameter setup, `cluster_isoforms()` can pass any additional parameters supplied to `cutreeHybrid()`.

In spite of being more flexible than regular hierarchical clustering, the dynamicTreeCut algorithm can also generate inconsistent isoform assignments to clusters, i.e. group isoforms with rather different expression profiles.

We'll now use two of the **cluster visualization** functions in acorde to view *cluster 6* in `clusters` (`clusters[[7]]`, given that the first element of the list corresponds to unclustered isoforms) as an example of this. Acorde provides a function, `calculate_cluster_ctmeans()`, to compute the mean and standard error of cell type expression for each of the isoforms in a cluster. In this manner, the similarities of the expression profiles across cell types can be easily compared for same-cluster isoforms. The output of `calculate_cluster_ctmeans` can be directly provided to `plot_cluster_ctmeans()` to generate a visual summary of all isoforms in a cluster:

```
# scale isoform expression
tasic_scaled <- scale_isoforms(tasic_sp, method = "classic",
                               isoform_col = "transcript")

# calculate cell type mean expression for all isoforms
example_means <- calculate_cluster_ctmeans(tasic_scaled,
                                           isoform_ids = clusters[[7]],
                                           id_table = id_table,
                                           isoform_col = "transcript")

# plot isoform-level means for all isoforms in the cluster
ctlabs <- c("Astr", "End", "GABA", "Glut", "Micro", "Oligo", "OPC")

plot_cluster_ctmeans(example_means, ct_labels = ctlabs)
```



7.2 Cluster filtering with `filter_clusters()`

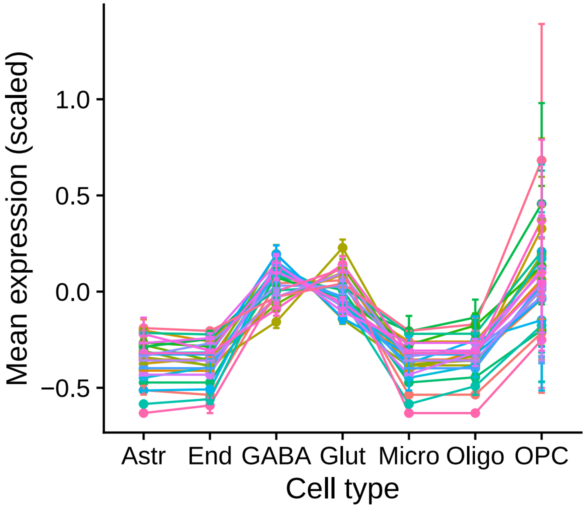
Some of the isoforms in *cluster 6* may have an expression pattern that is slightly different to the rest of the members of the cluster. To solve this, we can use the `filter_clusters()` function, which will move isoforms to the unclustered groups if they are poorly correlated with most of the isoforms in the cluster.

```
# see current number of unclustered isoforms
clusters[[1]] %>% length
#> [1] 634

# run filter_clusters
clusters_filt <- filter_clusters(clusters, cor_matrix = cors,
                                 min_cor = 0.9, lowcor_threshold = 2,
                                 contains_unclustered = TRUE,
                                 size_filter = TRUE, size_threshold = 10)

# see number of unclustered isoforms after filtering
clusters_filt[[1]] %>% length
#> [1] 4699

# number of isoforms remaining in clusters
clusters_filt[2:length(clusters_filt)] %>% map_int(length) %>% sum
#> [1] 2095
```

In this step, isoforms will be removed from a cluster if they have correlation values below `min_cor` with other members of the cluster. `lowcor_threshold` provides the maximum number of correlation values lower than `min_cor` that are allowed per isoform. In addition, `size_filter` and `size_threshold` can be used to discard clusters by their size, moving isoforms in clusters that are too small to the unclustered group.

As a result, many of the isoforms that were initially input for clustering are currently not assigned, and we have successfully cleaned the signal of our initial set of clusters. *cluster 6* (now in position 6 of the `clusters_filt` list due to removal of clusters below the size threshold) now looks like this:

```
# calculate cell type mean expression for all isoforms
example_means.filt <- calculate_cluster_ctmeans(tasic_scaled,
                                                 isoform_ids = clusters_filt[[6]],
                                                 id_table = id_table,
                                                 isoform_col = "transcript")

# plot isoform-level means for all isoforms in the cluster
plot_cluster_ctmeans(example_means.filt, ct_labels = ctlabs)
```



7.3 Assigning unclustered isoforms to clusters with `expand_clusters()`

Next, we will use the `expand_clusters()` function in acorde to join unclustered isoforms to their most similar cluster. In this process, each cluster's profile is first summarized into a synthetic representative transcript that we named *metatranscript*. Metatranscripts are calculated as the mean of the percentile-summarized expression of all isoforms in the cluster. Then, the function computes percentile correlations between isoforms and cluster metatranscripts.

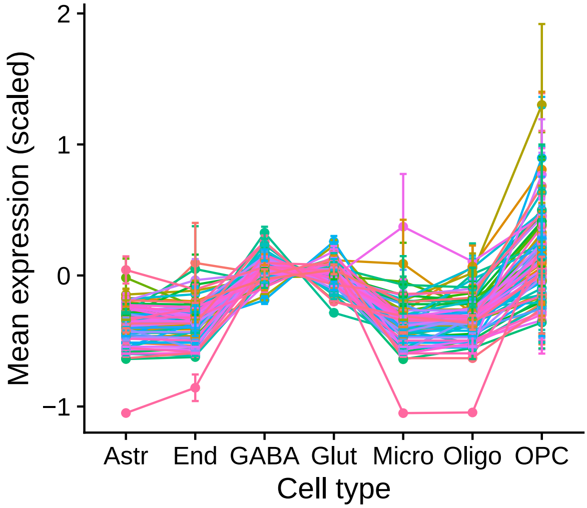
In our study, we assigned unclustered isoforms to a cluster if they showed correlation > 0.9 with its metatranscript (and the maximally correlated cluster was selected as the best match if there were ties).

```
# first round, expand using hard correlation threshold
clusters_expanded <- expand_clusters(tasic_sp, isoform_col = "transcript",
                                     id_table = id_table,
                                     cluster_list = clusters_filt[2:length(clusters_filt)],
                                     unclustered = clusters_filt[[1]],
                                     force_expand = FALSE,
                                     expand_threshold = 0.9,
                                     method = "percentile")

# show output format
names(clusters_expanded)
#> [1] "unclustered" "expanded"
map_chr(clusters_expanded, class)
#> unclustered    expanded
#> "character"      "list"
map_int(clusters_expanded, length)
#> unclustered    expanded
#>        1942          67

# check number of unclustered isoforms after first round
length(clusters_expanded$unclustered)
#> [1] 1942
```

The correlation threshold used to assign an isoform to a cluster can be adjusted via the `expand_threshold` parameter. To simplify, users may set `force_expand = TRUE` to assign isoforms to the cluster reporting the highest correlation. In this case, `expand_threshold` will be ignored.

We can now check the effect of cluster expansion on our example cluster, *cluster 6*. Note that after expansion, unclustered isoforms are now assigned to the `unclustered` list element, while the list containing the clusters is situated in the `expanded` slot. Therefore, *cluster 6* is now `clusters_expanded$expanded[[5]]`:

```
# compare cluster sizes
  # before expansion
  clusters_filt[[6]] %>% length
#> [1] 26

  # after expansion
  example_expanded <- clusters_expanded$expanded[[5]]
  example_expanded %>% length
#> [1] 82


# calculate cell type mean expression for all isoforms in cluster
example_means.exp <- calculate_cluster_ctmeans(tasic_scaled,
                                               isoform_ids = example_expanded,
                                               id_table = id_table,
                                               isoform_col = "transcript")

# plot isoform-level means for all isoforms in the cluster
plot_cluster_ctmeans(example_means.exp, ct_labels = ctlabs)
```



7.4 Eliminating redundancies across cluster profiles with `merge_clusters()`

At this point, we have focused on the reduction of within-cluster variability, which results in a large number of small, redundant clusters. To mitigate this, acorde includes the `merge_clusters()` function, which detects expression profile similarities across clusters (redundancy) and merges them into a single cluster. Briefly, `merge_clusters` employs a clustering approach using the metatranscripts for computed clusters as input, which is done via regular hierarchical clustering (by default) or using the dynamic approach implemented in the `cutreeHybrid()` function from the dynamicTreeCut package (when `dynamic = TRUE` is set).

