Universidad Nacional de Córdoba

Facultad de Ciencias Exactas, Físicas y Naturales

Tesis Doctoral

Minería de Datos Aplicada al Estudio de Modificaciones Post-Transcripcionales del ARN

Autor: Bioing. Gabriela Alejandra Merino

Director: Dr. Elmer Andrés Fernández

Mayo de 2018

Minería de Datos Aplicada al Estudio de Modificaciones Post-Transcripcionales del ARN

por

Bioing. Gabriela Alejandra Merino

Dr. Elmer Andrés Fernández
Director

Comisión Asesora

Dr. Elmer Andrés Fernández
FCEFyN-UNC / CIDIE-UCC

Dra. Cristina Noemí Gardenal
FCEFyN-UNC / IDEA-CONICET

Dra. Teresa María Reyna
FCEFyN-UNC

Esta Tesis fue enviada a la Facultad de Ciencias Exactas, Físicas y Naturales de la Universidad Nacional de Córdoba para cumplimentar los requerimientos de obtención del grado académico de Doctor en Ciencias de la Ingeniería.

Córdoba, Argentina
Mayo de 2018

UNIVERSIDAD NACIONAL DE CORDOBA
Facultad de Cs. Exactas, Fisicas y Naturales

ACTA DE EXAMENES

Libro: 00001 Acta: 04288 Hoja 01/01
LLAMADO: 1 03/05/2018
CATEDRA - MESA:

DI002 TESIS DOCTORADO EN CIENCIAS DE LA INGENIERIA

NUMERO	APELLIDO Y NOMBRE	DOCUMENTO	INGRESO	COND.	NOTA	FIRMA
33652509	MERINO, Gabriela Alejandra	DNI: 33652509	2014	T	Aprobado	

FLESIA, Ana Georgina - YANOVSKY, Marcelo - STEGMAYER, Georgina - MALDONADO, Ana - ROJAS, Nadina -

Observaciones:

Sergio Elaskar
DIRECTOR
Doctorado en Ciencias de la Ingeniería
F.C.E.F. y N. - U.N.C.

Córdoba, ____/____/_____.
Certifico que la/s firma/s que ha/n sido puesta/s en la presente Acta pertenece/n a: A. G. Flesia

Inscriptos	Ausentes	Examinados	Reprobados	Aprobados
1	0	1	0	1

25/04/2018 10:07:38 Libro/Acta: 0000104288 Hoja: 01/ 01

Dedicado a Diego, Cleta,
mis padres y a toda mi familia.

Agradecimientos

Luego de un largo camino culmina una etapa llena de aprendizaje académico y personal. El recorrido no ha sido fácil, pero no me arrepiento de haberlo transitado. Si bien a veces podemos sentir que caminamos solos, en realidad hay muchas personas acompañándonos y apoyándonos. Por lo tanto, esta tesis no me pertenece sólo a mí, también es de todos aquellos que estuvieron siempre a mí lado, a quienes hoy quiero agradecer.

Este trabajo no habría sido posible sin el apoyo y el estímulo de mi Director, el Doctor Elmer Fernández, quien sin conocerme confió en mí para emprender un gran desafío en un área que hasta el momento no había sido estudiada en su grupo. Hoy en día, ese desafío lo considero superado, por lo que no puedo decir más que gracias. Quiero además agradecer a todos los integrantes y/o colaboradores del Grupo de Minería de Datos en Biociencias con los que he tenido la oportunidad de trabajar. Especialmente, gracias Cristóbal por la ayuda constante y desinteresada que siempre nos brindaste a todos. Gracias también a Germán, Anibal y Juan Cruz por todos los momentos compartidos a lo largo de estos años. Me gustaría además mencionar a todos mis compañeros y docentes de la Maestría en Estadística Aplicada, cohorte 2014, y a mis compañeros de la cátedra Análisis Matemático II de la FCEFyN, especialmente a Claudia por apoyarme, enseñarme y guiarme en mi camino como docente.

Por supuesto que nada es posible sin esos amigos, los de ayer, los de hoy, los de siempre. A cada uno de ellos quiero agradecerles su apoyo y amistad incondicional. Pese a las distancias o las diferencias que podamos tener, ellos siempre estuvieron conmigo, en los buenos y en los malos momentos. Gracias a mis amigas de la niñez que aún conservo conmigo, a mis amigos de Sierra, mis amigos de la facu y mis amigos de Córdoba.

No puedo terminar sin agradecer a mi ejemplo, mi familia, en cuyo estímulo constante y amor he confiado a lo largo de mis años de estudio. Mis padres, Luis

y Ana, mis hermanas, Vanesa y Agostina quienes han sido pilares fundamentales y fuente de inspiración para seguir siempre hacia adelante. A mis casi hermanos, Laura, Andrea y Juancy gracias por ser incondicionales. A mi ahijado Alejo por todo el amor puro, sincero y desinteresado que siempre le da a su madrina. Gracias a mis abuelas, cuñados, sobrinos, primos y tíos que me han acompañado y apoyado en todo momento. Mención a parte para mis abuelos Yiyo, Fede y Arturo, mis ángeles, mi fuerza. Por útlimo, a mi pequeña gran familia, mis grandes amores, mis compañeros, mis amigos y cómplices, Diego y Cleta. Es a ellos a quienes dedico este trabajo.

Resumen

El splicing alternativo es uno de los principales mecanismos post-transcripcionales del ARN, responsable de la obtención de varias isoformas a partir de un único gen. Alteraciones en este proceso impactan en el nivel de expresión absoluta y relativa de dichas isoformas. La exploración de cambios en el splicing alternativo, splicing diferencial, se realiza mediante experimentos transcriptómicos. Si bien existen herramientas para el análisis de tales experimentos, no existe un consenso a la hora de optar por una u otra. Más aún, tales herramientas indagan distintos tipos de cambio en la expresión, por lo que la falta de integración de sus resultados conlleva, muchas veces, a pérdida de información biológica relevante. Esta pérdida se ve también acrecentada por la falta de un control de calidad a lo largo de todo el análisis.

Esta tesis presenta diferentes estrategias que conforman un flujo de análisis estructurado de los datos transcriptómicos, enriqueciendo los resultados y la información obtenida. Se desarrolló `TarSeqQC`, una herramienta de control de calidad que permite detectar sesgos globales y puntuales afectando a un experimento. Además, ésta posee funcionalidades gráficas que facilitan la exploración de los resultados del control de calidad, focalizando la atención en regiones genómicas específicas. Se comparó objetivamente distintos flujos de análisis de cambios en la expresión, obteniendo una guía práctica para asistir la adecuada selección de métodos de análisis. Ante la necesidad de contar con un método que cuantifique y detecte el splicing diferencial, se desarrolló `NBSplice`. Éste fue evaluado con datos sintéticos, superando en desempeño a las herramientas actuales. Las metodologías propuestas han sido aplicadas a diversos experimentos, demostrando su utilidad en el análisis transcriptómico.

Palabras claves: BIOINGENIERÍA - BIOINFORMÁTICA - SECUENCIACIÓN DE ALTO RENDIMIENTO - MODELOS ESTADÍSTICOS.

Abstract

Alternative splicing is one of the most important RNA post-transcriptional mechanisms, responsible for the obtention of several isoforms from a single gene. The exploration of this mechanism and its changes is done using transcriptomic experiments. Currently, there are several tools available for analyzing this kind of experiments, however, up to now do not exist a consensus to guide the choice of one over another. Moreover, those tools inquire different levels of expression changes without integration of their results, which could drive to a loss of relevant biological information. This lost is also increased by the missing of quality control steps along the data processing.

This thesis presents different strategies conforming a structured analysis pipeline for transcriptomic data, enriching the obtained results and information. Here, `TarSeqQC` is presented, a quality control tool allowing the detection of global and punctual biases affecting the experiment. It provides graphical functionalities facilitating the exploration and quality control of results focusing on specific genomic regions. In addition, a practical guide for the selection of the most appropriate workflow for alternative splicing analysis is proposed. This guide is based on a comparative study of several popular pipelines. Finally, a new method for quantifying and detecting differential splicing changes, called `NBSplice`, is proposed. This method was evaluated using synthetic data, overcoming the performance of current tools. The proposed approaches were applied to several real experiments demonstrating and validating their usefulness for transcriptomic analysis.

Keywords: BIOENGINEERING - BIOINFORMATICS - HIGH THROUGHPUT SEQUENCING - STATISTICAL MODELS.

Resumo

O splicing alternativo é um dos principais mecanismos pós-transcricionais do RNA, responsável pela obtenção de várias isoformas a partir de um único gene. Alterações neste processo afetam o nível de expressão absoluta e relativa de tais isoformas. A exploração de mudanças no splicing alternativo, splicing diferencial, se realiza mediante experimentos transcriptômicos. Se bem existem ferramentas para análise de tais experimentos, não existe um consenso na hora de optar por uma ou outra. Além disso, tais ferramentas questionam distintos tipos de variação na expressão, por este motivo a falta de integração de seus resultados leva, muitas vezes, a perdas de informação biológica relevante. Estas perdas se veem aumentadas também por falta de um controle de qualidade ao longo de toda análise. Esta tese apresenta diferentes estratégias que formam um fluxo de análise estruturado dos dados transcriptômicos, enriquecendo os resultados e a informação obtida. Se desenvolveu `TarSeqQC`, uma ferramenta de controle de qualidade que permite detectar vieses globais e pontuais afetando a um experimento. Ademais, esta ferramenta possui funcionalidades gráficas que facilitam a exploração dos resultados do controle de qualidade, focalizando a atenção em regiões genômicas específicas. Se comparou objetivamente distintos fluxos de análise de variações na expressão, obtendo um guia prático para ajudar na adequada seleção de métodos de análise. Frente à necessidade de contar com um método que quantifique e detecte o splicing diferencial, se desenvolveu `NBSplice`. Este último foi avaliado com dados sintéticos, superando em desempenho às ferramentas atuais. As metodologias propostas têm sido aplicadas a diversos experimentos, demostrando sua utilidade na análise transcriptômica.

Palavras-chave: BIOENGENHARIA - BIOINFORMÁTICA - SEQUENCIAMENTO DE ALTO DESEMPENHO - MODELOS ESTATÍSTICOS.

Abreviaturas

ACC : Accuracy o exactitud

ADN : Ácido DesoxirriboNucleico

ADNc : Ácido DesoxirriboNucleico complementario

ARN : Ácido RiboNucleico

ARNm : ARN mensajero

ARNnc : ARN no codificante

ARNnp : ARN nuclear pequeño

ARNr : ARN ribosomal

ASG : Alternative Spliced Genes, genes con cambios en el splicing alternativo

BAM : Binary Alignment Map o mapa de alineamiento binario

cfDNA : cell free DNA o ADN celular libre

ChIP-Seq : Chromatine ImmunoPrecipitation Sequencing o secuenciación de inmunoprecipitación de cromatina

CPM : Conteos Por Millón o Counts Per Million

DE : Differential Expression, expresión diferencial

DI : Differential Isoforms, isoformas diferenciales

DIE : Differential Isoform Expression, expresión diferencial de isoformas

DIEDS : Differential Isoform Expression and Differential Splicing, expresión diferencial de isoformas y splicing diferencial

DGE : Differential Gene Expression, expresión diferencial de genes

DM : Data Mining, minería de datos

DNA : DesoxiriboNucleic Acid, Ácido DesoxirriboNucleico

DNA-seq : DNA Sequencing, secuenciación de ADN

DS : Differential Splicing, splicing diferencial

FDR : False Discovery Rate, Tasa de descubrimientos falsos

FPKM : Fragments Per Kilobase per Million of mapped Reads, fragmentos por kilobase por millón de lecturas mapeadas

FP : False Positive o Falso Positivo

FN : False Negative o Falso Negativo

FPR : False Positives Rate, tasa de falsos positivos

GBS : Genotyping by Sequencing, genotipificación por secuenciación

GLM : Generalized Linear Model, modelo lineal generalizado

ID : IDentificador

InDel : Inserción o Deleción de una o más nucleótidos en un fragmento de ADN/ARN

KDD : Knowledge Discovery in Databases, descubrimiento de conocimiento en bases de datos

$\log_2$: logaritmo en base 2

MAPQ : MAPping Quality, calidad de mapeo

Met : Metastásico

miARN : micro ARN

N : Negative o Negativo

NB : Negative Binomial, binomial negativa

NGS : Next Generation Sequencing, secuenciación de próxima o segunda generación

Non-Met : No Metastásico

OSCC : Oral Squamous Cell Carcinoma, carcinoma de células orales escamosas

P : Positive o Positivo

pb : pares de bases

PCA : Principal Component Analysis, análisis de componentes principales

PCR : Polimerase Chain Reaction, reacción en cadena de la polimerasa

PPDE : Posterior Probability of being Differentially Expressed, probabilidad a posteriori de estar diferencialmente expresado

PPV : Positive Predicted Value o precisión

preARNm : ARN primario o precursor

q-RT-PCR : quantitative Real Time Polimerase Chain Reaction o reacción en cadena de la polimerasa cuantitativa en tiempo real

RLE : Relative Logarithmic Normalization, normalización logarítmica relativa

RNA : RiboNucleic Acid, ácido ribonucleico

RNA-seq : RNA Sequencing, secuenciación de ARN

RPKM : Reads Per Kilobase per Million of mapped Reads, lecturas por kilobase por millón de lecturas mapeadas

SA : Splicing Alternativo

SAM : Sequence Alignment/Map, mapa o alineamiento de secuencias

SNP : Single Nucleotide Polymorphism, polimorfismo de nucleótido único

SRA : Sequencing Read Archive, archivo de lecturas de secuenciación

TCGA : The Cancer Genome Atlas, atlas del genoma del cáncer

TMM : Trimmed Mean of M values, media recortada en M valores

TN : True Negative, verdadero negativo

TP : True Positive, verdadero positivo

TPR : True Positive Rate o sensibilidad

UQ : Upper Quartile, cuartil superior

Prefacio

El *splicing alternativo* (SA) es un mecanismo post-transcripcional característico de los organismos eucariotas, encargado de la generación de múltiples transcritos de ARNm, isoformas, a partir de un único gen. Más aún, el SA es el principal responsable de la complejidad funcional y diversidad de proteínas en estos organismos, obtenida a partir de un bajo número de genes. Por ejemplo, más del 90 % de genes humanos y aproximadamente un 60 % de los genes de la mosca de la fruta (*Drosophila melanogaster*) son afectados por este mecanismo en condiciones normales (Hooper, 2014; Liu et al., 2014). Asimismo, diversos estudios han revelado que *alteraciones* en el mecanismo de SA están vinculadas con numerosas patologías, lo cual ha motivado su estudio en profundidad. En particular, cambios en la expresión tanto absoluta como relativa de las isoformas de un gen, *splicing diferencial* (del inglés *differential splicing*, DS), han sido directamente relacionados con patologías humanas como el cáncer (Feng et al., 2013).