For our study, we run regular hierarchical clustering of metatranscripts via the `dynamic = FALSE` parameter. When set to `TRUE`, this function can perform dynamic clustering for cluster merge and passes arguments to `cutreeHybrid()`. In this case, however, we just set `height_cutoff = 0.1` as non-default parameters for `merge_clusters()` to pass on to R stats function `cutree()`:

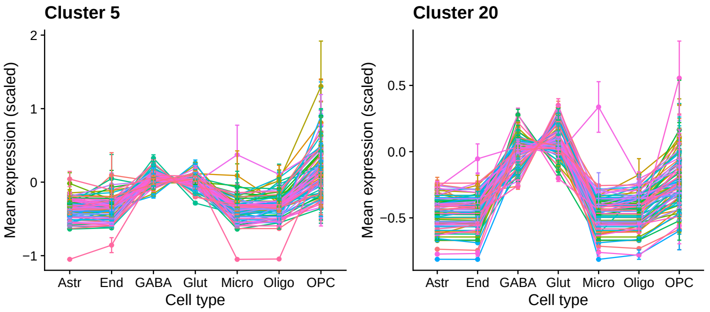
```
merge.output <- merge_clusters(tasic_sp, id_table = id_table,
                               cluster_list = clusters_expanded$expanded,
                               method = "percentile",
                               dynamic = FALSE,
                               height_cutoff = 0.1,
                               isoform_col = "transcript")
```

`merge_clusters()` returns a nested list including two elements: first, a list containing the merged cluster indices, allowing traceback of all merged decisions; and second, a list containing the merged clusters obtained as a result.

```
# show output format
map_chr(merge.output, class)
#> merged_groups      clusters
#>        "list"        "list"
map_int(merge.output, length)
#> merged_groups      clusters
#>            26            26

# retrieve outputs
merged_groups <- merge.output[[1]]
clusters_merged <- merge.output[[2]]

# show merge decision tree
head(merged_groups)
#> $`1`
#> [1]  1  2  6 21
#>
#> $`2`
#> [1]  3  4  8 27 34 36
#>
#> $`3`
#> [1]  5 20
#>
#> $`4`
#> [1] 7
#>
#> $`5`
#> [1]  9 13 44 51
#>
#> $`6`
#> [1] 10 15

# show merged cluster formats
map(head(clusters_merged), head, 4)
#> $`1`
#>               11               12               13               14
#> "NM_001346518.1" "XM_006514310.3" "XR_001778696.2" "XM_006538935.2"
#>
#> $`2`
#>               31               32               33               34
#> "NM_001033304.2" "XM_006512556.2"    "NR_027907.1"      "PB.1034.1"
#>
#> $`3`
#>               51               52               53               54
#>    "XR_386367.2" "XM_011241643.3"     "PB.10772.5"     "PB.11344.3"
#>
#> $`4`
#>           71            72            73            74
#>  "PB.10330.2" "PB.11604.12" "PB.11604.28"  "PB.13168.1"
#>
#> $`5`
#>           91           92           93           94
#> "PB.10081.2" "PB.10599.3" "PB.11122.2" "PB.11848.3"
#>
#> $`6`
#>          101          102          103          104
#> "PB.11505.3" "PB.1155.17" "PB.11784.4" "PB.14081.8"
```

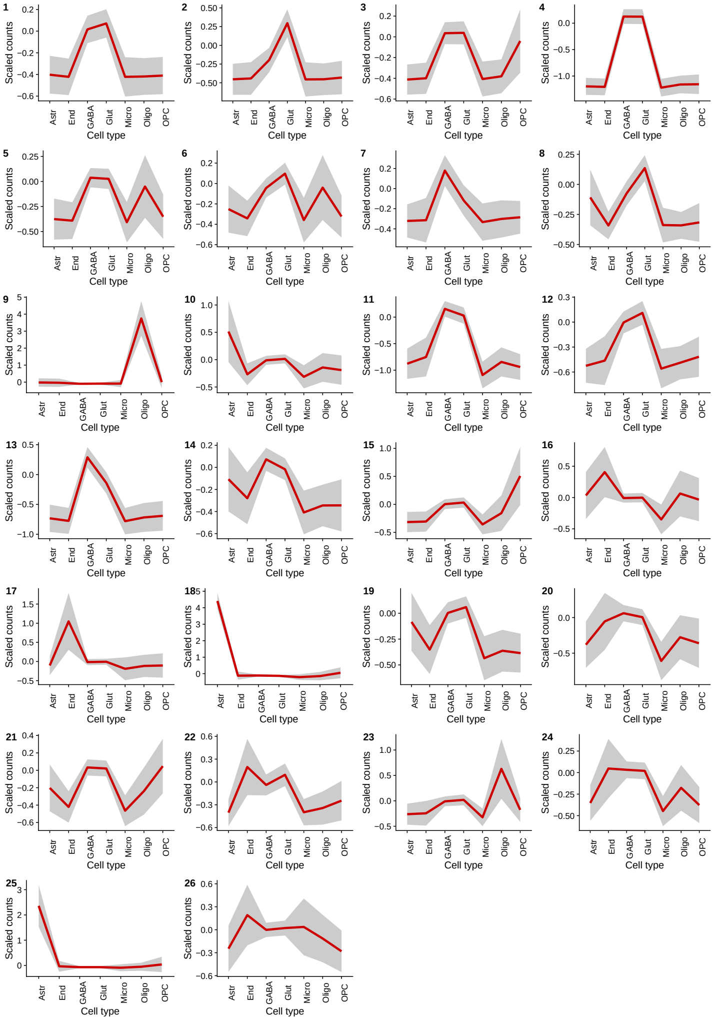
After merge, the number of clusters has been greatly reduced. We can now check the results of any of the cluster merge decisions made by `merge_clusters` to verify that they are correct. Let's take, for instance, `merged cluster 3`:

```
# plot an example of clusters that have been merged together
example_group <- merged_groups[[3]]

example_group
#> [1]  5 20

# compute cell type mean expression for merged clusters
merge_check <- clusters_expanded$expanded[example_group] %>%
  map(~calculate_cluster_ctmeans(tasic_scaled,
                                 isoform_ids = .,
                                 id_table = id_table,
                                 isoform_col = "transcript"))

# create plots and plot grid
merge_check_plots <- seq(1, length(merge_check)) %>%
  map(~plot_cluster_ctmeans(merge_check[[.]],
                            plot_title = paste("Cluster", example_group[.]),
                            ct_labels = ctlabs))

plot_grid(plotlist = merge_check_plots)
```



To enable a more summarized and elegant view of cluster profiles than that generated by `plot_cluster_ctmeans()`, acorde also includes a two functions, `calculate_cluster_profile()` and `plot_cluster_profile()`. These will average cell type mean values for all transcripts and compute the standard deviation. In this manner, we can better evaluate similarities among cluster members, and see whether the patterns represented by clusters are truly distinct or still contain redundancies.

```
# compute cluster mean and standard deviations
patterns_merged <- map(clusters_merged,
                       ~calculate_cluster_profile(tasic_scaled,
                                   isoform_ids = .,
                                   id_table = id_table,
                                   isoform_col = "transcript"))

patterns_merged[[1]]
#> # A tibble: 7 x 4
#>   tr            mean    sd cell_type
#>   <chr>        <dbl> <dbl> <chr>
#> 1 silhouette -0.402  0.175 Astrocyte
#> 2 silhouette -0.422  0.168 Endothelial Cell
#> 3 silhouette  0.0161 0.127 GABA-ergic Neuron
#> 4 silhouette  0.0714 0.131 Glutamatergic Neuron
#> 5 silhouette -0.422  0.182 Microglia
#> 6 silhouette -0.419  0.170 Oligodendrocyte
#> 7 silhouette -0.410  0.172 Oligodendrocyte Precursor Cell

# generate plots using output from calculate_cluster_profile()
pattern_plots_merged <- map(patterns_merged, plot_cluster_profile,
                                ct_labels = ctlabs)

# plot a grid with all clusters
plot_grid(plotlist = pattern_plots_merged,
              labels = seq(1, length(pattern_plots_merged)), ncol = 4)
```



8 Additional steps to improve clustering results

As shown above, clustering of metatranscripts generally works well for removing redundancies. However, clustering results can be improved by further combining acorde functions. For our manuscript, we performed a series of cluster refinement steps to assign isoforms that remained unclustered and generate fewer, more accurate clusters.

8.1 Clusters with noisy profiles

First, among the clusters generated as a result of unsupervised clustering, there are a few that show strong variability in comparison to the global cluster mean, given the broad standard deviation ribbon in some of the panels in the plot above above. In particular, we tackled **clusters 7, 12 and 14**. Note that, for cases where clusters were noisy but represented a unique profile (i.e. an expression pattern across cell types not captured by any other cluster, for instance, cluster 26), we did not perform this step to avoid removing the expression pattern entirely.

To mitigate this, we decided to join isoforms in clusters that had a visibly inaccurate (i.e. noisy) profile to those that remained unclustered:

```
# select noisy clusters
noisy_idx <- c(7, 12, 14)

# join all non-assigned isoforms into one list
unclustered <- c(clusters_expanded$unclustered,
                 unlist(clusters_merged[noisy_idx]))

# remove noisy clusters from clusters_merged list
clusters_merged <- clusters_merged[-noisy_idx]

length(unclustered)
#> [1] 2381
```

8.2 Refinement of merged clusters

Merging clusters may create discrepancies between same-cluster isoforms, since they were originally assigned to different clusters. To make clusters more homogeneous for downstream assignment of unclustered isoforms, we run `filter_clusters()` again with the following parameters:

```
# run filter
clusters_merged.filtered <- filter_clusters(clusters_merged, cor_matrix = cors,
                                            min_cor = 0.7, lowcor_threshold = 1,
                                            contains_unclustered = FALSE,
                                            size_filter = TRUE,
                                            size_threshold = 10)

# join unclustered to the rest of isoforms removed by the filter
unclustered <- c(clusters_merged.filtered[[1]],
                 unclustered)

# view total no. of unclustered isoforms
length(unclustered)
#> [1] 3039

# check total no. of clusters (first slot contains only unclustered)
length(clusters_merged.filtered)
#> [1] 24
```

Note that clusters containing less than 10 isoforms (as defined by the `size_threshold`) parameter will be discarded and their isoforms moved to the unclustered group.

8.3 Assignment of remaining unclustered isoforms

We next assigned all unclustered isoforms to the remaining 23 clusters. Given that there is a high number of isoforms with no cluster assigned, we used a recursive assignment strategy in which the correlation threshold used for grouping decreased after each iteration. In this manner, isoforms with high similarities with the cluster profile will be assigned first, contributing to strengthen the profile and helping drive the assignment in the next iteration.

```r
# define threshold vector
thres <- c(0.9, 0.8, 0.7)

clusters_merged.expanded <- list()
clusters_merged.expanded$unclustered <- unclustered
clusters_merged.expanded$expanded <- clusters_merged.filtered[2:
                                            length(clusters_merged.filtered)]

for(t in thres){
  clusters_merged.expanded <- expand_clusters(tasic_sp,
                            id_table = id_table,
                            cluster_list = clusters_merged.expanded$expanded,
                            unclustered = clusters_merged.expanded$unclustered,
                            force_expand = FALSE,
                            expand_threshold = t,
                            method = "percentile",
                            isoform_col = "transcript")
}
```

```r
# remaining unclustered
length(clusters_merged.expanded$unclustered)
#> [1] 144
```

For the remaining isoforms, we set `force_expand = TRUE` to assign them to the most similar cluster (i.e. the one exhibiting the highest percentile correlation with each isoform's expression).

```r
clusters_merged.expanded <- expand_clusters(tasic_sp, id_table = id_table,
                              cluster_list = clusters_merged.expanded$expanded,
                              unclustered = clusters_merged.expanded$unclustered,
                              force_expand = TRUE,
                              method = "percentile",
                              isoform_col = "transcript")
```

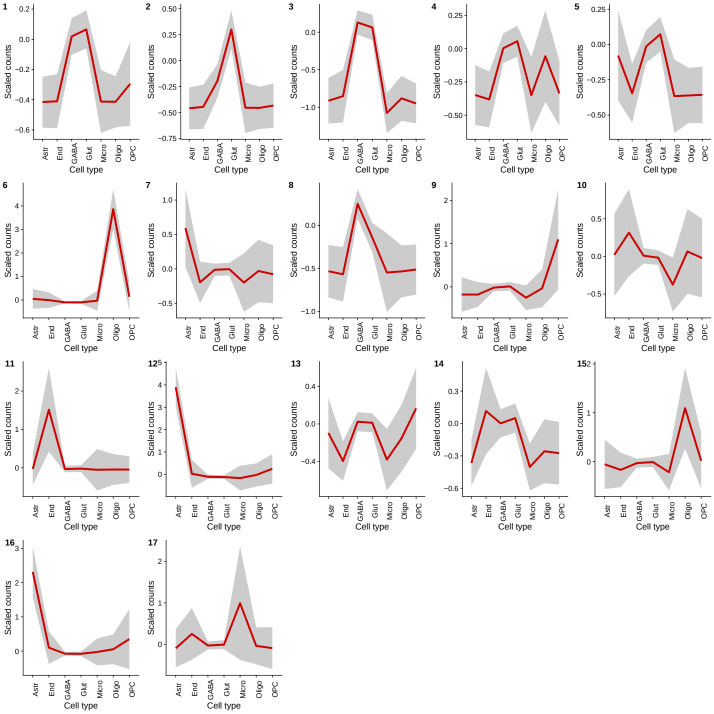
8.4 Final merge of redundant profiles

Our parameter choice for the initial cluster merge was rather lenient, meaning that it was adjusted to sacrifice some potentially correct merge decisions in order to avoid others that may result in highly dissimilar clusters being merged. For this reason, there may still be some cluster profiles that remain highly similar.

To remove these redundancies, we run `merge_clusters()` again with `dynamic = FALSE` and similar parameters:

```
# merge
merge_output_final <- merge_clusters(tasic_sp, id_table = id_table,
                                     cluster_list = clusters_merged.expanded,
                                     method = "percentile",
                                     dynamic = FALSE,
                                     height_cutoff = 0.1,
                                     isoform_col = "transcript")

clusters_final <- merge_output_final[[2]]

# plot final patterns
patterns_final <- map(clusters_final,
                      ~calculate_cluster_profile(tasic_scaled,
                                                 isoform_ids = .,
                                                 id_table = id_table,
                                                 isoform_col = "transcript"))

pattern_plots_final <- map(patterns_final, plot_cluster_profile,
                           ct_labels = ctlabs)
```

```
plot_grid(plotlist = pattern_plots_final,
          labels = seq(1, length(pattern_plots_final)))
```



Remarkably, there are still two pairs of highly similar clusters that have not been merged by automatic re-clustering, i.e. clusters 6 and 15, and clusters 12 and 16. To solve this and avoid the detection of false-positive coDIU genes due to the presence of redundant profiles, we merged these clusters manually:

```
# list manual merges
manual_list <- list(c(6, 15),
                    c(12, 16))

# merge clusters
clusters_final.curated <- c(clusters_final[-unlist(manual_list)],
                            map(manual_list, ~unlist(clusters_final[.])))

# set cluster names
names(clusters_final.curated) <- as.character(seq(1,
                                              length(clusters_final.curated)))
```

9 Keep isoforms from genes with Differential Isoform Usage (DIU)

As a result of clustering, we can next evaluate whether genes that have clustered isoforms are positive for **Differential Isoform Usage** (DIU). DIU genes must have more than one clustered isoform, and at least two of these isoforms assigned to different clusters. Differential cluster assignment indicates different isoform usage in at least one cell type, and is therefore a straightforward way to call DIU when multiple cell groups are considered.

To detect DIU genes, isoforms must be removed from clusters in the following cases: **1)** if these isoforms belong to genes that have a single isoform assigned to clusters and **2)** if they belong to genes with two or more clustered isoforms that have no same-gene counterparts in any of the other clusters. Both filters can be applied running the `keep_DIU()` function in acorde:

```
# remove isoforms from non-DIU genes
clusters_diu <- keep_DIU(clusters_final.curated,
                         gene_tr_table = gene_tr_ID)
#> Total no. of clusters: 15
#> Total isoforms in clusters: 6794
#> Isoforms clustered after >1 isoform/gene filter: 6794
#> Isoforms clustered after differential cluster assignment filter: 5278
```

To obtain a list of all DIU genes that have isoforms in our clusters, we simply need to use the `gene_tr_ID` table to translate transcript to gene IDs:

```
# obtain gene ID-based clusters
clusters_diu.gene <- map(clusters_diu,
                          ~gene_tr_ID[match(.,
                                            gene_tr_ID$transcript),]$gene)

# obtain list of DIU genes
diu_genes <- unlist(clusters_diu.gene) %>% unique

# total number of DIU genes:
length(diu_genes)
#> [1] 2017
```

In summary, we currently have clustered **5278 isoforms** into **15 expression patterns** across the 7 cell types in the Tasic dataset, and these isoforms belong to **2017 DIU genes**.

10 Detection of genes with co-Differential Isoform Usage (coDIU)

We define co-Differential Isoform Usage (coDIU) as an isoform expression pattern in which a group of genes shows co-expression of their isoforms, but no co-expression is detected when considering only gene-level expression. A coDIU situation between a pair of genes, *gene a* and *gene b*, is represented below:

First, we recommend adjusting some global parameters to allow heavy computation to take place (note that the exact value may depend on your system requirements):

```
options(future.globals.maxSize = 768 * 1024^2)
```

Then, we will use the `find_codiu_genes()` function in acorde to generate a list of potentially coDIU gene pairs, that is, genes that have at least two of their isoforms assigned to the same clusters, therefore showing isoform-level co-expression across cell types. We often refer to these as "shared genes", given that they share isoforms across two or more clusters:

```
# find shared gene pairs
shared_pairs <- find_codiu_genes(clusters_diu, gene_tr_table = gene_tr_ID)

# show dimensions of results
dim(shared_pairs)
```

However, clustering can allow expression pattern variability among cluster members, and sometimes isoforms in a cluster might not exactly reflect their expression pattern. Especially when coDIU is detected between two clusters reflecting a closely-related pattern (for instance, similar expression except for one cell type), there may be some false-positives coDIU genes among those detected by `find_codiu_genes()`.

To control for this, acorde provides a statistical test for each potentially coDIU gene pair, i.e. gene 1 and gene 2, for which isoforms were detected in two clusters, i.e. cluster 1 and cluster 2. We here test two different conditions for coDIU: **a)** that the average profile across cell types of the two isoforms in cluster 1 is significantly different to the average profile of the two isoforms in cluster 2; and **b)** that the average profile of the two isoforms of gene 1 is *not* different to the average profile of the two isoforms of gene 2.

For each pair of genes, the test will return two p-values, each corresponding to one of the above-described questions:

- *cluster:cell_type*: should be significant if **condition a** is fulfilled.
- *gene:cell_type*: should **NOT** be significant if **condition b** is satisifed.