El advenimiento de las *tecnologías de secuenciación masiva*, más conocidas del inglés *next generation sequencing* (NGS), ha favorecido la exploración, estudio e incluso descubrimiento de diversos fenómenos biológicos, entre ellos el SA. La versatilidad y el bajo costo de las NGS permitieron ampliar considerablemente tanto la escala como la complejidad de los experimentos biológicos. Específicamente, la técnica RNA-seq se ha vuelto un estándar para el estudio simultáneo de la *expresión* de genes e isoformas, como así también para la identificación y estudio del SA. Esto ha permitido la exploración simultánea de todos los transcritos existentes en un determinado momento, lo que se ha denominado *transcriptómica*. Generalmente los *estudios transcriptómicos* tienen como fin detectar los *cambios en la expresión* producidos por las diferencias entre condiciones específicas bajo estudio (Liu et al., 2014). En este contexto, el análisis de los datos requiere de *herramientas informáticas/estadísticas* que permitan optimizar la extracción de información a partir de datos complejos y voluminosos generados en este tipo

de experimentos. A grandes rasgos, el análisis transcriptómico implica tres pasos bien diferenciados: el ordenamiento de las lecturas de secuenciación a lo largo del genoma del organismo bajo estudio, o *alineamiento*, la *cuantificación* de la expresión a partir de las lecturas ordenadas, y la comparación de la expresión en diferentes condiciones para determinar la ocurrencia de *expresión diferencial* (Oshlack et al., 2010).

Desde la aparición de las NGS, numerosas herramientas se han desarrollado para abordar el análisis de este tipo de datos (Oshlack et al., 2010; Wang et al., 2009). En este contexto, el control de calidad y la exploración de los resultados de cada una de las etapas de dicho análisis son fundamentales para asegurar la confiabilidad de las conclusiones abordadas. No obstante, la mayoría de los flujos de análisis propuestos no lo incorporan en todas las etapas del análisis, a lo que se suma el hecho de que las herramientas desarrolladas sólo se focalizan en un control de calidad de la secuenciación (Andrews, 2011; Oshlack et al., 2010). Adicionalmente, dado que el resultado de la cuantificación es un valor de expresión para cada unidad (gen, isoforma o exón), las estrategias de visualización más utilizadas se basan en la exploración de estos valores. Sin embargo, éstos sólo son una estimación puntual cuando, en realidad, se cuenta con un perfil de lecturas distribuidas a lo largo del gen (isoforma o exón) compuesto por las frecuencias observadas para cada nucleótido del fragmento bajo estudio (Ramírez et al., 2014). Explorar estos perfiles de expresión resulta fundamental a la hora de discernir el comportamiento de regiones solapadas, como son las isoformas. Además, esta exploración es una herramienta clave para el *control de calidad* de los datos, el cual muchas veces es ignorado, llevando a resultados incompletos e incluso erróneos.

Por otra parte, se ha encontrado que los algoritmos existentes para el análisis de expresión diferencial persiguen su objetivo en base a distintos *niveles de información* (genes, isoformas o SA), criterios e hipótesis (Alamancos et al., 2014; Feng et al., 2013). Más aún, generalmente se utiliza sólo una herramienta para dicho análisis (Merino et al., 2017a). Por ejemplo, algunos de los enfoques que intentan determinar el DS se basan en análisis a nivel de exones, en cambio otros lo hacen a partir de expresión de isoformas. Luego, si sólo se considera herramientas que analicen exones, los resultados que se obtengan ignorarán lo que esté sucediendo a nivel de las isoformas e incluso de los genes como un todo. En este contexto, si bien se han reportado algunos trabajos comparando el desempeño de las distintas

herramientas disponibles para el estudio del SA, éstos tienen su base en análisis descriptivos (Alamancos et al., 2014; Hooper, 2014). Todo esto deriva en *falta de consenso e integración* tanto entre los distintos enfoques como en la elección de la herramienta más adecuada, lo cual conduce en la *pérdida* de parte de la información contenida en los datos de secuenciación. A esto se le suma el hecho de que el desarrollo de las herramientas de análisis de expresión diferencial se ha focalizado en el análisis a nivel de genes o exones, mientras que los enfoques a nivel de las isoformas han sido poco desarrollados (Feng et al., 2013; Sánchez-Pla et al., 2012). Consecuentemente, las estimaciones sobre los cambios en el SA se realizan indirectamente, ignorando tanto la identidad de las isoformas, cuya expresión resulta alterada, como la magnitud del cambio de dicho fenómeno. Luego, se pierde información fundamental para comprender el impacto biológico de las modificaciones ocurridas en el SA.

Esta tesis tiene como objetivo el desarrollo de *estrategias de procesamiento* de datos de RNA-seq de manera de proveer un *marco estructurado* que permita *optimizar* la extracción de información a partir de este tipo de experimentos, asegurando la *calidad* de los resultados obtenidos. Se pretende integrar los enfoques de detección, análisis y visualización de modificaciones post-transcriptcionales, como el SA, mediante la implementación de técnicas estadístico-computacionales. En particular, en esta tesis se presenta `TarSeqQC`, una herramienta de control de calidad y visualización desarrollada para la exploración e interpretación de resultados. Adicionalmente, se propone un flujo estructurado de análisis de datos, basado en herramientas ampliamente utilizadas, que prioriza la calidad de los datos bajo análisis y minimiza la pérdida de información. La comparación objetiva de las distintas estrategias y enfoques de análisis de SA disponibles, permitió obtener una guía para la selección de herramientas en base al tipo de experimento que se desea analizar. Finalmente, se propone `NBSplice`, un nuevo algoritmo de análisis desarrollado para la detección de DS en base a cuantificaciones a nivel de isoforma, que permite identificar las isoformas afectadas y cuantificar el DS.

La organización del documento de tesis es como sigue:

Capítulo 1: introduce al lector al concepto de **minería de datos** en el contexto del análisis transcriptómico. Se describen las diferentes etapas involucradas en el análisis (entendimiento del problema y datos, modelado, evaluación y reporte).

Capítulo 2: brinda una visión global del **análisis transcriptómico**, las dife-

rentes metodologías y herramientas existentes, introduciendo al lector en los diferentes problemas asociados a este tipo de análisis.

Capítulo 3: presenta los aportes realizados en este trabajo de tesis, en el contexto del análisis transcriptómico, a la etapa de **entendimiento de los datos**. Los aportes están dirigidos al control de calidad y su utilidad se demuestra mediante el análisis de dos conjuntos de datos. El primero de ellos es un experimento dirigido a la determinación de las diferencias en término de expresión existentes entre individuos que poseen cáncer oral con y sin presencia de metástasis linfonodal. En el segundo caso se estudia la caracterización genética de un individuo con adenoma mediante un experimento de secuenciación dirigida.

Capítulo 4: muestra aportes realizados a la etapa de **modelado**. Se presenta un análisis objetivo de las herramientas actuales para el estudio de SA, lo que concluye en una guía para la selección del flujo de trabajo más adecuado según las características del experimento que se quiera analizar.

Capítulo 5: presenta la **herramienta desarrollada** para el análisis de DS. La misma se ha evaluado utilizando medidas objetivas de desempeño y se ha comparado con otras herramientas utilizando bases de datos con expresión controlada mediante simulación.

Capítulo 6: presenta las **conclusiones y trabajos futuros** producto de la presente tesis. Se destacan los diferentes aportes realizados al estado del arte, así como también las posibles líneas que se pueden continuar a partir de lo realizado a lo largo del doctorado.

Índice general

Capítulo 1

Minería de Datos

La extracción de conocimiento útil, implícito y previamente desconocido a partir de grandes volúmenes de datos es un proceso denominado **Descubrimiento de Información en Bases de Datos**, más comúnmente conocido del inglés como *Knowledge Discovery in Data bases* o **KDD**. El KDD abarca desde la comprensión de los datos y su preparación hasta la interpretación de los resultados obtenidos a partir de ellos. En particular, la **Minería de Datos** (del ingés Data Mining, DM), es una etapa dentro del proceso de KDD que involucra el análisis de los datos (Torralbo and Alfonso, 2010).

El KDD proporciona un marco de referencia *ordenado* de trabajo, aportando herramientas y dirigiendo el trabajo hacia la búsqueda de *información relevante*. Éste comprende distintas etapas que van desde la conceptualización de los experimentos, la obtención de los datos de entrada, el control de calidad, la adecuación e integración de distintas fuentes de datos, el análisis con las herramientas elegidas, hasta la presentación de los resultados mediante informes con visualizaciones apropiadas.

En este capítulo se desarrollan brevemente los conceptos del KDD y las distintas etapas de este proceso que guiarán el análisis de datos transcriptómicos con la finalidad de estudiar las modificaciones post-transcripcionales del ARN.

1.1. Generalidades

El KDD persigue la extracción automatizada de conocimiento no trivial, implícito, previamente desconocido y potencialmente útil a partir de grandes volúmenes de datos. Este proceso enfatiza la utilización de un método particular

de DM en un contexto de más alto nivel (Fayyad et al., 1996b). El concepto de DM refiere al proceso de descubrimiento de nuevos *patrones*, *tendencias* y *conocimiento* subyacentes en un conjunto de datos de gran volumen (Sumathi and Sivanandam, 2006). Las técnicas de análisis que se utilizan en DM provienen de diversas áreas: estadística, aprendizaje maquinal, recuperación de la información, etc. Entre las tareas del DM se destacan la clasificación, asociación, regresión, agrupamiento y la detección de datos no esperados (más conocidos como *outliers*). En este contexto, *eficiencia, escalabilidad, desempeño y optimización* son criterios claves a la hora de desarrollar nuevos algoritmos de DM (Han et al., 2011).

1.2. Objetivos

En términos generales, los dos principales objetivos del DM son la *descripción* y la *predicción*.

Descripción: pretende identificar patrones que explican o resumen los datos mediante la caracterización de las propiedades de los datos examinados.

Predicción: se focaliza en inferir relaciones entre los datos en un conjunto en particular, para luego poder predecir valores futuros de variables de interés por medio de otras variables que contiene la base de datos.

Estos objetivos pueden alcanzarse mediante la realización de algunas de las siguientes tareas:

Resumen: Comprende métodos que describen los datos en forma resumida. Los más utilizados son los métodos de estadística descriptiva clásicos como la media muestral, la mediana, la desviación estándar y el rango, y métodos de visualización como diagramas de cajas (*boxplots* en inglés), histogramas, diagramas de tallo y hoja, etc. (Walpole et al., 2012).

Agrupamiento: También conocido como *clustering*. Consiste en encontrar un agrupamiento natural de los datos en un número finito de categorías o grupos, los *clusters*, de manera de maximizar la similaridad intra-grupo y minimizar la similaridad entre grupos (Han et al., 2011). Estas categorías pueden ser jerárquicas, exclusivas o superpuestas. El agrupamiento puede tener fin *exploratorio*, si se desconoce los grupos que hay entre los datos, o *confirmatorio*, cuando antes de hacer el agrupamiento se tiene definidas

un número de categorías en las cuales los datos deberían agruparse (Jain et al., 1999).

Clasificación: Es el proceso de encontrar un modelo o función que describa la relación existente entre las características o atributos de los datos y la variable de interés o salida, la cual es del tipo categórico. Un problema de clasificación es un problema de aprendizaje supervisado ya que la construcción del modelo/función se realiza utilizando conjuntos de datos en los cuales la categoría/clase de la variable de salida se conoce. De esta manera, una vez ajustado el modelo/función, éste puede ser utilizado para predecir las clases de nuevos conjuntos de datos (Sumathi and Sivanandam, 2006).

Regresión: Es similar a la tarea de clasificación, con la particularidad de que la variable de interés o salida es del tipo numérica (Sumathi and Sivanandam, 2006).

Detección de cambios: Se centra en la detección de diferencias significativas en los datos, basándose en observaciones pasadas de los mismos.

En el contexto del *análisis transcriptómico*, la **predicción** implica la utilización de los datos para la predicción de expresión de genes, isoformas e incluso funciones biológicas que puedan estar modificadas por las variables de interés. Por otra parte, la **descripción** se focaliza en encontrar *patrones* o *relaciones* que proporcionen una explicación de los datos, que sea fácilmente interpretable por una persona.

1.3. Etapas

El KDD es un proceso *iterativo* e *interactivo* que involucra un conjunto de etapas sucesivas, las cuales por lo general se estructuran de la siguiente manera: i) Entendimiento del problema, ii) Entendimiento de datos, iii) Modelado, iv) Evaluación y v) Reportes.