The test is implemented in the `test_codiu_genes()` function, and can be run as follows (set `t` and `parallel = TRUE` -default- for parallel computation):

```
# test shared gene pairs
codiu_test <- test_codiu_genes(tasic_sp,
                               isoform_col = "transcript_id",
                               cluster_list = clusters_DIU,
                               shared_genes = shared_pairs,
                               gene_tr_table = gene_tr_ID,
                               id_table = id_table,
                               t = 7)

# obtain pvalues
pvalue.df <- map(codiu_test, "pvalues") %>% bind_rows

# adjust p-values
pvalue.df$`cluster:cell_type` <- p.adjust(pvalue.df$`cluster:cell_type`,
                                          method = "BH")
pvalue.df$`gene:cell_type` <- p.adjust(pvalue.df$`gene:cell_type`,
                                       method = "BH")
```

Now, we can filter the `shared_pairs` matrix to only keep genes that satisfy the two conditions for coDIU when tested with at least one other gene:

```
# filter shared gene list to only keep significant interactions
sig_pairs <- shared_pairs[,pvalue.df$`cluster:cell_type` < 0.05 &
                              pvalue.df$`gene:cell_type` > 0.05]
sig_all <- sig_pairs %>% as.character %>% unique
```

Finally, using the gene IDs in `sig_all`, we can now filter the clusters to only include isoforms from significantly coDIU genes, and generate a list of clusters with coDIU gene IDs:

```
# filter clusters to only include isoforms from coDIU genes
keep_coDIU <- clusters_DIU.gene %>% map(~(. %in% sig_all))
clusters_coDIU <- map2(clusters_DIU, keep_coDIU, ~(.x[.y]))

# convert coDIU clusters to gene IDs
clusters_coDIU.gene <- map(clusters_coDIU,
                           ~gene_tr_ID[match(., gene_tr_ID$transcript),]$gene)
```

11 Using acorde results for functional analysis

Providing a full description of functional analysis in the acorde manuscript is beyond the scope of this vignette. However, we will provide users with some tips, should they want to do a similar analysis of their single-cell expression clusters using tappAS.

11.1 Obtaining a functionally-annotated transcriptome with IsoAnnotLite

Prior to functional analysis, we transferred functional annotations to our long read-defined isoforms using IsoAnnotLite. Briefly, IsoAnnotLite takes SQANTI3 output files and a previously-annotated tappAS GFF3 file [4] (which can be found here) as inputs to generate a new, tappAS-compatible GFF3 file. To achieve this, transcripts are positionally matched to those in the pre-existing annotation and functional features transferred if they are situated in overlapping genomic positions.

For this study, given that we used a RefSeq transcriptome for long-read isoform definition, we used tappAS' *Mus musculus* RefSeq functional annotation (GRCm38, RefSeq release 78). Details for this process are available in Supplementary Note 1 in our manuscript (Arzalluz-Luque et al.).

11.2 Generating tappAS project inputs from single-cell data

Even though tappAS does not currently support single-cell data, our study took advantage of the qualitative analysis modules in the application, i.e. **Functional Diversity Analysis** and **Functional Enrichment Analysis** [4], to obtain insights on the functional features and functionalities that changed as a result of isoform co-expression and coDIU mechanisms.

To load single-cell RNA-Seq data into tappAS, we pretended to have a Time-Course Single Series design using cell IDs as sample IDs and cell types as time points. Please note that this strategy **is not valid for quantitative analyses**, since tappAS does not implement single-cell dedicated analysis methods.

Using the `metadata` table, a **tappAS design file** can be generated as follows:

```
# create design table from metadata
design <- metadata %>%
  select(run, cell_type) %>%
  dplyr::rename(sample = "run", time = "cell_type") %>%
  mutate(time = factor(time) %>% as.numeric(),
         group = rep("case", nrow(metadata))) %>%
  arrange(time)

# subset design to alleviate computational burden
design_sub <- design %>%
  group_by(time) %>%
  slice_sample(n = 10)

# export
write_tsv(design_sub, "tappas_design.tsv")
```

To generate an **expression matrix** for tappAS, run:

```
# create matrix with samples in design file
matrix_sp <- tasic_sp %>%
  select(transcript, all_of(design_sub$sample)) %>%
  column_to_rownames("transcript")

write.table(matrix_sp, sep = "\t", "tappas_matrix_sp.tsv")
```

Depending on the target isoform set, users may want to export the DE expression (to use the full dataset) or the muli-isoform count matrices (to only include isoforms input for clustering). This will mostly affect Functional Enrichment Analyses and the definition of background gene lists. For instance, to run a functional enrichment of DIU genes vs genes with at least one DE isoform, users must create a project including transcripts from all genes in that background list, i.e. using the `tasic_de` expression matrix.

Creating specific test and background **gene lists** is particularly useful for Functional Enrichment Analysis. Following the example in our study, we generated a specific list with all coDIU genes that included isoforms in oligodendrocyte, neuron and neuron-oligo clusters(i.e. clusters 1, 4 and 14). This list was used as test list for Functional Enrichment of GO terms, and can be generated using the `fromList()`function (modified from the UpSetR package), which generates a binary occurrence table indicating the clusters in which each coDIU gene has isoforms:

```
# create occurrence table
gene_occurrence <- acorde::fromList(clusters_coDIU.gene) %>%
  rownames_to_column("gene")

# select clusters
clust_select <- c(1, 4, 14)
```

```
# select shared gene group
clust_pair1 <- dplyr::select(gene_occurrence, "1", "4")
gene_group1 <- gene_occurrence$gene[rowSums(clust_pair1) == 2] %>%
  tibble(gene = .)

clust_pair2 <- dplyr::select(gene_occurrence, "4", "14")
gene_group2 <- gene_occurrence$gene[rowSums(clust_pair2) == 2] %>%
  tibble(gene = .)

gene_group <- bind_rows(gene_group1, gene_group2) %>% unique

# write file
write_tsv(gene_group,
          "clust_1_4_14_genes.tsv",
          col_names = FALSE)
```

In addition, a **transcript inclusion list** can be provided upon tappAS project creation to filter both the expression matrix and annotation files and create a project including only a transcript set of interest. This may be interesting if a particular group of genes is to be analyzed further. Following the same example, this would be the list used to create a project including only isoforms from clusters 1, 4 and 14; in order to perform the Functional Diversity Analysis in our neuron-oligo cluster results section:

```
# select transcripts from shared genes in clusters
tr_clust.idx <- map(clust_select,
                    ~which(clusters_coDIU.gene[[.]] %in% gene_group$gene))
tr_clust <- map2(clust_select, tr_clust.idx,
                 ~clusters_coDIU[[.x]][.y])

tr_clust_shared <- tibble(transcript = unlist(tr_clust))

# write file
write_tsv(tr_clust_shared,
          "clust_1_4_14_transcripts.tsv",
          col_names = FALSE)
```

In conclusion, the final output of the acorde pipeline consists in:

- **Isoform clusters** representing unique expression patterns across cell types.
- A list of identifiers of both **DIU** and **coDIU genes**, which can then be used for functional characterization.

More information about DIU and coDIU gene characterization and the type of functional insights that can be obtained from the acorde output can be found in Arzalluz-Luque et al..

12 References

If you use **acorde** in your research, please cite:

- [1] Arzalluz-Luque, A., Salguero, P., Tarazona, S. and Conesa, A. *acorde* unravels functionally interpretable networks of isoform co-usage from single cell data. *Nature Communications* 13, 1828 (2022). https://doi.org/10.1038/s41467-022-29497-w

Long-read data was obtained from the ENCODE consortium, and made publicly available by Wyman et al. in their bioRxiv preprint:

- [2] Dana Wyman, Gabriela Balderrama-Gutierrez, Fairlie Reese, Shan Jiang, Sorena Rahmanian, Weihua Zeng, Brian Williams, Diane Trout, Whitney England, Sophie Chu, Robert C. Spitale, Andrea Tenner, Barbara Wold, Ali Mortazavi: A technology-agnostic long-read analysis pipeline for transcriptome discovery and quantification. *bioRxiv* 672931 (2020); doi: https://doi.org/10.1101/672931

Short-read scRNA-seq data was obtained from Tasic et al. (2016):

- [3] Tasic, B. et al. Adult mouse cortical cell taxonomy revealed by single cell transcriptomics. *Nature Neuroscience*. 19, 335–346 (2016).
- [4] de la Fuente, L., Arzalluz-Luque, Á., Tardáguila, M. et al. tappAS: a comprehensive computational framework for the analysis of the functional impact of differential splicing. *Genome Biology* 21, 119 (2020). https://doi.org/10.1186/s13059-020-02028-w.

“Simulating co-expression in pre-simulated scRNA-seq data with acorde”

Ángeles Arzalluz-Luque [1]

[1] Institute for Integrative Systems Biology (I2SysBio), Valencia, Spain

Last updated: 15 junio, 2023

Package

acorde 0.1.0

Contents

1 Introduction

In this vignette, we will demonstrate how to introduce co-expression between features in an already-simulated scRNA-seq dataset. For this, we will first explain how we originally simulated single-cell data in the acorde manuscript using the SymSim R package, and then describe how to introduce and visualize co-expression patterns in the previously simulated scRNA-seq expression matrix.

2 Simulating scRNA-seq data with SymSim

If required, the SymSim R package can be installed from GitHub as follows (note usage of `build_vignettes = TRUE` to be able to access package documentation):

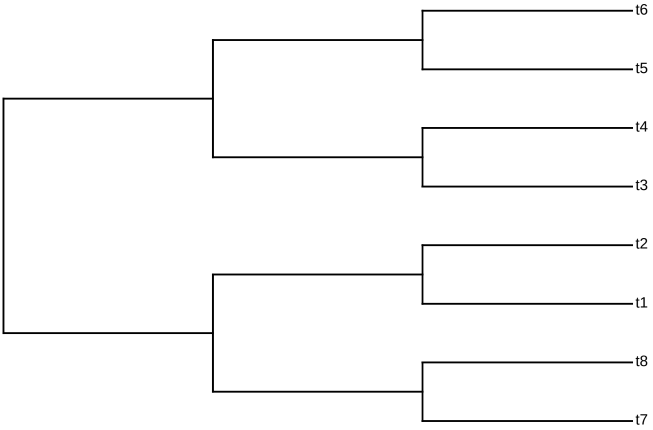
```
devtools::install_github("YosefLab/SymSim", build_vignettes = TRUE)
```

Load the SymSim package, acorde and other required dependencies:

```
suppressPackageStartupMessages({
  library(SymSim)
  library(tidyverse)
  library(acorde)
  library(scater)
})
```

First, we simulated true counts for 8 cell types, which requires the generation of a tree object detailing the relationships between the cell types (in this case, discrete cell types):

```
tree <- pbtree(n = 7, type = "discrete")
#> Warning:
#>   due to multiple speciation events in the final time interval
#>   realized n may not equal input n
plotTree(tree)
```



SymSim implements a two-step simulation, in which true counts are first simulated to then add technology-specific noise and user-defined biasess that are normally found on scRNA-seq data. For the first step, we run SymSim with the following parameters to generate a dataset containing 1000 total cells and 8000 genes:

```
# true counts
true_counts <- SimulateTrueCounts(ncells_total = 1000, ngenes = 8000,
                                  min_popsize = 100, i_minpop = 1,
                                  nevf = 10, n_de_evf = 9,
                                  evf_type = "discrete", phyla = tree,
                                  vary = "s", Sigma = 0.25,
                                  gene_effect_prob = 0.5, bimod = 0.4,
                                  prop_hge = 0.03, mean_hge = 5,
                                  randseed = 123)
```

Next, we run the simulation of observed counts as follows:

```
observed_counts <- True2ObservedCounts(true_counts = true_counts$counts,
                                       meta_cell = true_counts$cell_meta,
                                       protocol = "nonUMI",
                                       alpha_mean = 0.1, alpha_sd = 0.005,
                                       lenslope = 0,
                                       gene_len = rep(1000,
                                                      nrow(true_counts$counts)),
                                       depth_mean = 4e6, depth_sd = 1e4)
```

Here, we selected `"nonUMI"` as the simulated protocol to mimic the properties of the Smart-seq2 dataset used in our study. For a detailed description of the rest of the parameters, please see the SymSim documentation.

Finally, we created a `SingleCellExperiment` object for further processing, including a Principal Component Analysis (PCA) to better characterize the data structure and the variability between cells and cell types:

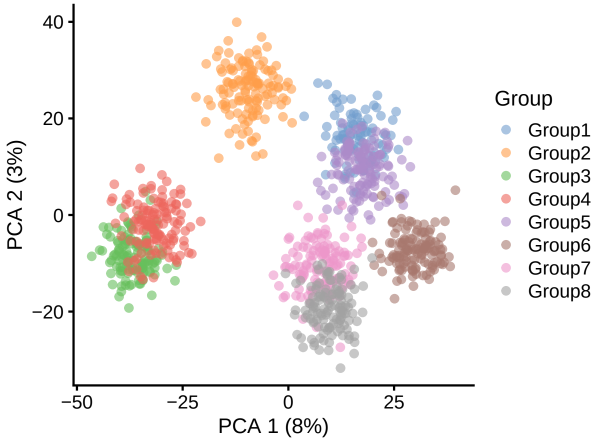
```
# feature and cell IDs as metadata
rownames(observed_counts$counts) <- paste0("Feature", seq(1, 8000))
colnames(observed_counts$counts) <- paste0("Cell", seq(1, 1000))
colData <- tibble(Cell = colnames(observed_counts$counts),
                  Group = paste0("Group", observed_counts$cell_meta$pop))

# create SCE
symsim_sce <- SingleCellExperiment(
  assays = list(counts = observed_counts$counts,
                logcounts = log2(observed_counts$counts+1)),
  colData = colData)
```

3 Introducing co-expression relationships in the simulated dataset

Our co-expression simulation method relies on **breaking the cell-type connectivity between features to rearrange them**. To perform this rearrangement and build new, synthetic features, we require the user to provide **cross-cell type expression patterns**. These patterns are qualitative, i.e. just indicate low or high expression in a given cell type.

If you started at this point, or you want to reproduce the results in our manuscript, load the already-simulated SymSim data object stored within this package:

```
# load data
data("symsim_sce")

# view object
symsim_sce
#> class: SingleCellExperiment
#> dim: 8000 1000
#> metadata(0):
#> assays(2): counts logcounts
#> rownames(8000): Feature1 Feature2 ... Feature7999 Feature8000
#> rowData names(0):
#> colnames(1000): Cell1 Cell2 ... Cell999 Cell1000
#> colData names(2): Cell Group
#> reducedDimNames(1): PCA
#> mainExpName: NULL
#> altExpNames(0):

# cell type IDs and composition
symsim_sce$Group %>% table()
#> .
#> Group1 Group2 Group3 Group4 Group5 Group6 Group7 Group8
#>    100    129    129    128    128    129    129    128

# plot PCA using scater function
plotPCA(symsim_sce, colour_by = "Group")
```



Next, it is required that users build an **expression pattern table**. This must have a structure where feature-level expression patterns are defined row-wise, meaning that cell-types are situated in the columns. Note that the number and order of cell types must be the same as in the simulated dataset. Each row in the `dataframe` should then include a `TRUE` value whenever high expression in a given cell type is desired, and `FALSE` to select low or no expression in that cell type. Here is an example of how to generate this structure:

```
# create cluster patterns
patterns <- tibble(one.a = c(TRUE, rep(FALSE, 7)),
                     one.b = one.a[sample(seq_along(one.a))],
                     one.c = one.b[sample(seq_along(one.b))],
                     one.d = one.c[sample(seq_along(one.c))],
                     one.e = one.d[sample(seq_along(one.d))],
                     two.a = c(rep(TRUE, 2), rep(FALSE, 6)),
                     two.b = two.a[sample(seq_along(two.a))],
                     two.c = two.b[sample(seq_along(two.b))],
                     two.d = two.c[sample(seq_along(two.c))],
                     two.e = two.d[sample(seq_along(two.d))],
                     three.a =  c(rep(TRUE, 3), rep(FALSE, 5)),
                     three.b = three.a[sample(seq_along(three.a))],
                     three.c = three.b[sample(seq_along(three.b))],
                     three.d = three.c[sample(seq_along(three.c))],
                     three.e = three.d[sample(seq_along(three.d))]) %>%
  t %>% as_tibble()
#> Warning: The `x` argument of `as_tibble.matrix()` must have unique column names if
#> `.name_repair` is omitted as of tibble 2.0.0.
#> i Using compatibility `.name_repair`.
#> This warning is displayed once every 8 hours.
#> Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
#> generated.

# show the results
patterns
```

```
#> # A tibble: 15 x 8
#>    V1    V2    V3    V4    V5    V6    V7    V8
#>    <lgl> <lgl> <lgl> <lgl> <lgl> <lgl> <lgl> <lgl>
#>  1 TRUE  FALSE FALSE FALSE FALSE FALSE FALSE FALSE
#>  2 FALSE FALSE FALSE FALSE FALSE TRUE  FALSE FALSE
#>  3 FALSE FALSE FALSE TRUE  FALSE FALSE FALSE FALSE
#>  4 FALSE FALSE FALSE FALSE FALSE FALSE TRUE  FALSE
#>  5 TRUE  FALSE FALSE FALSE FALSE FALSE FALSE FALSE
#>  6 TRUE  TRUE  FALSE FALSE FALSE FALSE FALSE FALSE
#>  7 FALSE FALSE FALSE FALSE FALSE TRUE  FALSE TRUE
#>  8 TRUE  FALSE FALSE TRUE  FALSE FALSE FALSE FALSE
#>  9 FALSE FALSE FALSE FALSE FALSE FALSE TRUE  TRUE
#> 10 TRUE  FALSE TRUE  FALSE FALSE FALSE FALSE FALSE
#> 11 TRUE  TRUE  TRUE  FALSE FALSE FALSE FALSE FALSE
#> 12 TRUE  FALSE FALSE FALSE TRUE  FALSE TRUE  FALSE
#> 13 FALSE FALSE TRUE  FALSE TRUE  TRUE  FALSE FALSE
#> 14 FALSE TRUE  FALSE FALSE TRUE  FALSE TRUE  FALSE
#> 15 FALSE FALSE FALSE TRUE  TRUE  TRUE  FALSE FALSE
```

In this example, we have 15 expression patterns, each corresponding to one gene cluster. The first pattern, for instance, where only the `Group1` cell type has a `TRUE` value, corresponds to genes where there is Group1-specific expression. Conversely, the last pattern will contain genes that are highly expressed in cell types `Group3`, `Group6`and `Group8`.

Now, we are ready to run the `simulate_coexpression()` function in acorde to generate the selected co-expression patterns in the already-simulated dataset:

```
coexpr_results <- simulate_coexpression(symsim_sce,
                                        feature_no = 1400,
                                        patterns,
                                        cluster_size = 200)

names(coexpr_results)
#> [1] "sim_matrix"   "sim_clusters"
```

As a result, we obtain a named list including the expression matrix and the feature IDs that go into each of the 15 clusters.