1.3.1. Entendimiento del problema

Es la primera etapa, también conocida como **entendimiento del negocio**. En ella el investigador se interioriza, involucra y relaciona en los aspectos del problema a abordar. Durante esta etapa se plantean las preguntas a responder,

los diferentes procesos que van a generar los datos, el tipo de datos que van o no a estar disponibles, etc. De esta manera, es posible entonces comprender el problema, delimitarlo y definirlo de forma clara y concisa para así plantear los objetivos específicos que se abordarán para su resolución. En esta tesis, el dominio de aplicación es el *análisis transcriptómico*. Específicamente, ésta pretende analizar los cambios de expresión de genes e isoformas como consecuencia de alteraciones de los procesos post-transcripcionales del ARNm, mediante el análisis de datos de secuenciación masiva.

1.3.2. Entendimiento de los datos

En esta etapa se definirán los datos que se considerarán para resolver el problema planteado en la etapa anterior. El objetivo es construir una **base de datos** que contenga aquéllos que se supone podrían aportar información para la resolución del problema. Para ello, primero es necesario comprender los datos desde diversas perspectivas: el proceso mediante por el cual se generan, qué significado tienen, el tipo de variables con las que se trabajará, cuáles son los datos que necesitaremos como entrada del proceso y cuáles, si es que los habrá, se generarán como resultado del mismo. En general, al inicio de esta etapa se cuenta con un conjunto de datos de gran tamaño lo cual dificulta su manipulación.

La creación de la base de datos comienza con la **familiarización de los datos**. Esta tarea abarca una serie de actividades como revisar la base de datos, identificar el/los tipos de datos y sus atributos, revisar la integridad de los datos y consistencia de sus registros, aplicar transformaciones sobre los datos, etc. De esta manera será posible identificar problemas de calidad y descubrir signos iniciales que orienten las estrategias para detectar información oculta. Para ello se utilizan técnicas estadísticas de resumen y visualización, de manera tal de obtener una visión global del comportamiento de los datos. Entre las diferentes tareas es posible particularizar:

- **Obtención de un conjunto de datos**: Selección del conjunto de observaciones sobre las que se va a realizar el análisis, que se cree aportan información para responder a los objetivos propuestos, o bien, son potencialmente útiles para el proceso de descubrimiento de la información.

  En el contexto del *análisis transcriptómico* desarrollado en esta tesis, se tomará como punto de partida los datos generados por tecnologías de se-

cuenciación de segunda generación mediante la técnica RNA-seq. Esto comprende información generada mediante secuenciación de muestras biológicas e información provista por el fabricante (secuencia de adaptadores, códigos de barra, etc.).

- **Consistencia de los datos**: Refiere a que un concepto puede estar representado por diferentes nombres en diferentes bases de datos, así como diferentes conceptos pueden estar representados por el mismo identificador. Esto es posible cuando se trabaja con distintas fuentes de información tales como diferentes bases de datos de anotación de genes e isoformas.

  Habitualmente se contextualiza en operaciones entre bases de datos como la unión (*join* o *merge* en inglés), o la transformación o mapeo de identificadores de genes o transcritos de una a otra base. Aquellos identificadores que pertenezcan a ambas bases, representan datos consistentes, es decir, están mapeados. En caso contrario, no podrán ser utilizados en el análisis posterior salvo que se realice un proceso de transformación, posterior a su identificación.

- **Integridad de los datos**: Este concepto evalúa que los datos que conforman la base sean correctos y estén completos. Para ello es necesario conocer los posibles valores o rangos de cada uno de los atributos de forma de poder asegurar que no han sido alterados durante su registro o posterior consulta.

  Por ejemplo, en el caso de la expresión de un gen, siempre se esperan valores mayores o iguales a cero. Por lo tanto, si se identifica un gen cuyo valor de expresión es menor a cero, es sabido que este dato es erróneo. Además, el conocimiento del rango de valores de expresión permite evaluar potenciales valores extremos o atípicos. Estos valores deben identificarse, ya que en general requieren un tratamiento especial y suelen tener un impacto significativo en las distintas técnicas de DM.

- **Filtrado de los datos**: También llamado limpieza de datos. Abarca un conjunto de operaciones básicas a los efectos de *limpiar* los datos, en el sentido de eliminar aquellas características no deseable: ruido en la señal, datos de baja calidad, identificación y eliminación de sesgos, gestión de la ausencia de datos y datos atípicos.

En el contexto de las NGS, los datos pueden presentar sesgos propios de la secuenciación producidos por la longitud de los transcriptos, su nivel de expresión o su contenido en GC. Adicionalmente, los genes de nulo o bajo nivel de expresión constituyen una fuente de ruido para los modelos que analizarán DE, por lo cual deben ser removidos previo a su aplicación (Conesa et al., 2016).

- **Reducción, proyección e integración de datos**: Búsqueda de características relevantes, que permitan una mejor representación de los datos para el objetivo propuesto. Se pueden utilizar métodos de transformación para reducir el número efectivo de variables a considerar o para encontrar representaciones alternativas de los datos. Un enfoque multivariado clásico es la utilización de técnicas como análisis de componentes principales (*PCA* del inglés Principal Component Analysis, Bro and Smilde, 2014).

  En el contexto del análisis de datos de RNA-seq hay herramientas que transforman los datos de conteos de expresión a escalas continuas para poder luego aplicar modelos basados en este tipo de datos (Ritchie et al., 2015).

1.3.3. Modelado

Esta etapa es comúnmente denominada de DM. Consiste en la aplicación de algoritmos de **aprendizaje automático**, **inteligencia artificial** y **métodos estadísticos** para extraer **patrones** previamente desconocidos y potencialmente útiles del conjunto de datos previamente seleccionado (Sumathi and Sivanandam, 2006). Los patrones encontrados serán transformados a *conocimiento* con el fin de responder a las consignas planteadas en la etapa de "entendimiento del problema". Según Fayyad et al. (1996a), esta etapa comprende a las siguientes tareas:

- **Elección de la tarea de Minería de Datos**: Selección del tipo de tarea que se utilizará para alcanzar los objetivos. Es decir, se deberá definir si la solución se buscará mediante un proceso de *predicción* o *descripción* (ver Sección 1.2). Posteriormente, se estudiarán las técnicas aplicables al tipo de problema definido, así como también, los requerimientos e hipótesis específicas que deberán satisfacerse para su aplicación.

- **Ejecución de la tarea de Minería de Datos**: Realización del análisis de los datos. Se aplicarán los diferentes algoritmos y modelos computacionales, buscando aquellos patrones que caractericen los datos y puedan ser útiles para extraer el conocimiento buscado.

1.3.4. Evaluación

En esta etapa es donde se resalta la **naturaleza iterativa** del KDD. La etapa anterior producirá un conjunto de resultados obtenidos mediante los diferentes algoritmos y modelos estadístico/computacionales propuestos. Éstos deberán ser evaluados y comparados con el objetivo de seleccionar el mejor modelo. Dicha tarea deberá ser ejecutada en forma objetiva y estará dirigida a resolver los objetivos del problema planteado. También implicará la interpretación de los resultados en el contexto del problema bajo estudio. En muchas oportunidades, esta etapa se realiza conjuntamente a la etapa anterior ya que los modelos seleccionados para la tarea de DM deben ser constantemente evaluados y comparados para obtener el mejor posible (Fayyad et al., 1996b).

Esta etapa se basa fundamentalmente en estrategias de **validación**, con la finalidad de determinar la validez de los patrones encontrados sobre la base de datos. Una estrategia comúnmente utilizada en el análisis de cambios en la expresión de genes, isoformas o en el SA, es la evaluación de los modelos con **datos simulados** donde éstos y sus modificaciones son controlados (Conesa et al., 2016; Liu et al., 2014; Soneson and Delorenzi, 2013; Soneson et al., 2016).

1.3.5. Reporte

El ultimo paso en el proceso de KDD consiste en presentar la solución encontrada de forma adecuada, precisa y fácilmente comprensible. La presentación se deberá hacer a todas las partes involucradas: negocios/comercios, investigadores, médicos, o bien la comunidad científica en general. El tipo de reporte que se generará depende de los objetivos del proyecto. En general, se utilizarán como reportes los informes de avances, la presentación de gráficos, páginas web de difusión, artículos científicos y hasta la implementación de herramientas informáticas. Más allá del tipo de reporte que se seleccione, hay dos aspectos fundamentales que deberán ser considerados:

- **Visualización**: Existen diferentes maneras de mostrar la información: ta-

blas resumen, gráficos, páginas web e incluso informes tabulares. La técnica de visualización que se elija debe permitir no solo la visualización de los resultados sino además su exploración e interpretación.

- **Consolidación del conocimiento adquirido**: Incorporar el conocimiento adquirido mediante el proceso dentro del sistema, o simplemente documentar y presentar lo realizado a las partes interesadas, mediante un informe, un reporte final o incluso un artículo en una revista científica del ámbito académico.

Como se mencionó anteriormente, el proceso de KDD es un proceso iterativo por naturaleza; el hecho de finalizar alguna de las cinco etapas anteriores, no implica que no se deba volver a realizar alguna de ellas para considerar alguna corrección o modificación en busca de la mejor solución. La iteración se puede realizar en cualquier momento en que se considere oportuno, dado que la obtención de la solución del problema no es un proceso lineal (Fayyad et al., 1996a).

En el contexto del estudio de las modificaciones post-transcripcionales del ARN, se utilizan datos almacenados en archivos de texto de estructura compleja que pueden alcanzar las centenas de Gigabytes (Gb). Luego, la aplicación de métodos tradicionales de análisis donde un investigador, analista de datos o un científico manipula directamente los datos para extraer información y/o realizando búsquedas guiadas por su experiencia o pericia se ha vuelto impracticable. Sumado a esto, se presenta la complejidad de los datos, los cuales pueden también estar afectados por sesgos propios de la técnica con la que se han generado. Además, las escasas capacidades para visualizar eficientemente el conocimiento encontrado, generan restricciones que limitan los posibles análisis. Todas estas características convierten al problema en un desafío importante que requiere de la aplicación eficiente y ordenada de metodologías estadístico-computacionales para maximizar la extracción de conocimientos mediante la obtención de resultados confiables. En este contexto, el KDD y más aún el DM ofrece un marco apropiado para abordar este problema.

Capítulo 2

Análisis transcriptómico

2.1. Transcriptómica

En esta sección se abordan los conceptos involucrados en la primera etapa del proceso del KDD: *entendimiento del problema* (Sección 1.3.1). Los conceptos desarrollados a continuación son fundamentales para comprender la génesis y el contexto en el cual se enmarcan el problema y los datos que se utilizarán en su resolución.

2.1.1. ADN, ARN y proteínas

La unidad básica morfológica y funcional de un organismo vivo es la *célula*. Existen dos tipos de células, las *eucariotas*, que contienen un núcleo celular, y las *procariotas*. Estas últimas por lo general son en sí organismos unicelulares, mientras que las células eucariotas pueden ser individuos unicelulares o formar parte de organismos multicelulares como lo son las plantas o los animales (Sastry, 2010).
La célula eucariota consiste de tres componentes fundamentales: la **membrana**, el **núcleo** y el **citoplasma** (Figura 2.1). La membrana es una estructura fosfolipídica que delimita y protege a la célula y además controla el intercambio de sustancias entre ella y el ambiente que la rodea. El núcleo celular es un orgánulo que contiene la mayor parte del material genético, en forma de ácido desoxirribonucleico o **ADN**, altamente compactado en los cromosomas. El citoplasma es todo lo que se encuentra entre la membrana y el núcleo. Consiste de dos partes, una dispersión coloidal (citosol) conteniendo diferentes moléculas y un conjunto

de orgánulos celulares que desempeñan diferentes funciones. Entre las moléculas citoplasmáticas se destaca el ácido ribonucleico o **ARN** y las **proteínas**. El primero de ellos es una molécula obtenida a partir del ADN, entretanto las proteínas son cadenas de aminoácidos que tienen **funcionalidades básicas** tanto para el metabolismo como para la fisiología celular y consecuentemente del organismo (Parker and Honeysett, 2008).

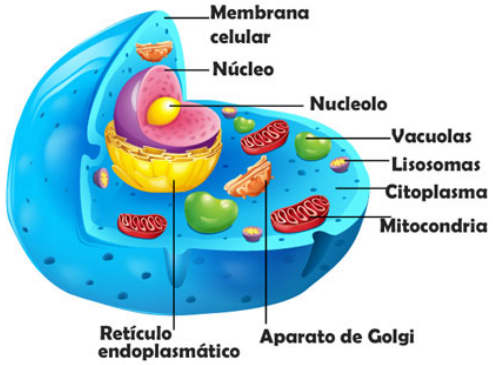

Figura 2.1: Diagrama simplificado de una célula eucariota. Imagen extraída de `http://lamaravilladelascelulas.blogspot.com.ar/p/partes-de-las-celulas.html`

El **material genétic**o está organizado en moléculas de doble cadena de ADN formadas a partir de nucleótidos conteniendo las bases nitrogenadas: adenina (**A**), citosina (**C**), timina (**T**) y guanina (**G**) (ver Figura 2.2). La *secuencia* de nucleótidos específica que forma cada cadena determina la información biológica propia del individuo. La doble hélice de ADN se mantiene estable gracias a la formación de puentes de hidrógeno[1] entre *las bases complementarias*, **A**-**T** y **C**-**G**, de cada una de las dos hebras. Como resultado de esta complementariedad, toda la información contenida en la secuencia de doble cadena de la hélice de ADN está duplicada en cada hebra. Las regiones relevantes del ADN se encuentran localizadas en los cromosomas y se denominan ***genes***. Los genes son las unidades hereditarias de los organismos biológicos y constituyen en su conjunto el *genoma* (Brooker, 2017). Si bien la cadena de ADN contiene millones de bases, sólo un pequeño porcentaje de ella codifica para proteínas. Por ejemplo, esta cantidad es aproximadamente entre el 1 y 3 % del genoma humano ($\approx 30Mb$) (Ng et al.,

[1]Los puentes de hidrógeno son enlaces químicos que se conforman por la gran fuerza electrostática atractiva entre un átomo electronegativo y un átomo de hidrógeno unido covalentemente a otro átomo electronegativo.