In this function, `feature_no` corresponds to the number of highly and lowly expressed features that are to be selected to produce clusters, i.e. a total of `feature_no * 2` features. In this example, we selected to generate 200-feature clusters. Given that our pattern matrix has 15 patterns, we expect to generate a new expression matrix with 3000 synthetically co-expressed features:

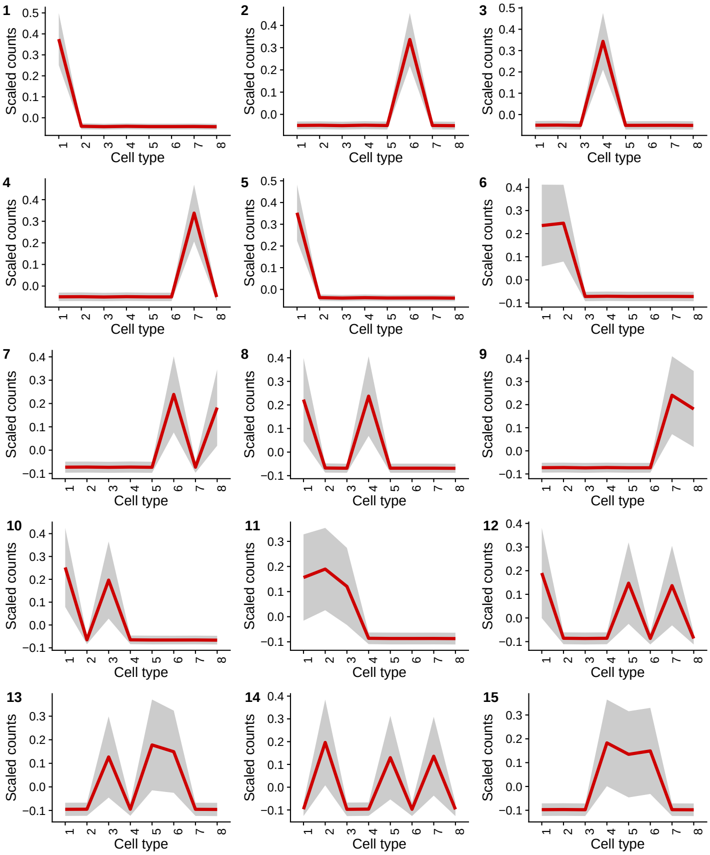
```
# show expression matrix
coexpr_results$sim_matrix
#> # A tibble: 3,000 x 1,001
#>    feature  Cell1 Cell2 Cell3 Cell4 Cell5 Cell6 Cell7 Cell8 Cell9 Cell10 Cell11
#>    <chr>    <int> <int> <int> <int> <int> <int> <int> <int> <int>  <int>  <int>
#>  1 Feature1   417   189   797   554   209   523   390   283   353    487    366
#>  2 Feature2  2304  3230  2652  3080  2467  1398  2552  2248  2635   2167   2225
#>  3 Feature3  1289   603  2506  2871  1935  1441  1539  2681  2380   1112   2190
#>  4 Feature4   764   306   485   349   574   439   827   125   811    455    748
#>  5 Feature5   6750  5305  6771  5569  7196  6880  6547  5032  7813   6276   6166
#>  6 Feature6    760   998  1457   777   856   438   789   510  1506    666    568
#>  7 Feature7   6694  7896  9459 10270  8800  7272  9732 10927 12295   5259  11009
#>  8 Feature8    277   666   830   267   228   464   214   214   278    433    219
#>  9 Feature9    540   741  1555  1717  1126   635   807  1768   480    667    993
#> 10 Feature10   494   185   401  1015   334   603   520   484   810    447    690
#> # i 2,990 more rows
#> # i 989 more variables: Cell12 <int>, Cell13 <int>, Cell14 <int>, Cell15 <int>,
#> #   Cell16 <int>, Cell17 <int>, Cell18 <int>, Cell19 <int>, Cell20 <int>,
#> #   Cell21 <int>, Cell22 <int>, Cell23 <int>, Cell24 <int>, Cell25 <int>,
#> #   Cell26 <int>, Cell27 <int>, Cell28 <int>, Cell29 <int>, Cell30 <int>,
#> #   Cell31 <int>, Cell32 <int>, Cell33 <int>, Cell34 <int>, Cell35 <int>,
#> #   Cell36 <int>, Cell37 <int>, Cell38 <int>, Cell39 <int>, Cell40 <int>, ...

# show summary of clusters
map_int(coexpr_results$sim_clusters, length)
#>   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15
#> 200 200 200 200 200 200 200 200 200 200 200 200 200 200 200
```

4 Visualizing the simulated co-expression patterns

Now, we can use some of the `acorde` visualization functions to make sure that our data contains the specified patterns:

```
# scale the matrix to enhance visualization
coexpr.scaled <- scale_isoforms(coexpr_results$sim_matrix,
                                isoform_col = "feature")

# create cell-to-cell-type ID table
ct <- colData(symsim_sce) %>%
  as_tibble %>%
  rename(cell = "Cell", cell_type = "Group")

# compute average-by-cell type cluster patterns
cluster_patterns <- map(coexpr_results$sim_clusters,
                    ~calculate_cluster_profile(coexpr.scaled,
                                               isoform_ids = .,
                                               id_table = ct,
                                               isoform_col = "feature"))


# plot patterns
library(cowplot)
#>
#> Attaching package: 'cowplot'
#> The following object is masked from 'package:lubridate':
#>
#>     stamp
#> The following object is masked from 'package:reshape':
#>
#>     stamp
```

```
theme_set(theme_cowplot())

pattern_plots <- map(cluster_patterns,
                     plot_cluster_profile,
                     ct_labels = seq(1, 8))

plot_grid(plotlist = pattern_plots,
          labels = seq(1, 15),
          ncol = 3)
```



Appendix II

CoDIU functional analysis results

II.1 Introduction

This appendix includes, whenever relevant, a more detailed account of the functional analysis results discussed in chapter 5, section 5.3.4. These are performed for the biological characterization of co-Differential Isoform Usage (coDIU) and the functional properties of the isoforms that are coordinately expressed by coDIU genes.

Note that, although omitted here for the sake of brevity, the databases or predictor tools from which the functional features within the different functional categories were retrieved upon the annotation process can be found in table 4.2.

II.2 Partially-overlapping samples z test for CoDIU vs DIU genes

This analysis reports functional attributes that are significantly overrepresented in coDIU genes when compared to DIU genes, comparing their proportions. These results were obtained using a partially-overlapping samples z test, as described in chapter 5, sections 5.2.7 and 5.3.4. To do this, the number of coDIU and DIU genes including each feature were used to compute proportions with respect to the total number of DIU genes and genes with at least one DE isoform, respectively.

Table II.1: Partially overlapping samples z-test results. Feature IDs are unique identifiers of the different functional features included within each functional category in the annotation.

Feature ID	Adjusted p-value	Functional category	Description	CoDIU genes w/ feature	DIU genes w/ feature
GO:0000122	0.01	GeneOntology	Negative regulation of transcription from RNA polymerase II promoter	60	65
GO:0000978	0.048	GeneOntology	RNA polymerase II promoter proximal region sequence-specific DNA binding	22	24
GO:0001067	0.045	GeneOntology	Regulatory region nucleic acid binding	43	48
GO:0001669	0.04	GeneOntology	Acrosomal vesicle	10	10
GO:0002682	0.047	GeneOntology	Regulation of immune system process	17	19
GO:0003690	0.048	GeneOntology	Double-stranded DNA binding	46	51
GO:0003723	0.015	GeneOntology	RNA binding	66	69
GO:0003735	0.013	GeneOntology	Structural constituent of ribosome	32	34
GO:0004672	0.037	GeneOntology	Protein kinase activity	47	49
GO:0004888	0.013	GeneOntology	Transmembrane signaling receptor activity	21	23
GO:0005198	0.004	GeneOntology	Structural molecule activity	49	52
GO:0005215	0.016	GeneOntology	Transporter activity	50	58
GO:0005615	2.6e-04	GeneOntology	Extracellular space	69	74
GO:0005730	0.037	GeneOntology	Nucleolus	63	69
GO:0005783	0.037	GeneOntology	Endoplasmic reticulum	82	90
GO:0005794	0.045	GeneOntology	Golgi apparatus	65	74
GO:0006357	0.004	GeneOntology	Regulation of transcription by RNA polymerase II	69	74
GO:0006412	0.018	GeneOntology	Translation	60	64
GO:0006468	0.037	GeneOntology	Protein phosphorylation	50	52
GO:0006629	0.036	GeneOntology	Lipid metabolic process	54	60
GO:0007017	0.038	GeneOntology	Microtubule-based process	34	37
GO:0007154	0.004	GeneOntology	Cell communication	75	88
GO:0007165	0.004	GeneOntology	Signal transduction	65	75
GO:0007186	0.037	GeneOntology	G protein-coupled receptor signaling pathway	13	15
GO:0009057	0.037	GeneOntology	Macromolecule catabolic process	24	24

Feature ID	Adjusted p-value	Functional category	Description	CoDIU genes w/ feature	DIU genes w/ feature
GO:0009653	0.027	GeneOntology	Anatomical structure morphogenesis	51	60
GO:0010604	0.037	GeneOntology	Positive regulation of macromolecule metabolic process	35	39
GO:0015075	0.049	GeneOntology	Ion transmembrane transporter activity	43	51
GO:0016021	0.016	GeneOntology	Integral component of membrane	76	91
GO:0016773	0.046	GeneOntology	Phosphotransferase activity, alcohol group as acceptor	54	57
GO:0019899	0.037	GeneOntology	Enzyme binding	42	47
GO:0022857	0.037	GeneOntology	Transmembrane transporter activity	45	53
GO:0023052	0.004	GeneOntology	Signaling	75	87
GO:0031324	0.004	GeneOntology	Negative regulation of cellular metabolic process	88	99
GO:0031325	0.048	GeneOntology	Positive regulation of cellular metabolic process	36	41
GO:0031327	0.004	GeneOntology	Negative regulation of cellular biosynthetic process	69	75
GO:0032268	0.015	GeneOntology	Regulation of cellular protein metabolic process	60	69
GO:0032270	0.037	GeneOntology	Positive regulation of cellular protein metabolic process	20	22
GO:0034220	0.037	GeneOntology	Ion transmembrane transport	49	58
GO:0043043	0.045	GeneOntology	Peptide biosynthetic process	62	68
GO:0043085	0.04	GeneOntology	Positive regulation of catalytic activity	37	43
GO:0043603	0.022	GeneOntology	Cellular amide metabolic process	77	84
GO:0043604	0.028	GeneOntology	Amide biosynthetic process	70	76
GO:0044255	0.037	GeneOntology	Cellular lipid metabolic process	44	49
GO:0044265	0.037	GeneOntology	Cellular macromolecule catabolic process	21	21
GO:0044283	0.037	GeneOntology	Small molecule biosynthetic process	17	19

Feature ID	Adjusted p-value	Functional category	Description	CoDIU genes w/ feature	DIU genes w/ feature
GO:0045892	0.004	GeneOntology	Negative regulation of transcription, DNA-templated	65	71
GO:0046394	0.032	GeneOntology	Carboxylic acid biosynthetic process	7	7
GO:0048518	0.013	GeneOntology	Positive regulation of biological process	55	63
GO:0048522	0.016	GeneOntology	Positive regulation of cellular process	54	62
GO:0048583	0.037	GeneOntology	Regulation of response to stimulus	32	39
GO:0050790	0.048	GeneOntology	Regulation of catalytic activity	60	74
GO:0051172	0.004	GeneOntology	Negative regulation of nitrogen compound metabolic process	80	91
GO:0051173	0.047	GeneOntology	Positive regulation of nitrogen compound metabolic process	32	36
GO:0051253	0.004	GeneOntology	Negative regulation of RNA metabolic process	67	73
GO:0055085	0.015	GeneOntology	Transmembrane transport	52	61
GO:0071944	0.036	GeneOntology	Cell periphery	64	77
GO:0098656	0.048	GeneOntology	Anion transmembrane transport	17	18
GO:2000113	0.004	GeneOntology	Negative regulation of cellular macromolecule biosynthetic process	68	74
miR-103-3p	0.04	miRNA binding	3UTR	60	64
miR-106a-5p	0.04	miRNA binding	3UTR	44	46
miR-129-2-3p	0.036	miRNA binding	3UTR	27	29
miR-132-3p	0.036	miRNA binding	3UTR	27	29
miR-136-5p	0.047	miRNA binding	3UTR	30	31
miR-137-3p	0.018	miRNA binding	3UTR	36	39

Feature ID	Adjusted p-value	Functional category	Description	CoDIU genes w/ feature	DIU genes w/ feature
miR-181a-5p	0.043	miRNA binding	3UTR	61	70
miR-1897-5p	0.036	miRNA binding	3UTR	30	32
miR-1907	0.035	miRNA binding	3UTR	75	83
miR-202-5p	0.036	miRNA binding	3UTR	25	26
miR-205-5p	0.018	miRNA binding	3UTR	30	31
miR-218-1-3p	0.036	miRNA binding	3UTR	53	57
miR-223-3p	0.031	miRNA binding	3UTR	33	35
miR-24-3p	0.039	miRNA binding	3UTR	55	62
miR-290a-3p	0.021	miRNA binding	3UTR	23	24
miR-295-5p	0.036	miRNA binding	3UTR	40	43
miR-29b-1-5p	0.031	miRNA binding	3UTR	30	30
miR-30a-5p	0.028	miRNA binding	3UTR	92	100
miR-322-3p	0.036	miRNA binding	3UTR	76	83
miR-322-5p	0.018	miRNA binding	3UTR	111	119
miR-335-3p	0.028	miRNA binding	3UTR	80	86
miR-337-3p	0.04	miRNA binding	3UTR	30	33
miR-338-5p	0.036	miRNA binding	3UTR	49	55
miR-350-3p	0.04	miRNA binding	3UTR	81	89
miR-363-3p	0.032	miRNA binding	3UTR	25	25

Feature ID	Adjusted p-value	Functional category	Description	CoDIU genes w/ feature	DIU genes w/ feature
miR-410-3p	0.036	miRNA binding	3UTR	41	44
miR-450a-2-3p	0.035	miRNA binding	3UTR	18	18
miR-450b-3p	0.043	miRNA binding	3UTR	24	27
miR-455-3p	0.036	miRNA binding	3UTR	20	21
miR-465a-5p	0.019	miRNA binding	3UTR	49	51
miR-466d-5p	0.018	miRNA binding	3UTR	123	136
miR-466g	0.048	miRNA binding	3UTR	40	43
miR-485-5p	0.018	miRNA binding	3UTR	71	77
miR-495-3p	0.036	miRNA binding	3UTR	160	182
miR-501-5p	0.028	miRNA binding	3UTR	42	43
miR-532-3p	0.018	miRNA binding	3UTR	49	50
miR-574-5p	0.04	miRNA binding	3UTR	47	50
miR-665-3p	0.04	miRNA binding	Non-coding region	48	49
miR-666-3p	0.04	miRNA binding	3UTR	64	69
miR-674-5p	0.036	miRNA binding	3UTR	49	50
miR-693-3p	0.04	miRNA binding	3UTR	47	51
miR-761	0.019	miRNA binding	3UTR	79	83
miR-762	0.04	miRNA binding	3UTR	70	81
miR-764-5p	0.04	miRNA binding	3UTR	24	25
miR-770-5p	0.036	miRNA binding	3UTR	30	31

Feature ID	Adjusted p-value	Functional category	Description	CoDIU genes w/ feature	DIU genes w/ feature
miR-7a-5p	0.031	miRNA binding	3UTR	41	42
miR-873a-5p	0.043	miRNA binding	3UTR	38	39
miR-876-5p	0.03	miRNA binding	3UTR	36	38
miR-877-5p	0.036	miRNA binding	3UTR	24	25
miR-881-5p	0.021	miRNA binding	3UTR	78	85
miR-92a-2-5p	0.018	miRNA binding	3UTR	43	48
miR-96-5p	0.04	miRNA binding	3UTR	23	24
NLS	0.041	MOTIF	Nuclear Localization Signal	84	91
LINE/L1	0.025	Repeat	Lx8	49	54
Low complexity	0.002	Repeat	GA-rich	207	236
LTR/ERVL -MaLR	0.005	Repeat	MTC	49	53
Simple repeat	3.46e-12	Repeat	(GGCT)n	724	827
SINE/Alu	0.002	Repeat	PB1D10	197	218
SINE/ID	0.003	Repeat	ID2	50	51
SINE/MIR	0.009	Repeat	MIRc	37	39
Signal peptide	5.12e-06	SIGNAL	Signal peptide cleavage site	123	147
TMhelix	0.004	TRANSMEM	Region of a membrane-bound protein predicted to be embedded in the membrane	20	22
Acetylation	2.267e-08	PTM	Acetyllysine	683	778
Active site	1.4e-04	ACTIVE SITE	Proton Acceptor	171	199
Binding	1.4e-04	BINDING	ATP	172	201
Carbohydrate	2.28e-07	PTM	N-Linked	184	221
Coiled	0.002	COILED	Regions of coiled coil within the protein	157	169

Feature ID	Adjusted p-value	Functional category	Description	CoDIU genes w/ feature	DIU genes w/ feature
Compositional bias	6.3e-40	COMPBIAS	Poly-Ser	190	210
Crosslink	0.018	BINDING	Glycyl Lysine Isopeptide (Lys-Gly) (Interchain With G-Cter In Sumo1)	117	136
Disulfide	4e-05	PTM	Disfulfide bond	91	111
Lipid	0.042	PTM	Gpi-Anchor Amidated Aspartate	62	74
Metal	0.01	BINDING	Magnesium	127	148
Methylation	5.4e-04	PTM	Methylarginine	146	165
Motif	0.001	MOTIF	Sh3-Binding	99	113
NP binding	0.001	BINDING	Nucleotide phosphate binding (ATP)	157	180
Phospho-Ser	5.57e-16	PTM	Phosphoserine	1097	1246
Phospho-Thr	1.34e-14	PTM	Phosphothreonine	746	834
Phospho-Tyr	3.37e-07	PTM	Phosphotyrosine	507	581
Compositional bias	0.045	COMPBIAS	Pro-Rich	42	47
PTM	6.57e-13	PTM	Ubiquitination	827	943
Region	4.00e-05	MOTIF	Interaction with Daxx	314	363
Transmembrane	4.5e-08	TRANSMEM	Helical	278	325
Zinc finger	4.6e-04	BINDING	Ubr-Type	89	98
U0006	0.016	3UTRmotif	Cytoplasmic polyadenylation element	29	32
U0017	4.3e-04	3UTRmotif	UNR binding site	169	187
U0023	1.8e-04	3UTRmotif	K-Box	274	311
U0024	3.00e-05	3UTRmotif	Brd-Box	163	183
U0025	6.00e-05	3UTRmotif	GY-Box	159	175
U0033	1.53e-11	uORF	Upstream Open Reading Frame	500	555
U0035	6.09e-14	3UTRmotif	Musashi binding element	1039	1177
U0043	4.24e-09	PAS	Polyadenylation signal	605	685

II.3 Functional Diversity Analysis of DE isoform, DIU and CoDIU genes

This section contains the results of the Functional Diversity Analysis (FDA) performed using tappAS for all genes with at least one Differentially Expressed (DE) isoform, with Differential Isoform Usage (DIU) and with co-Differential Isoform Usage (coDIU), as described in chapter 5, sections 5.2.7 and 5.3.4. FDA analyses were run using both the position and presence-based approaches implemented in tappAS. Analysed genes were grouped and counted by functional category based on their varying status for functional features from said categories. Since a gene is labeled as varying for a given functional category if at least one feature shows differential inclusion across its isoforms, each gene may be reported up to once per category. This table corresponds to the results shown in chapter 5, figure 5.12.