2009).

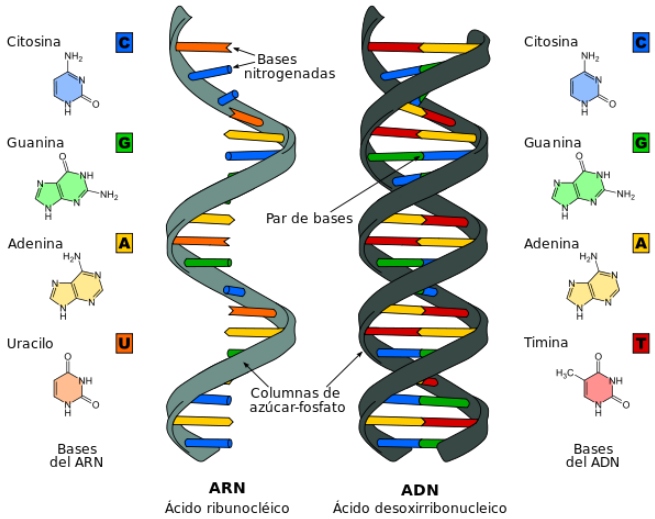

Figura 2.2: Estructura del ADN y del ARN. Imagen extraída de `https://www.lifeder.com/funciones-adn-arn/3`

La decodificación del material genético se inicia en el interior del núcleo celular donde el ADN es copiado a ARN en un proceso llamado *transcripción*[2]. El ARN (Figura 2.2) se encuentra en moléculas de una sola hebra, conformadas por cuatro nucleótidos en los cuales las bases **T** han sido reemplazadas por uracilos (**U**). Existen diversos tipos de ARN con diferentes funcionalidades: ARN mensajero (ARNm), micro ARN (miARN), ARN de transferencia (ARNt), ARN ribosomal (ARNr), ARN no codificante (ARNnc), ARN nuclear pequeño (ARNnp), etc. De todos ellos, el **ARNm** es el que posee la información que se utilizará para sintetizar proteínas, durante la *traducción* (Figura 2.3) (Sastry, 2010).

Los genes se componen de dos tipos de regiones: las que codificarán a proteínas y las que no, denominadas *exones* e *intrones*, respectivamente. La transcripción de ADN genera preARNm que es una copia directa del ADN. Posteriormente, el preARNm es procesado para conservar solamente los exones mediante un proceso llamado ***splicing***. Parte del ARNm procesado es luego traducido a proteína en el citoplasma, por medio de unos orgánulos llamados ribosomas. En este contexto, cada uno de los 64 posibles tripletes de los nucelótidos es llamado codón, de los cuales 61 codifican a uno de los 20 aminoácidos que conformarán las proteínas.

[2]La transcripción del ADN en eucariotas está mediada por una enzima, la ARN polimerasa II, la cual a partir de una de las dos hebras del ADN sintetiza la cadena complementaria de *ARN*, conservando así la secuencia contenida en el ADN.

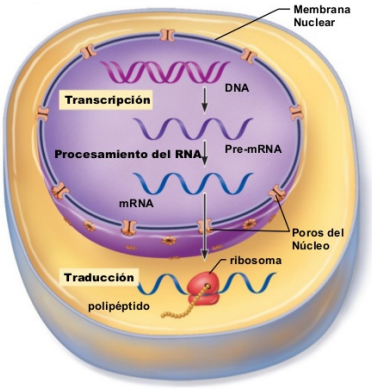

Figura 2.3: Esquema simplificado del flujo de información del ADN a las proteínas. Imagen extraída de `https://es.slideshare.net/gustavotoledo/sintesis-de-proteinas-39163433`

Luego, los ribosomas se encargan de formar la cadena de aminoácidos decodificados a partir de la secuencia de ARNm (Clark, 2005).

2.1.2. Splicing Alternativo

El proceso de splicing está mediado por un complejo molecular llamado *spliceosoma* conformado por ARNnp unido a proteínas formando ribonucleoproteínas. Este complejo es el encargado de la escisión de los fragmentos del preARNm que no estarán presentes en el ARNm. En la escisión de un intrón participan tres sitios, el sitio 5' donador de splicing, el sitio 3' aceptor de splicing y la secuencia de ramificación situada a unos 40 nucleótidos del extremo 3' del intrón (Figura 2.4), todos caracterizados por secuencias nucleotídicas específicas. El proceso de splicing comienza en el sitio donador de splicing donde la acción de las ribonucleoproteínas produce el primer corte sobre el intrón, el cual se pliega formando un lazo con el sitio de ramificación. Al mismo tiempo, otro grupo de estas moléculas se unen al extremo 3'. Como consecuencia del primer corte, los dos grupos de ribonucleoproteínas quedan enfrentados y se unen para formar el spliceosoma uniendo así los extremos de los exones y liberando el extremo 3' del intrón (Stamm et al., 2012).

Durante muchos años se sostuvo la teoría que un gen era transcrito a un ARNm que luego sería traducido a una única proteína, lo que se llamó **dogma central de la biología molecular**. El desarrollo de técnicas bioquímicas sofisticadas develó que en organismos eucariontes el número de proteínas diferentes

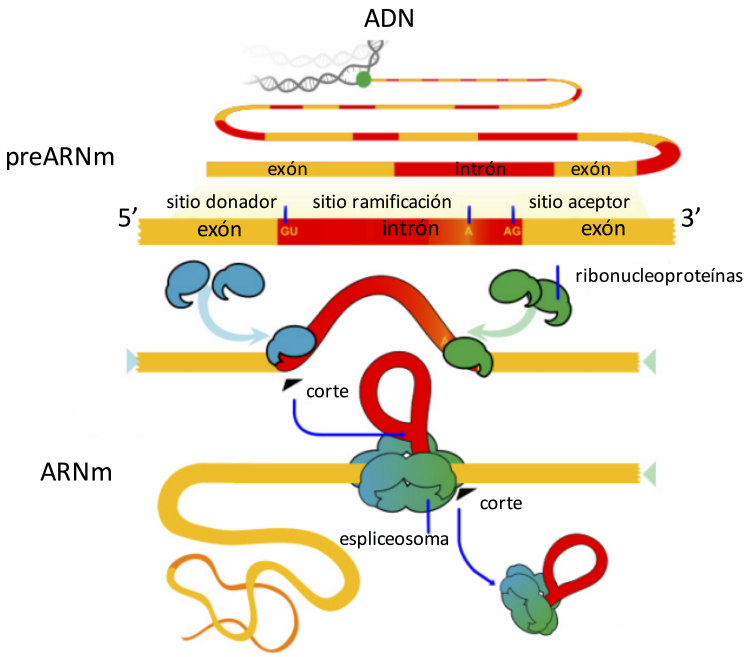

Figura 2.4: Proceso de splicing. Adaptación de la imagen extraída de `https://pt.wikipedia.org/wiki/Splicing`

era superior en varios órdenes al número de genes existentes en dicho organismo. En particular, hubo un hallazgo que cambió totalmente el paradigma biológico (Blencowe and Graveley, 2008). En 1978, Walter Gilbert contradijo el dogma central de la biología molecular, postulando que un gen puede dar origen a diversas proteínas como consecuencia de un proceso post-transcripcional al que llamaron ***Splicing Alternativo*** (SA) (Gilbert, 1978). A lo largo de los años posteriores se ha determinado que existe un conjunto de mecanismos post-transcripcionales y post-traduccionales que regulan la expresión génica, entre los cuales el SA es uno de los responsable de la diversidad biológica y complejidad funcional que caracteriza a los organismos eucariotas multicelulares (Gallego-Paez et al., 2017; Pan et al., 2008).

El descubrimiento del SA develó que en realidad el procesamiento de los preARNms generados a partir de un único gen involucraba mecanismos más complejos y diferentes a la simple eliminación de los intrones. Como consecuencia, el proceso de SA puede generar distintos *transcritos* de ARNm, también llamados ***isoformas***, que traducirán luego a proteínas diferentes incluso desde un punto de vista funcional (Figura 2.5) (Black, 2003). Los eventos de SA ocurren como con-

secuencia de tres tipos de variaciones durante el proceso de splicing: variaciones en el sitio donador, en el sitio aceptor y la selección de exones. Estas variaciones dan lugar a seis tipos de eventos de SA, ilustrados en la Figura 2.6. En particular, los mecanismos que involucran la selección de exones son los más frecuentes en humanos, en cambio, la retención de intrones es la más observada en plantas (Liu et al., 2014).

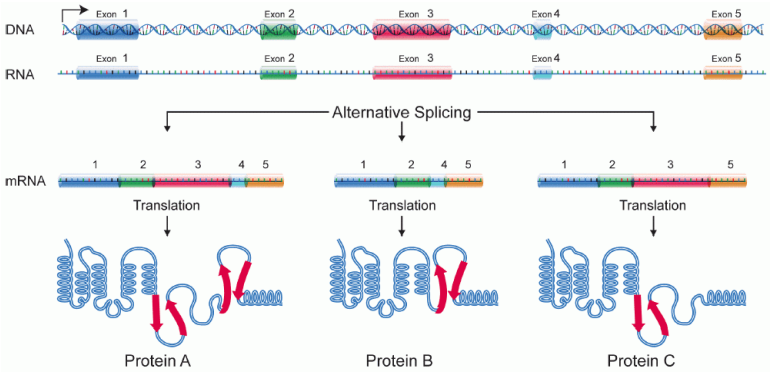

Figura 2.5: Ilustración del splicing alternativo en un gen conteniendo cinco exones. A modo de ejemplo, se muestran tres posibles ARNm obtenidos del procesamiento alternativo del preARNm, los cuales dan origen a tres proteínas diferentes. Adaptación de la imagen extraída de `https://en.wikipedia.org/wiki/Alternative_splicing`

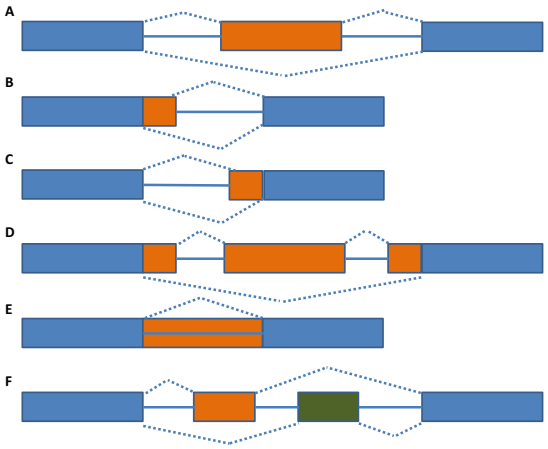

Figura 2.6: Tipos de splicing alternativo. **A)** Inclusión/exclusión de exones, **B)** sitio donador alternativo, **C)** sitio aceptor alternativo, **D)** patrones complejos de splicing, **E)**, retención de intrones y **F)** exones mutuamente excluyentes

2.1.3. Transcriptoma y su exploración

El conjunto de todos los transcritos de una célula se denomina **transcriptoma**. A diferencia del genoma, el cual es el mismo para todas las células de un organismo, el transcriptoma varía a lo largo del desarrollo y entre diferentes tipos de tejidos así como también en respuesta a cambios en el entorno. La composición del transcriptoma en una célula específica y en un momento específico revela cuáles son los genes que están siendo transcriptos. Esta información resulta clave para comprender e interpretar los elementos funcionales del genoma, revelar los componentes moleculares de células y tejidos y para entender procesos claves como el desarrollo y las enfermedades (Sánchez-Pla et al., 2012).

Transcriptómica refiere a la observación y análisis simultáneo de *todos* los transcritos (transcriptoma) en un momento dado (Wang et al., 2009). Diversas tecnologías han sido desarrolladas para estudiar el transcriptoma. La técnica convencional por excelencia para determinar los transcritos de una muestra es la *PCR cuantitativa en tiempo real* (*qRT-PCR*). Si bien esta técnica es la más precisa que se ha desarrollado, su aplicación al estudio de transcriptomas completos es inviable en términos de tiempo y costos económicos. En el año 1998 surgieron los **microarreglos de ADN**, los cuales permitieron la exploración simultánea de miles de genes ampliando el campo de estudio y aplicación de los experimentos transcriptómicos. Un microarreglo es una superficie sólida donde se han arreglado miles de oligonucleótidos o fragmentos pequeños de ADNc (*sondas*) en forma de matriz bidimensional. Estas *sondas* representan pequeñas secciones de los genes del organismo cuyo transcriptoma se desea explorar. Esta característica de los microarreglos constituye su principal limitación, ya que para su construcción es necesario conocer las secuencias de los genes que se desea estudiar. Sumado a esto, la capacidad de secuenciación está limitada por la dimensión física del microarreglo. Pese a sus limitaciones, los microarreglos han sido ampliamente utilizados incluso para el estudio del SA y diversas enfermedades (Gresham et al., 2008; Heller, 2002; Ramaswamy and Golub, 2002; Rhodes and Chinnaiyan, 2005; Stoughton, 2005; Van't Veer et al., 2002; Xu et al., 2002).

La era transcriptómica nació con los microarreglos de ADN pero tuvo su máximo desarrollo con el advenimiento de las tecnologías de secuenciación de alto rendimiento o **Next Generation Sequencing** (NGS). La exploración del transcriptoma completo en forma simultánea y a profundidades sin precedentes ha sido posible a partir de estas tecnologías. El término secuenciación refiere a

la identificación de la secuencia de nucleótidos que posee un fragmento de ADN (o ARN), mientras que la denominación masiva refiere al paralelismo en la secuenciación, lo que permite indagar millones de fragmentos de ADN en una sola ejecución. Las NGS surgieron comercialmente en el año 2005 superando altamente las limitaciones de tiempo, precisión, rendimiento y costo que presentaban las tecnologías de ese momento (Morozova and Marra, 2008). A modo ilustrativo, la Figura 2.7 muestra el impacto de las NGS en los costos de secuenciación de un genoma humano desde el año 2001 al año 2016. Su rápida expansión tanto en el

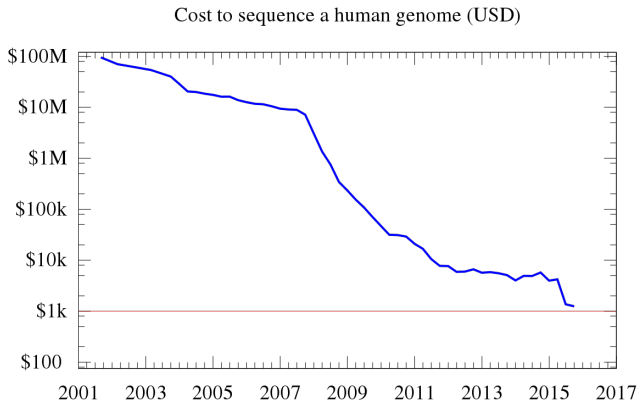

Figura 2.7: Evolución del costo de secuenciación de un genoma humano desde el año 2001 hasta el año 2016. Imagen extraída de `https://en.wikipedia.org/wiki/Whole_genome_sequencing`

ámbito científico como en el clínico se justifica además en su versatilidad, lo que ha permitido el estudio de experimentos complejos a escalas anteriormente inalcanzables e incluso en organismo nunca antes estudiado. A la fecha se han desarrollado decenas de aplicaciones de las NGS: *secuenciación de ADN* o *DNA-seq*, *secuenciación de ARN* o *RNA-seq*, *secuenciación de inmunoprecipitación de cromatina* o *ChIP-Seq*, *secuenciación dirigida*, *genotipificación por secuenciación GBS*, entre otras. Genómica funcional, genómica poblacional, transcriptómica, mejoramiento vegetal y medicina personalizada son apenas algunas de las tantas áreas que se han favorecido y desarrollado ampliamente en los últimos 12 años en base a tales aplicaciones (Goodwin et al., 2016). De todas las aplicaciones de las NGS, **RNA-seq** es la comúnmente utilizada para explorar el transcriptoma.

2.2. Creación del conjunto de datos transcriptómicos

En esta sección se detalla el proceso de creación del conjunto de datos, refiriendo a la etapa *entendimiento de los datos* del KDD (Sección 1.3.2). El primer paso en la creación del conjunto de datos transcriptómicos es el **diseño del experimento**. En este sentido, es clave respetar dos principios fundamentales para cualquier experimento donde se estudiarán variaciones como consecuencias del diseño experimental, para así asegurar la validez y eficiencia del estudio (Fang and Cui, 2011). Esos principios son:

- **Aleatorización:** Este principio implica que cualquier asignación que se deba realizar se debería hacer en forma aleatoria. Por ejemplo, la asignación de los tratamientos o condiciones a los sujetos bajo estudio, con el fin de eliminar cualquier efecto que pueda confundir o afectar los resultados. En el contexto de RNA-seq, durante la secuenciación existen etapas donde es posible aplicar este principio, que se discutirán más adelante.

- **Replicación:** La replicación es esencial para estimar y disminuir el error experimental y así detectar con mayor precisión el efecto biológico que se desea estudiar. Una verdadera réplica es una repetición de la misma condición experimental o tratamiento cuya realización y adquisición se hace de forma independiente. En este contexto, las réplicas pueden ser técnicas o biológicas. Las réplicas técnicas son muestras del mismo individuo que se utilizan para reducir errores de la técnica de extracción de la información, entretanto, las réplicas biológicas son repeticiones de la misma condición experimental en distintos sujetos. Las **réplicas biológicas** son fundamentales para cuantificar la variabilidad de dicha condición en la población por lo que son recomendadas en el contexto de las NGS (Marioni et al., 2008). Mientras más réplicas biológicas se puedan generar, mejores serán las estimaciones. No obstante, el número de réplicas a utilizar es un tema de discusión y evaluación permanente que suele estar principalmente limitado por el presupuesto disponible para la secuenciación (Conesa et al., 2016; Robles et al., 2012).

Durante el diseño del experimento, se deberán tomar decisiones importantes respecto de la elección de parámetros y/o protocolos para la creación del conjunto

de datos de secuenciación. Con el objetivo de comprender dicho proceso y la implicancia de las decisiones que se tomen, se describe a continuación el proceso de generación de datos transcriptómicos mediante la técnica de RNA-seq.

2.2.1. Secuenciación de alto rendimiento

La creación de los datos de secuenciación mediante RNA-seq está definida por la tecnología NGS que se utilice para tal fin. No obstante, independientemente de la tecnología, el procesamiento general para llevar a cabo la secuenciación consta de los mismo pasos. El primero de ellos siempre es la **extracción** de las moléculas de interés en la muestra biológica. En el caso de que el objetivo de análisis es la expresión génica, el ARN se extrae y retrotranscribe para obtener ADN complementario (ADNc) a estas cadenas. El proceso que sigue es el de la **preparación de la librería** que tiene como objetivo preparar a los fragmentos para la tercer etapa: la **secuenciación**.

Extracción de ARNm

Idealmente, en un estudio transcriptómico cuyo fin es el análisis de expresión génica, el paso de **extracción** debería poder captar todo el ARNm circundante en la muestra biológica. El resto del ARN: ARNnp, miARN, ARNr, etc. deberá ser eliminado. Respecto de las metodologías que se utilizan, no se entrará en detalle ya que escapan del tema de esta tesis. Lo importante es destacar que se utilizan técnicas que extraen todo el ARN y luego *purifican* o *aislan* el ARNm (Marioni et al., 2008).

Preparación de la librería

Esta etapa se inicia con la *fragmentación* del ADNc y continúa con *ligación de adaptadores*[3] que se utilizarán para una etapa de amplificación clonal. La fragmentación del ADNc es un proceso de cortes *aleatorios*, cuya finalidad es obtener fragmentos más cortos, entre 300 y 1000 pares de base (pb), que los transcritos originales. Cada uno de estos fragmento de ADNc es considerado una *plantilla* que será posteriormente clonada muchas veces gracias a la acción de enzimas polimerasas. Esta etapa de amplificación permite generar millones de copias de cada fragmento de ADNc en la muestra, con el objetivo de aumentar

[3]Un adaptador es un oligonucleótido de secuencia conocida.

la precisión en la identificación de las secuencias de dichos fragmentos.(Goodwin et al., 2016). Las tecnologías *SOLiD* y *Ion Torrent* utilizan la **amplificación por PCR en emulsión** para preparar la librería, en cambio, las plataformas *Illumina* emplean la técnica de **PCR en puente** (Metzker, 2010; Shendure and Ji, 2008).

Secuenciación masiva

Finalmente se realiza la **secuenciación** propiamente dicha y se graba la información de la secuencia identificada, en forma simultánea para todos los fragmentos. En este paso, la secuencia de cada fragmento original de ADNc es inferida a partir de los resultados observados para sus copias en la librería. Los principios de secuenciación que siguen las NGS se pueden agrupar en dos grandes categorías: *secuenciación por síntesis* y *secuenciación por ligación*. Los enfoques basados en secuenciación por síntesis (*Illumina* y *Ion Torrent*) utilizan la enzima ADN polimerasa II y una señal (un fluoróforo o un cambio de concentración iónica) para identificar la incorporación de los nucleótidos en una cadena en crecimiento. Por otro lado, en la secuenciación por ligación (*SOLiD*) se utiliza la enzima ligasa y un conjunto de sondas[4] conteniendo nucleótidos marcados con fluoróforos. Las sondas compiten entre sí para ser ligadas al fragmento de ADN que se está indagando. El color de los fluoróforos determina los nucléotidos incorporados a la cadena en crecimiento (Goodwin et al., 2016; Shendure and Ji, 2008).

Independientemente de la tecnología, el proceso de secuenciación genera un conjunto de secuencias identificadas a partir de la librería, conocidas como *lecturas de secuenciación*. Es importante destacar que las lecturas generadas mediante la técnica de *Illumina* tendrán todas la misma longitud. Estos secuenciadores ofrecen una amplia variedad de combinación de cantidad y longitud de lecturas generadas, dependiendo del equipo y el kit químico que se utilice para la secuenciación. Es así que las lecturas podrán tener longitudes de 25pb, 36 pb, 50pb, 75pb, 100pb, 150pb, 250 pb o 300 pb, mientras que en una corrida se podrán generar entre 12 y 4.000 millones de ellas (Shendure and Ji, 2008). Al igual que en la secuenciación con *Illumina*, la longitud de las lecturas de los secuenciadores *SOLiD* también es fija y podrá ser de 50 pb y 75 pb. Según los fabricantes, una corrida de secuenciación con estos equipos puede generar aproximadamente entre 700 y 1.400 millones de lecturas (Metzker, 2010). Contrariamente, las lecturas

[4]Fragmento de ADNc de a lo sumo 10 pares de bases

generadas por los equipos *Ion Torrent* tendrán, por lo general, distinta longitud dependiendo de la frecuencia y longitud de los homopolímeros contenidos en los fragmentos de la librería. Dependiendo del kit que se utilice se pueden conseguir lecturas de hasta 400pb, con un promedio de 200pb. La *cantidad total* de lecturas depende del equipo y el *chip* que se han seleccionado. Con el equipo *Ion PGM* y el chip más pequeño se pueden obtener 400.000 lecturas, mientras que con el *Ion Proton* se puede llegar a alcanzar los 80 millones de lecturas (Goodwin et al., 2016).

2.2.2. Protocolos de secuenciación

Las tres tecnologías descritas anteriormente comparten dos características importantes que tienen una gran influencia en los datos generados. La primera, es que todas ellas pueden utilizar protocolos de secuenciación *single-end* o *paired-end*. Éstos determinan cómo se secuenciará un fragmento de ADNc. Específicamente, el primero de ellos implica la secuenciación del fragmento en un sólo sentido (3′-5′ o 5′-3′), en cambio, el protocolo paired-end implica la secuenciación en ambos sentidos del mismo fragmento (ver Figura 2.8). Como consecuencia, el protocolo single-end genera una lectura para cada fragmento, mientras que el paired-end genera dos lecturas por plantilla. En particular, estas lecturas podrán estar solapadas, si el fragmento del que se originaron resultó más corto que la cantidad de ciclos de secuenciación o separadas por una distancia estimable (Goodwin et al., 2016).

Si bien la secuencia intermedia entre dos lecturas paired-end se desconoce el hecho de tener dos en vez de una lectura puede favorecer los procesamientos bioinformáticos posteriores. Por lo general, en experimentos de RNA-seq dirigidos al estudio del SA es preferible el protocolo *paired-end* ya que genera el doble de datos para el mismo fragmento de ADN y ha demostrado ser más sensible y específico que el *single-end* (Fang and Cui, 2011; Williams et al., 2014).

2.2.3. Multiplexado de muestras

La segunda característica en común que tiene las NGS es que permiten el *multiplexado* de individuos en una misma corrida de secuenciación. Este proceso es una forma de *aleatorización* que impide que algún efecto de secuenciación influencie sólo unas muestras en particular, ya que todas estarán sometidas a él.

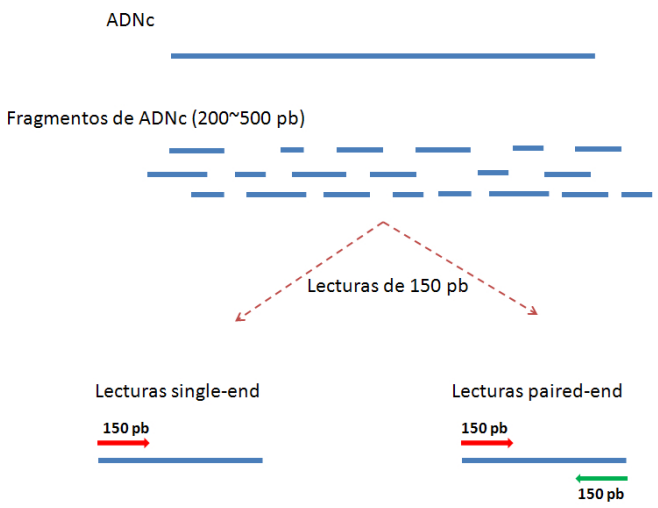

Figura 2.8: Protocolos de secuenciación *single-end* y *paired-end*.

El proceso de *multiplexado* utiliza etiquetas moleculares, conocidas como *códigos de barra* (*barcodes*), para identificar a cada una de las muestras biológicas. Los *códigos de barra* son secuencias cortas (menos de 20pb) que se añaden a los fragmentos de ADNc, luego de los adaptadores, durante la **preparación de la librería**. Una vez que esto se ha realizado para todas las muestras, éstas son mezcladas para formar una única solución conteniendo los fragmentos que serán posteriormente amplificados y secuenciados. Cada código tiene una secuencia diferente y conocida que luego será utilizada para identificar las lecturas de cada muestra (Robles et al., 2012).