Table II.2: Functional Diversity Analysis (FDA) results for all coDIU genes, by functional category. The level at which the feature was annotated (Transcript/Protein) and the type of FDA analysis (Position/Presence) are indicated in each case. For each gene group (i.e. at least one DE isoform, DIU and coDIU) and functional category, the number of varying genes and the total number of genes with an annotated feature are shown. The proportion of varying genes was calculated by dividing varying by total number of genes.

Functional category	Feature annotation level	FDAtype	Gene set	Varying genes	Total genes	Proportion varying
3' UTR length	Transcript	Position	DE isoform	1847	2571	0.718
			DIU	1418	1925	0.736
			coDIU	1270	1700	0.747
3' UTR motif	Transcript	Position	DE isoform	1215	1865	0.651
			DIU	949	1420	0.668
			coDIU	854	1257	0.679
3' UTR motif	Transcript	Presence	DE isoform	1058	1865	0.567

Functional category	Feature annotation level	FDAtype	Gene set	Varying genes	Total genes	Proportion varying
			DIU	834	1420	0.587
			coDIU	747	1257	0.594
5' UTR length	Transcript	Position	DE isoform	1374	2571	0.534
			DIU	1039	1925	0.539
			coDIU	918	1700	0.54
5' UTR motif	Transcript	Position	DE isoform	43	65	0.661
			DIU	32	51	0.627
			coDIU	28	44	0.636
5' UTR motif	Transcript	Presence	DE isoform	43	65	0.661
			DIU	32	51	0.627
			coDIU	28	44	0.636
ACTIVE SITE	Protein	Position	DE isoform	58	270	0.214
			DIU	38	193	0.196
			coDIU	33	165	0.2
ACTIVE SITE	Protein	Presence	DE isoform	58	281	0.206
			DIU	36	199	0.180
			coDIU	32	171	0.187
BINDING	Protein	Position	DE isoform	252	726	0.347
			DIU	195	538	0.362
			coDIU	174	475	0.366
BINDING	Protein	Presence	DE isoform	215	771	0.278
			DIU	169	571	0.295
			coDIU	155	500	0.31
CDS	Transcript	Position	DE isoform	1583	2571	0.615
			DIU	1230	1925	0.638
			coDIU	1099	1700	0.646
COILED	Protein	Position	DE isoform	103	237	0.434
			DIU	85	193	0.440
			coDIU	80	181	0.441
COILED	Protein	Presence	DE isoform	105	257	0.408
			DIU	87	210	0.414
			coDIU	77	193	0.398
COMPBIAS	Protein	Position	DE isoform	152	342	0.444
			DIU	125	274	0.456
			coDIU	112	246	0.455
COMPBIAS	Protein	Presence	DE isoform	160	365	0.438
			DIU	128	289	0.442
			coDIU	116	259	0.447
DISORDER	Protein	Position	DE isoform	44	91	0.483

Functional category	Feature annotation level	FDAtype	Gene set	Varying genes	Total genes	Proportion varying
			DIU	35	74	0.472
			coDIU	32	67	0.477
DISORDER	Protein	Presence	DE isoform	26	97	0.268
			DIU	22	78	0.282
			coDIU	19	69	0.275
DOMAIN	Protein	Position	DE isoform	955	2098	0.455
			DIU	742	1587	0.467
			coDIU	657	1401	0.468
DOMAIN	Protein	Presence	DE isoform	932	2230	0.417
			DIU	726	1681	0.431
			coDIU	646	1481	0.436
INTRAMEM	Protein	Position	DE isoform	4	19	0.210
			DIU	4	14	0.285
			coDIU	4	14	0.285
INTRAMEM	Protein	Presence	DE isoform	5	20	0.25
			DIU	5	15	0.333
			coDIU	4	14	0.285
miRNA binding	Transcript	Position	DE isoform	1496	2207	0.677
			DIU	1145	1664	0.688
			coDIU	1020	1467	0.695
miRNA binding	Transcript	Presence	DE isoform	1496	2207	0.677
			DIU	1145	1664	0.688
			coDIU	1020	1467	0.695
MOTIF	Protein	Position	DE isoform	221	688	0.321
			DIU	169	524	0.322
			coDIU	156	465	0.335
MOTIF	Protein	Presence	DE isoform	215	726	0.296
			DIU	166	548	0.302
			coDIU	150	481	0.311
PAS	Transcript	Position	DE isoform	678	894	0.758
			DIU	513	685	0.748
			coDIU	457	605	0.755
PAS	Transcript	Presence	DE isoform	643	894	0.719
			DIU	488	685	0.712
			coDIU	436	605	0.721
polyA site	Transcript	Position	DE isoform	1495	1700	0.867
			DIU	2228	2571	0.874
			coDIU	1682	1925	0.879
PTM	Protein	Position	DE isoform	925	1936	0.477

Functional category	Feature annotation level	FDAtype	Gene set	Varying genes	Total genes	Proportion varying
			DIU	714	1458	0.489
			coDIU	633	1289	0.491
PTM	Protein	Presence	DE isoform	759	2073	0.366
			DIU	593	1553	0.381
			coDIU	526	1369	0.384
Repeat	Transcript	Position	DE isoform	1042	1504	0.692
			DIU	815	1142	0.713
			coDIU	725	1014	0.715
Repeat	Transcript	Presence	DE isoform	882	1504	0.586
			DIU	710	1142	0.622
			coDIU	637	1014	0.628
RBP	Transcript	Position	DE isoform	116	188	0.617
			DIU	88	144	0.611
			coDIU	79	123	0.642
RBP	Transcript	Presence	DE isoform	95	188	0.505
			DIU	74	144	0.514
			coDIU	69	123	0.561
SIGNAL	Protein	Position	DE isoform	78	235	0.332
			DIU	54	148	0.365
			coDIU	44	123	0.357
SIGNAL	Protein	Presence	DE isoform	93	244	0.381
			DIU	63	153	0.412
			coDIU	53	128	0.414
TRANSMEM	Protein	Position	DE isoform	159	482	0.330
			DIU	113	329	0.343
			coDIU	97	280	0.346
TRANSMEM	Protein	Presence	DE isoform	124	515	0.241
			DIU	83	347	0.239
			coDIU	75	298	0.251
uORF	Transcript	Position	DE isoform	577	739	0.781
			DIU	439	555	0.791
			coDIU	395	500	0.79
uORF	Transcript	Presence	DE isoform	449	739	0.607
			DIU	347	555	0.625
			coDIU	311	500	0.622
Complex	Protein	Presence	DE isoform	3	18	0.167
			DIU	3	12	0.25
			coDIU	2	10	0.2

II.4 Functional Diversity Analysis of oligodendrocyte-neuron clusters

This section contains the results of the Functional Diversity Analysis (FDA) performed using tappAS for 118 genes showing coDIU between showing coDIU between clusters 1, 4 and 14, i.e. representing a neuron-oligodendrocyte co-expression pattern, as described in chapter 5, sections 5.2.7 and 5.3.5. FDA analyses were run using both the position and presence-based approaches implemented in tappAS. Analysed genes were grouped and counted by functional category based on their varying status for functional features from said categories. Since a gene is labeled as varying for a given functional category if at least one feature shows differential inclusion across its isoforms, each gene may be reported up to once per category. This table corresponds to the results shown in Chapter 5, figure 5.13.b.

Table II.3: Functional Diversity Analysis (FDA) results for coDIU genes in the oligodendrocyte-neuron clusters (118 genes total). For each functional category, the level at which the feature was annotated (Transcript/Protein) and the type of FDA analysis (Position/Presence) are indicated. The proportion of varying genes was calculated with respect to the total number of genes containing at least one feature from a given functional category.

Functional category	**Feature annotation level**	**FDAtype**	**Varying genes**	**Proportion varying**
3' UTR length	Transcript	Position	82	0.695
3' UTR motif	Transcript	Position	56	0.475
		Presence	0.559	
5' UTR motif	Transcript	Position	3	0.025
		Presence	3	0.025

Functional category	**Feature annotation level**	**FDAtype**	**Varying genes**	**Proportion varying**
ACTIVE SITE	Protein	Position	1	0.008
		Presence	1	0.008
BINDING	Protein	Position	11	0.093
		Presence	9	0.076
CDS	Transcript	Position	66	0.559
COILED	Protein	Position	4	0.034
		Presence	5	0.042
COMPBIAS	Protein	Position	4	0.034
		Presence	7	0.059
DISORDER	Protein	Position	2	0.017
		Presence	4	0.034
DOMAIN	Protein	Position	34	0.288
		Presence	35	0.297
MOTIF	Protein	Position	7	0.059
		Presence	8	0.068
PAS	Transcript	Position	32	0.271
		Presence	31	0.263
PTM	Protein	Position	31	0.263
		Presence	29	0.246
RBP	Transcript	Position	1	0.008
SIGNAL	Protein	Position	1	0.008
		Presence	1	0.008
TRANSMEM	Protein	Position	2	0.017
		Presence	1	0.008
miRNA binding	Transcript	Position	71	0.602
		Presence	71	0.602
polyA Site	Transcript	Position	102	0.864

Functional category	**Feature annotation level**	**FDAtype**	**Varying genes**	**Proportion varying**
repeat	Transcript	Position	51	0.432
		Presence	48	0.407
uORF	Transcript	Position	31	0.263
		Presence	22	0.186