La selección del protocolo de secuenciación y la realización de multiplexado son decisiones que se deben tomar considerando un parámetro importante del diseño de la secuenciación que es la *profundidad* o *cobertura* que se desea obtener. Ésta es una medida aproximada de cuántas veces se leerá un transcrito, considerando que todos tiene el mismo nivel de expresión. Indirectamente, la *cobertura* esta determinada por la cantidad de lecturas que generarán para cada muestra, lo que se conoce como el *tamaño de la librería*. Mientras más lecturas se generen, más veces se habrán secuenciado los transcritos, por ende mayor será la profundidad de lectura. Sin embargo, como cada gen genera un número diferente de transcritos, los cuales se expresan en distintos niveles, la cobertura no resulta nada uniforme. Mientras más veces se haya expresado una isoforma, más transcritos habrá generado, lo cual se traducirá en un más fragmentos de ADNc que

originarán más lecturas de secuenciación. De forma análoga, cuanto más largo sea un transcrito, más fragmentos de ADNc y por ende más lecturas se obtienen. Es así que para poder capturar todos los transcritos de baja expresión y/o cortos es necesario incrementar la profundidad de secuenciación. Ahora bien, a medida que se aumenta la profundidad de secuenciación por muestra se reduce el número posible de individuos a multiplexar y/o secuenciar en la misma corrida. En los estudios transcriptómicos dirigidos a evaluar cambios en el transcriptoma como respuesta a distintas condiciones experimentales, se recomienda en lo posible aumentar siempre el número de réplicas antes que la profundidad para mejorar la exactitud y eficiencia de los experimentos de RNA-seq. Por consiguiente, la elección del secuenciador, del kit de secuenciación y la cantidad de muestras a multiplexar son decisiones que se deben tomar en forma conjunta durante el **diseño del experimento** (Liu et al., 2013; Sims et al., 2014).

2.2.4. Alineamiento contra referencia

Las lecturas de secuenciación son almacenadas en archivos en formato *FASTQ*. Por lo general, se generarán tantos archivos *FASTQ* como muestras diferentes se hayan secuenciado. Adicionalmente, si el protocolo de secuenciación utilizado es el *paired-end*, es posible que se obtengan dos archivos para cada muestra, cada uno conteniendo una de las dos lecturas del par. El formato *FASTQ* es de tipo texto, almacena las **secuencias de las lecturas** y los valores de **calidad** asociados a cada una de las bases que las componen. La calidad del proceso de identificación de cada nucleótido durante la secuenciación se mide en escala Phred. Un valor Q en esta escala está relacionado logarítmicamente con la probabilidad P de que se haya cometido un error al identificar al nucleótido en cuestión, según la Ecuación 2.1.

$$Q = -10log_{10}P \tag{2.1}$$

Luego, un valor $Q = 10$ se corresponde con un valor $P = 0{,}1$, es decir que la probabilidad de que el fragmento original *NO* haya tenido la base reportada en dicha posición es 1 en 10; equivalentemente, indica que la exactitud en la estimación del nucleótido es del 90 %. Cada lectura de secuenciación tiene asociadas 4 líneas de texto en dicho formato, como se muestra en la Figura 2.9. La primera línea siempre comienza con el carácter "@" que se utiliza para identificar el comienzo de

una nueva lectura. Este carácter está acompañado de la descripción de la lectura, la cual puede contener información referente al secuenciador o al experimento. La segunda línea contiene la secuencia, entretanto la tercera siempre contiene un signo "+" y sirve para indicar el fin de la secuencia y el comienzo de la cuarta línea, la cual contiene la calidad de cada base leída. Estos valores de calidad se codifican con caracteres ASCII, desplazados positivamente en 33 unidades, para simplificar el almacenamiento. Así, por ejemplo, el primer valor de calidad en la Figura 2.9 es un "!" que en código ASCII le corresponde el número 33, luego como la escala de calidad está desplazada en 33, el valor Q al que refiere este carácter es 0. De la misma forma, se determina que el segundo carácter, " ' ", representa en código ASCII al número 39, luego el valor Q que codifica es 6.

```
@SEQ_ID
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

Figura 2.9: Ilustración de cuatro líneas de un archivo en formato de FASTQ.

En el contexto del análisis de expresión mediante la técnica de RNA-seq, las lecturas de secuenciación contienen, en principio, toda la información necesaria para cuantificar el transcriptoma de la especie bajo estudio. Sin embargo, las lecturas necesitan ser procesadas para poder extraer dicha información. El primer paso en este procesamiento siempre es la reconstrucción de los transcritos a partir de las lecturas con el fin de determinar de qué genes o isoformas provienen. Una alternativa es realizar lo que se conoce como un **ensamblado de novo** de las lecturas para inferir las secuencias de los transcritos sin utilizar más información que la contenida en las lecturas. Por otro lado, el conocimiento a priori del genoma o transcriptoma de la especie bajo estudio proveerá de una **referencia** que permitirá acelerar y hacer más precisa esta etapa de reconstrucción, la cual recibe el nombre de **alineamiento contra referencia**. Por lo general, las especies más estudiadas, como por ejemplo el humano, cuentan con dicha información, la cual se encuentra almacenada en *bases de datos de anotación*. Éstas contienen la información de anotación de genes e isoformas en un formato ordenado disponible para toda la comunidad científica. A saber, cada gen/isoforma es asociado a un identificador (ID), un cromosoma y una posición sobre él, una secuencia de nucleótidos y una relación gen-isoformas correspondiente. En particular, durante la

etapa de alineamiento sólo se utilizarán las secuencias del genoma/transcriptoma de referencia.

El proceso de **alineamiento** se conoce también como **mapeo**, y consiste en *alinear* las lecturas de secuenciación a las secuencias de la referencia de manera de asignarles una posición en dicho genoma/transcriptoma, dependiendo si se quiere identificar genes o isoformas. En términos sencillos, si se piensa que las lecturas son las piezas de un rompecabezas, el genoma/transcriptoma de referencia es la imagen que se obtendrá al unir correctamente las piezas. Dado que los genes se transcriben más de una vez y que la secuenciación de los fragmentos de ADNc es un proceso *aleatorio*, el proceso de secuenciación generará lecturas repetidas y/o solapadas de las distintas isoformas.

El alineamiento de las lecturas de secuenciación es un proceso complejo porque la referencia nunca será una representación perfecta del genoma o transcriptoma que ha dado origen a la secuenciación ya que cada individuo presentará sus propias variaciones (SNPs, InDels, etc). Uno de los principales problemas que se presenta durante este proceso es la ambigüedad a la hora de asignar una posición a una lectura, fundamentalmente cuando se utilizan lecturas muy cortas y/o se estudian especies cuyos genomas contienen muchas regiones repetidas. Otro inconveniente que se encuentra cuando se alinean lecturas contra el genoma de referencia, es que éstas no representan genes sino ARNm. Luego, es posible que una porción de la lectura corresponda a un exón y otra porción, a otro, como se ilustra en la Figura 2.10. La referencia sobre la que se realiza el mapeo es un fragmento de ADN, por lo tanto se compone de exones e intrones, por el contrario, las lecturas representan fragmentos cortos del ARNm sintetizado a partir de ese ADN y posteriormente procesado. Es así que se generarán lecturas representando regiones exclusivas de un exón (en celeste) y lecturas conteniendo sitios de uniones de dos exones contiguos (en rojo). Luego, durante la etapa de alineamiento, se encontrará con que las lecturas rojas no mapearán a una región de ADN y se deberá recurrir a distintas estrategias para poder ubicarlas correctamente (Wolf, 2013).

Diversas herramientas se han desarrollado para realizar el mapeo de las lecturas de secuenciación obtenidas mediante RNA-seq (Engström et al., 2013). Los programas que realizan mapeo contra el *genoma* de referencia se denominan *spliced mappers* ya que son exclusivos para trabajar con lecturas generadas a partir de ARNm, considerando la posibilidad de SA durante el mapeo. Para ello utilizan diferentes estrategias. Entre los programas más utilizados se destacan `TopHat2`

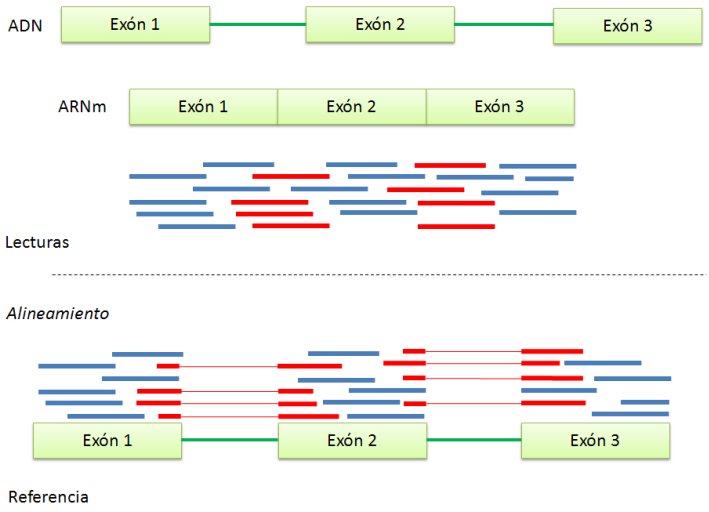

Figura 2.10: Ilustración de los tipos de lecturas que se obtendrán al secuenciar transcritos. Cada gen es transcrito a ARN y luego procesado para formar los ARNm que contiene regiones exónicas. Los ARNm son secuenciados obteniendo lecturas que representarán regiones de uno (celeste) o dos exones contiguos (rojo).

(Kim et al., 2013) y `STAR` (Dobin et al., 2013). Éstos dividen el proceso de mapeo en dos etapas. En primer lugar, intentar mapear todas las lecturas a una única región, para lo cual suelen permitir un número máximo de posiciones sin coincidencia con la referencia. Luego, intentan alinear las lecturas que no fueron mapeadas considerando que éstas pertenecen a regiones genómicas separadas por porciones de ADN que se removieron durante el procesamiento del ARN. Cuando el alineamiento es contra el *transcriptoma* de referencia, se utilizan programas que permiten que las lecturas mapeen a más de una región, es decir, a más de un transcrito anotado. Notablemente, el alineamiento es más sencillo que cuando se usa el genoma como referencia. Entre los algoritmos que hacen mapeo contra transcriptoma se destacan `BWA` (Li and Durbin, 2009) y `Bowtie` (Langmead et al., 2009). En particular, éstos han sido desarrollados para mapear lecturas de secuenciación originadas a partir de fragmentos de ADN contiguos, por lo que la aplicación al alineamiento de transcriptoma es una de las tantas para las cuales son utilizados. Como contrapartida, el alineamiento contra transcriptoma no permite la identificación de nuevos eventos de SA que generen isoformas no contenidas en el transcriptoma de referencia. En ambas situaciones el uso del protocolo de secuenciación *paired-end* contribuye a la toma de decisión de posición de una lectura ya que se poseen dos lecturas, separadas por una distancia estimada, que deben coincidir con la referencia (Hatem et al., 2013).

Como resultado de la etapa de alineamiento, se genera para cada lectura un archivo de alineamiento en formato *SAM* (Sequence Alignment/Map) o *BAM* (Binary Alignment/Map) (Li et al., 2009). El archivo *SAM* es un fichero de texto tabulado, mientras que el formato *BAM* es su versión comprimida. Un archivo de alineamiento se compone por una *sección de cabecera*, opcional, y una *sección de alineamiento*. Las líneas de estas dos secciones se diferencian en que las de la cabecera empiezan con el carácter "@". La *sección de alineamiento* contiene 11 campos obligatorios de información esencial acerca del alineamiento y un número variable de campos opcionales de información más específica. Los campos obligatorios describen el alineamiento inferido para cada lectura. Es decir, si ha sido o no alineada, a qué secuencia (cromosoma o transcrito) y en qué posición, el número de coincidencias entre la secuencia y la referencia. Particularmente, hay un campo llamado *MAPQ* que indica la calidad de mapeo. Éste es un valor calculado según la Ecuación 2.1, donde P ahora es la probabilidad de que la posición sea incorrecta.

2.2.5. Cuantificación del nivel de expresión

El proceso biológico en el cual la información de un gen es utilizado para la síntesis de productos génicos funcionales se conoce como ***expresión génica***. Usualmente estos productos son proteínas sintetizadas a partir de ARNm, aunque también pueden ser ARNm, ARNt, ARNnp, entre otros (Brooker, 2017). La cantidad de ARNm transcritos a partir de un gen se conoce como su ***nivel de expresión*** y la cantidad de ARNm iguales en su secuencia, representando una misma isoforma del gen, se conoce como *nivel de expresión* de dicha isoforma. A modo ilustrativo, la Figura 2.11 ilustra estos conceptos a partir de un gen con cuatro exones que como consecuencia del SA da origen a tres isoformas diferentes (Figura 2.11**A**). Luego, si la cantidad de transcritos para cada isoforma es la ilustrada en la Figura 2.11**B**, entonces el nivel de expresión de la Isoforma I será 51, el de la Isoforma II, 15 y el de la Isoforma III, 84. Por lo tanto, el nivel de expresión del gen será $51 + 15 + 84 = 150$ y las proporciones de las isoformas estarán dadas por el porcentaje de la expresión del gen que cada una de ellas representa (Figura 2.11**C**). La proporción de las isoformas también se conoce como *expresión relativa*.

En el contexto del análisis de datos transcriptómicos, una vez que se ha asignado a cada lectura una posición o ubicación en la referencia, la siguiente tarea

es la **cuantificación**. Ésta consiste en resumir y agregar las lecturas obtenidas sobre una región con relevancia biológica como puede ser un gen, una isoforma o incluso un exón. El número de lecturas obtenidas de un fragmento de ADN es una medida indirecta del nivel de expresión de dicha región. El proceso de cuantificación en individuos bien caracterizados, como el humano, es asistido por archivos de anotación. En ellos se especifica la identidad de los exones, isoformas y genes del genoma así como también la relación entre ellos. La herramienta que se utilice tendrá como objetivo determinar a qué región anotada pertenece una lectura para posteriormente agregar las lecturas de cada región anotada. En particular, es posible realizar la cuantificación a nivel de *exones*, *isoformas* y *genes*.

Cuantificación de exones

La cuantificación de exones tiene como objetivo asignar una estimación del valor de expresión de cada uno de los exones anotados en los genes identificados en cada una de las muestras secuenciadas. Para ello toma como partida los archivos generados por el alineamiento de las lecturas contra el *genoma de referencia* y el archivo de anotación de referencia. De los tres niveles de cuantificación es el más sencillo ya que, por lo general, los exones no se solapan y son fragmentos cortos (Oshlack et al., 2010).

La forma más sencilla de resumir las lecturas a nivel de exón es contar cuántas de ellas caen dentro de la región exónica. Por supuesto, habrá lecturas que no estarán totalmente contenidas en un exón, ya que pueden haberse obtenido de una región de ADNc perteneciente a dos exones contiguos. Por lo general las herramientas que cuantifican exones permiten elegir al usuario qué hacer con estas lecturas. El enfoque más conservativo es considerar solamente las lecturas completamente contenidas en un único exón. En el otro extremo, la versión más relajada permite que las lecturas coincidan en al menos una sola base con un exón para que sean consideradas como tales (Anders et al., 2012).

Cuantificación de isoformas

El proceso de alineamiento contra transcriptoma de referencia asigna a cada lectura al menos una posición en las isoformas contenidas en dicha referencia. En el proceso de cuantificación, se debe decidir de qué isoforma proviene cada lectura. Para ello, las primeras estrategias propuestas recomendaban la cuantificación en base sólo a aquellas lecturas que se alinearon a una única isoforma. Si bien ésto

funciona para algunos genes que han sido afectados por SA, no lo hace en aquellos genes que no contienen exones únicos para sus isoformas. Luego, dado que las isoformas de un gen comparten muchos de sus exones, su cuantificación es uno de los procesos más complejos (Garber et al., 2011).

Las herramientas desarrolladas para ejecutar esta tarea se basan en diversas estrategias y modelos estadísticos que permiten considerar la incerteza a la hora de decidir si una lectura ha sido generada de una u otra isoforma de un gen. Por ejemplo, el programa `RSEM` propone un modelo estadístico para estimar la probabilidad de que una lectura provenga de un determinado transcrito y utiliza el algoritmo de maximización de la esperanza para estimar las abundancias de los transcritos a partir de las probabilidades estimadas (Li and Dewey, 2011). Por otro lado, las herramientas más nuevas, como `Kallisto`, no requieren de una etapa previa de alineamiento preciso de las lecturas sino que basan sus estimaciones en pseudo-alineamientos. Para ello combina el uso de tablas hash con gráficos de Bruijn creados a partir del transcriptoma de referencia (Bray et al., 2016).

Cuantificación de genes

Tal y como se definió anteriormente, el nivel de expresión de un gen es la suma del nivel de expresión de todos sus transcritos. De manera que una forma sencila de estimar el nivel de expresión de un gen es a través de la suma de los niveles de expresión estimados para todas sus isoformas. Sin embargo, la cuantificación de isoformas es en sí un desafío demasiado complejo para abordar si sólo interesa conocer la expresión de los genes. Es así que los desarrolladores de los primeros programas empleados para la cuantificación de genes utilizaron estrategias alternativas para estimar los niveles de expresión prescindiendo de las estimaciones a nivel de las isoformas. En general, los dos enfoques más utilizados son los denominados: método de *intersección* de exones y el método de *unión* de exones. El método de *intersección* de exones cuantifica las lecturas que han mapeado a los exones que son incluidos en todas las isoformas del gen; por otro lado, el método de *unión* de exones cuenta las lecturas alineadas a cualquiera de los exones que forman las isoformas del gen (Garber et al., 2011). Si bien la cuantificación de genes basada en modelos de exones es sencilla, no está libre de limitantes. Por ejemplo, si un gen tiene isoformas poco similares, el número de exones compartidos por todas ellas será bajo, lo cual conducirá a una subestimación de la expresión por parte del método de intersección de exones. En el caso del método

de *unión* de exones, la subestimación de la expresión también ocurre, más aún cuando las isoformas son muy parecidas entre sí. Además, estos métodos pierden precisión como consecuencia de no estimar cuáles de las isoformas anotadas del gen están en realidad expresándose. Durante los últimos años, el desarrollo de métodos de estimación de expresión de isoformas robustos y eficientes en términos computacionales ha facilitado la cuantificación a nivel de genes basada en la expresión de los transcritos, lo cual es biológicamente más razonable que los métodos basados en modelos de exones (Zhao et al., 2015).

El resultado de la etapa de **cuantificación del nivel de expresión** es la **matriz de expresión**. Esta matriz podrá estar a nivel de *exones*, *isoformas* o *genes*. Cada fila de la matriz será una de estas unidades genómicas y cada columna representará una de las muestras experimentales. Luego, si en un experimento en el que se ha considerado P muestras, se han identificado N genes (isoformas o exones) la **matriz de expresión** será de la forma expresada en la Ecuación 2.2, donde m_{ij} indica el valor de expresión del i-ésimo gen (isoforma o exón) en la j-ésima muestra. En particular, en el caso de que esta matriz represente isoformas o exones, los análisis requerirán de una segunda matriz que relacione cada isoforma o exón con el gen de donde proviene.

$$ME = \begin{pmatrix} m_{11} & \dots & m_{1j} & \dots & m_{1P} \\ \vdots & & \ddots & & \\ m_{i1} & \dots & m_{ij} & \dots & m_{iP} \\ \vdots & & \ddots & & \\ m_{N1} & \dots & m_{Nj} & \dots & m_{NP} \end{pmatrix} \tag{2.2}$$

2.3. Análisis de expresión diferencial

El análisis transcriptómico por lo general tiene como objetivo determinar qué genes/isoformas evidencian cambios en su expresión cuando se comparan distintas células o condiciones experimentales, ya que éstos serán los responsables de las diferencias entre las mismas. Por ejemplo, estudiar los genes que modifican su expresión en personas enfermas respecto de personas sanas, contribuye con la identificación de genes responsables de patologías humanas, lo cual resulta fundamental tanto para su comprensión como para su tratamiento. Este tipo de análisis es el que se conoce como ***análisis de expresión diferencial***.

El *análisis de expresión diferencial* en transcriptómica se realiza sobre la **ma-**

triz de expresión obtenida en un experimento de RNA-seq. Este análisis también considera la **matriz de diseño** asociada al experimento, la cual tiene en las filas las P muestras analizadas (columnas de la matriz de expresión) y en columnas los M factores o condiciones actuando sobre dichas muestras. Por ejemplo, supongamos un experimento cuya matriz de diseño es la definida según la Ecuación 2.3. En este caso, cada muestra está asociada a dos factores: *condición* y *protocolo de secuenciación*. Es así que de las 12 muestras biológicas secuenciadas, seis están asociadas al nivel Control y seis al nivel Tratamiento del factor condición. Por otro lado, la mitad de las muestras (tres de cada condición) han sido secuenciadas con el protocolo single-end y la otra mitad (también tres de cada condición), con el protocolo paired-end.

$$MD = \begin{array}{c} \\ Muestra_1 \\ Muestra_2 \\ Muestra_3 \\ Muestra_4 \\ Muestra_5 \\ Muestra_6 \\ Muestra_7 \\ Muestra_8 \\ Muestra_9 \\ Muestra_{10} \\ Muestra_{11} \\ Muestra_{12} \end{array} \begin{array}{c} \begin{array}{cc} Condicion & Protocolo \end{array} \\ \left(\begin{array}{cc} Control & Single-end \\ Control & Single-end \\ Control & Single-end \\ Control & Paired-end \\ Control & Paired-end \\ Control & Paired-end \\ Tratamiento & Single-end \\ Tratamiento & Single-end \\ Tratamiento & Single-end \\ Tratamiento & Paired-end \\ Tratamiento & Paired-end \\ Tratamiento & Paired-end \end{array} \right) \end{array} \tag{2.3}$$

Dada la cantidad de experimentos de RNA-seq realizados a lo largo de los años, algunos consorcios han desarrollado plataformas o repositorios donde se han depositado los conjuntos de datos generados por estos experimentos. Algunos de los repositorios públicos más utilizados son *Sequencing Read Archive* (*SRA*, Leinonen et al., 2010) y *The Canger Genome Atlas* (*TCGA*, Weinstein et al., 2013). Por lo general, cada experimento estará disponible en uno de los dos formatos siguientes: **datos crudos** (lecturas de secuenciación) o **matriz de expresión**. En el primer caso, las lecturas estarán acompañadas de información pertienente al protocolo de secuenciación utilizado para su obtención, en cambio, en el segundo caso, se especificará el conjunto de herramientas que se han utilizado para

generar la **matriz de expresión**.

Por lo general, las preguntas biológicas que se desean responder son generales e involucran a una población de individuos. Luego, los experimentos que se realizan para poder responderlas, utilizan muestras de esa población para luego, a partir de los resultados observados, inferir acerca del comportamiento de toda la población. Es así que la determinación de la ***expresión diferencial*** (DE) involucrará herramientas estadísticas que permitan afirmar o refutar las hipótesis con cierto valor de significancia, en el contexto de las etapas de *modelado* (Sección 1.3.3) y *evaluación* (Sección 1.3.4) del KDD. Por ende, cuando se habla de DE, ésta alude a la existencia de diferencias significativas en los niveles de expresión observados en condiciones experimentales.

2.3.1. Tipos de cambios en la expresión

Por lo general, las comparaciones de niveles de expresión son a través de *fold changes* que no son más que cocientes de los valores de expresión. Por ejemplo, si se analiza un experimento del tipo caso-control, donde la expresión de un gen/isoforma en el control es C y su expresión en el caso de estudio es T, el *fold change* de dicho gen/isoforma se define como el cociente T/C.

Anteriormente se han descrito tres niveles de cuantificación de la expresión, luego tres tipos de cambios pueden esperarse cuando se comparan los valores de expresión observados en dos condiciones experimentales. Por un lado, se presenta una situación de ***expresión diferencial de un gen*** (DGE) cuando existen diferencias significativas en los niveles de expresión del gen entre las dos condiciones comparadas. Por otro lado, pueden presentarse cambios a nivel de las isoformas (Figura 2.12). La situación en la cual las isoformas cambian su expresión absoluta, conservando o no las proporciones, se denomina ***expresión diferencial de isoformas*** (DIE). La Figura 2.12**A** ilustra este caso donde las tres isoformas de un gen han duplicado su expresión en la condición Tratamiento, respecto de la condición Control, sin modificación de las expresiones relativas. Ahora bien, cuando las proporciones de las isoformas cambian en una condición respecto de otra, como consecuencia de una alteración en el mecanismo de splicing, el fenómeno se denomina ***splicing diferencial*** (DS, de las siglas en inglés). Esta situación es ilustrada por la Figura 2.12**B** donde se aprecia que la única que ha cambiado su expresión en la condición Tratamiento es la Isoforma I. Puede notarse que los valores de expresión de las isoformas en la condición Control son los previamente

ilustrados en la Figura 2.11**C**. Luego, los porcentajes de expresión de las Isoforma I, Isoforma II e IsoformaIII son 34 %, 10 % y 56 %, respectivamente. Si ahora se calculan los porcentajes correspondientes para la condición Tratamiento, es posible encontrar que éstos han cambiado a 50,7 %, 7,5 % y 41,8 %, respectivamente. Dado que el SA afecta a todo el gen, se hablará de genes con DS.

Los cambios en la expresión previamente descritos no son independientes unos de otros. Un cambio del tipo DIE como el ilustrado en la Figura 2.12**A** claramente implica que exista expresión diferencial a nivel de gen ya que el gen ha pasado de expresar 150 transcritos en la condición Control a expresar 300 en el Tratamiento. No obstante, como no hay cambios en las proporciones, no hay DS. Si ahora se supone que la expresión de la Isoforma III en el Tratamiento se hubiese mantenido igual a la observada en el Control, se estaría en presencia de una combinación DIE-DS ya que las Isoformas I y II exhibirían DIE y el gen tendría cambios en el SA. La existencia o no de DGE deberá además ser determinada. En el otro ejemplo presentado, donde se ilustra el DS (Figura 2.12**B**), la Isoforma II parece ser la única que no presenta DIE pero en conjunto el gen no tiene DGE. A través de estos ejemplos es posible visualizar que las modificaciones en el transcriptoma ocurren en tres niveles, por lo que analizar todos ellos resulta fundamental para poder comprender tales alteraciones y relacionarlas con las condiciones experimentales.

2.3.2. Herramientas para análisis de expresión diferencial

El análisis de DE se lleva a cabo mediante el uso de herramientas estadístico-computacionales las cuales, por lo general, se enfocan en uno de los tres tipos de cambios descritos: DGE, DIE y DS. Los valores de expresión a nivel de genes son usados para inferir DGE, entretanto, la expresión de las isoformas se utilizan para estudiar DIE (Oshlack et al., 2010). El DS se estudia de diversas formas, aunque la mayoría de las herramientas utilizan la expresión de los exones para determinarlo, esto se basa en que si un exón se usó diferencialmente es como consecuencia de un cambio en el mecanismo de SA. Si bien este enfoque es el más utilizado para DS, no provee información acerca de qué isoformas son las que han estado involucradas en dicho cambio, lo cual es relevante para inferir el impacto del DS en términos biológicos (Soneson et al., 2016).

Una de las grandes **limitaciones** con las que se afrontan las herramientas de análisis de DE es que el número de genes/isoformas/exones explorados es varios órdenes mayor que el número de muestras analizadas. Como ejemplo, en el

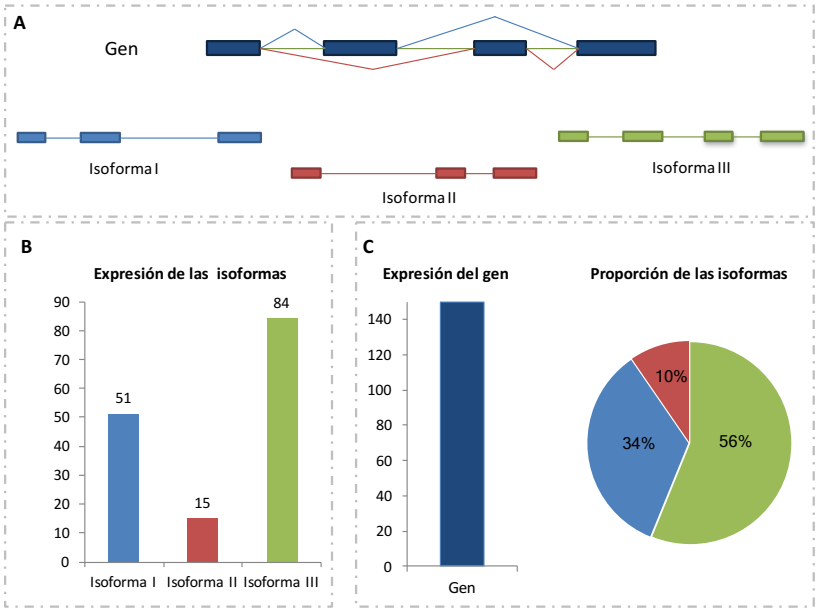

Figura 2.11: Ilustración del nivel de expresión del gen, de las isoformas y la proporción de las isoformas.

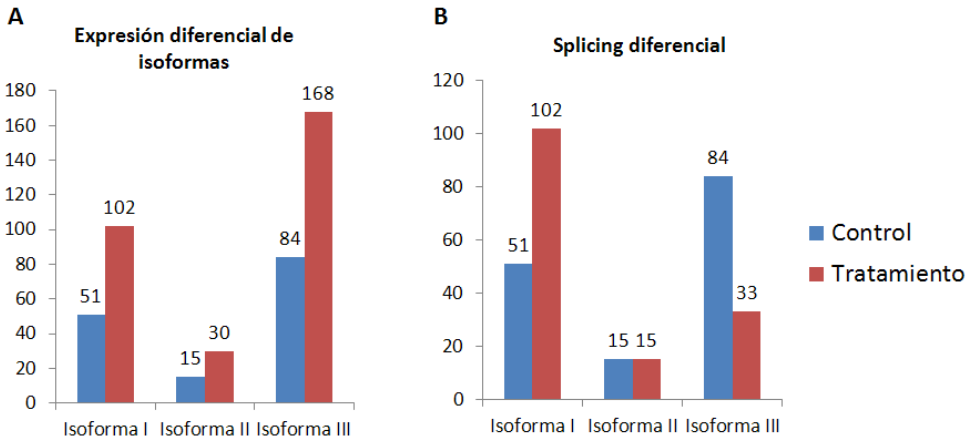

Figura 2.12: Ilustración de los cambios en la expresión a nivel de las isoformas.

caso de los estudios sobre humanos, una matriz de expresión contará con aproximadamente 20.000 filas si contiene expresión de genes, 130.000 filas si lo que se cuantificó son isoformas y más de 500.000 si se han explorado exones. En cuanto al número de muestras a analizar, en el ámbito de un laboratorio pequeño se suelen considerar, en algunos casos, tres réplicas por condición aunque en laboratorios más grandes se pueden llegar a las decenas de ellas (10, 20). En el mejor de los casos, los trabajos de consorcios enmarcados en grandes proyectos, como el *TCGA*, pueden alcanzar las centenas de muestras por condición, lo que aún dista bastante del número de regiones exploradas. Luego, modelar los cambios de expresión considerando las relaciones entre genes se torna prácticamente imposible. Por lo tanto, los enfoques más comunes utilizados para el análisis son del tipo univariados, donde cada gen/isoforma/exón es analizado de forma independiente, con el fin de poder estimar los parámetros de un modelo estadístico. La hipótesis nula del test que evalúa DE es que la expresión del gen/isoforma/exón no se altera con cambios en la covariable que explica las condiciones experimentales bajo estudio (Fang et al., 2012). En particular, en el análisis de DS, si bien cada exón se analiza individualmente, se suele agregar como covariable el total de conteos de los exones del mismo gen para así incorporar la correlación existente entre ellos. Adicionalmente, dado que se realizan decenas o centenas de miles de pruebas, es necesario realizar correcciones en los *valores p* para así reducir el número de falsos positivos. De todos los métodos desarrollados para tal fin, el *FDR* de Benjamini-Hochberg (Benjamini and Hochberg, 1995) es el más utilizado por las herramientas de análisis de DE.

Si bien existen algunas herramientas implementadas en otros lenguajes, el programa estadístico `R` (R Core Team, 2017) es el más utilizado para el análisis de expresión diferencial. Los modelos estadísticos pueden ser paramétricos o no paramétricos, siendo los primeros los más utilizados. Dado que los niveles de expresión generados a partir de los experimentos de RNA-seq se encuentran expresados en conteos (números enteros), las distribuciones del tipo Poisson fueron las que primeramente se utilizaron. Posteriormente se encontró que en realidad los datos no se ajustaban tan bien a esta distribución en donde la media y la varianza son iguales, sino que presentaban sobre-dispersión. Modelos lineales generalizados (GLMs, McCullagh, 1984) basados en la distribución Poisson con sobre-dispersión y en la distribución Binomial Negativa (NB) fueron la alternativa elegida para corregir estos errores. Actualmente, los modelos paramétricos basados en distribu-

ciones discretas asumen que los conteos siguen dicha distribución (Anders et al., 2012; Leng et al., 2013; Love et al., 2014; Robinson et al., 2010). Ejemplos de herramientas basadas en la distribución NB son los paquetes `R edgeR` y `DESeq2`, los cuales han sido diseñados para DGE aunque también han sido utilizados para DIE y el paquete `DEXSeq` para análisis de DS. Alternativamente, hay herramientas entre las cuales se destaca el paquete `R Limma`, que transforman los datos de conteos a escalas continuas para así poder utilizar modelos heterocedásticos basados en la distribución normal (Ritchie et al., 2015). Cualquiera sea el caso, no siempre se cumplen los supuestos del modelo, los cuales difícilmente pueden ser evaluados dada la cantidad de modelos que deben ajustarse. Para prescindir de la imposición de una distribución a los datos de expresión se han diseñado algunas herramientas con enfoques no paramétricos basadas en técnicas de remuestreo, como por ejemplo `NOISeq`, útil para evaluar DGE y DIE (Shi et al., 2015; Tarazona et al., 2015).

El resultado del análisis de DE producido por las herramientas mencionadas es una lista de genes con DGE, isoformas con DIE o genes con DS. Si el interés es conocer qué sucede en los tres niveles de información, es necesario hacer los tres análisis y luego combinar dicha información. Desde el surgimiento de la técnica de RNA-seq se han desarrollado diversas herramientas, aunque el mayor foco ha estado sobre el análisis de DGE. La gran similitud que presentan varias de las isoformas de un gen han complejizado el proceso de cuantificación de su expresión, por lo que el análisis a este nivel de información ha sido menos explorado. Consecuentemente, el desarrollo de nuevas metodologías fundamentalmente dirigidas al análisis de cambios en la expresión de las isoformas producidas por procesos post-transcripcionales como el SA es un requerimiento constante (Conesa et al., 2016).

Todos los conceptos desarrollados en este capítulo responden a las distintas etapas del proceso de KDD. En los capítulos siguientes se presentan las **estrategias y métodos desarrollados**, a los efectos de proporcionar *herramientas* que permitan un *análisis estructurado* de los datos transcriptómicos con el fin de contribuir al estudio y comprensión de las modificaciones post-transcripcionales del ARN.

Esta tesis se inició con una exploración bibliográfica profunda, como parte del desarrollo del primer objetivo del proceso de KDD: **entendimiento del problema**. Cabe destacar que el grupo de trabajo donde se desarrolló esta tesis no

había indagado hasta el momento en la temática que ésta involucra, por lo que esta etapa resultó fundamental. Como punto de partida, el Capítulo 3 presenta el *flujo estructurado* de análisis de DE de genes diseñado, utilizando herramientas disponibles (Merino et al., 2013). En particular, este flujo está orientado a la obtención de datos de elevada calidad que garanticen óptimos resultados (Merino et al., 2016). En el contexto de este flujo de trabajo, se desarrolló `TarSeqQC`, una herramienta para control de calidad y exploración de los datos (Merino et al., 2017b). En el contexto del KDD, esta contribución se focaliza en la etapa **entendimiento de los datos**. Como contribución a la etapa de **modelado**, el Capítulo 4 presenta un estudió comparativo de las metodologías existentes para el análisis de DS y DIE. Como resultado, se presenta una guía que orienta en la selección del conjunto de herramientas óptimos para cada estudio (Merino et al., 2017a). Finalmente, el Capítulo 5 presenta `NBSplice`, una herramienta de **modelado** desarrollada para determinar la ocurrencia de DS en experimentos de RNA-seq utilizando la información de expresión de isoformas, hasta ahora prácticamente ignorado. En los dos aportes anteriores, se diseñó una estrategia de **evaluación** basada en simulaciones con el objetivo de determinar el desempeño tanto de las herramientas existentes como de la herramienta desarrollada.

Capítulo 3

Exploración y control de calidad de los datos

3.1. Introducción

Si bien las NGS se comenzaron a comercializar en el año 2.005, las primeras publicaciones de trabajos científicos involucrando su utilización datan del año 2.008. Por ello, al momento de comenzar esta tesis, en el año 2.013, las NGS y sus aplicaciones estaban en pleno crecimiento. Del mismo modo, los investigadores y desarrolladores de herramientas de análisis también se encontraban en plena formación. Es así que la primera necesidad que se identificó al querer llevar a cabo un análisis transcriptómico fue la de contar con un **flujo de trabajo ordenado** que establezca cómo se debe realizar el análisis y con qué herramientas. La revisión bibliográfica realizada en ese momento develó que los análisis más frecuentes y sencillos eran los que analizan DGE. Por lo general, el flujo de trabajo común que sugerían los trabajos de la fecha es el ilustrado por la Figura 3.1 (Chu and Corey, 2012; Wolf, 2013). El análisis bioinformático toma como punto de partida las *lecturas de secuenciación*. El primer procesamiento involucra una etapa de *control de calidad* de dichas lecturas, donde se determinará su consistencia e integridad. Posteriormente, se lleva a cabo el *alineamiento contra referencia*, seguido de la *cuantificación*, en este caso, de los niveles de expresión de los genes. Finalmente, se realiza el *análisis de expresión diferencial*, para obtener una lista de genes cuya expresión puede haber sido afectada por las condiciones experimentales. El procesamiento puede terminar allí o continuar un paso más, en lo que se conoce como *análisis funcional*. Éste consiste en inferir el efecto biológico de la expresión

diferencial, para lo cual se utilizan en bases de datos que relacionan genes con funciones biológicas.

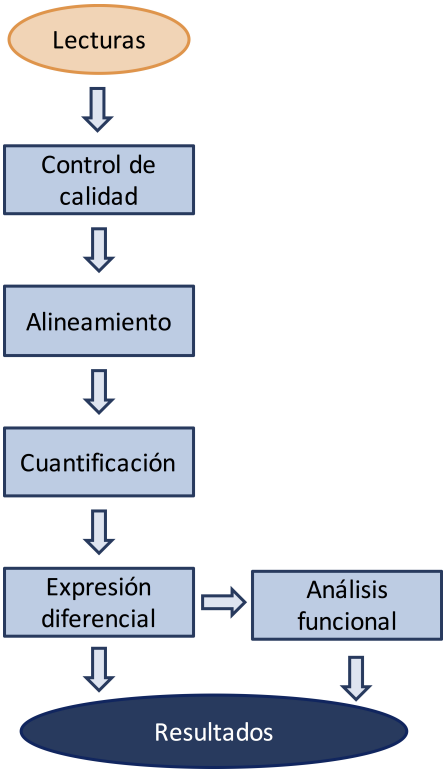

Figura 3.1: Flujo de análisis clásico en experimentos de RNA-seq.

En los primeros años de las NGS, las herramientas más utilizadas para llevar a cabo el análisis en datos de RNA-seq eran las llamadas *Tuxedo tools* (Trapnell et al., 2012, 2010). A saber, *TopHat* es el programa que realiza alineamiento, *Cufflinks* reconstruye transcritos y los cuantifica, obteniendo niveles de expresión de isoformas y genes y *Cuffdiff* realiza el análisis de expresión diferencial. A estos programas se les suma *FastQC*, útil para el control de calidad de las lecturas (Andrews, 2011). Estas herramientas se volvieron muy populares por su facilidad de uso ya que en tan sólo dos líneas de comando permiten obtener la lista de genes diferencialmente expresados entre dos condiciones experimentales. Otra característica que determinó el éxito de *Cuffdiff* es que no requiere de réplicas biológicas para realizar las estimaciones (Trapnell et al., 2012). Adicionalmente, todas ellas están disponibles para su uso libre en *Galaxy* (Goecks et al., 2010), una plataforma de análisis popular de uso libre e interactivo, y han sido sugeridas como flujo de procesamiento de experimentos de RNA-seq (Afgan et al., 2012).

Tal y como refleja la Figura 3.1, la única etapa donde se realiza un **control de calidad** es a principios del procesamiento de las lecturas. Sin embargo, en el medio de todas las etapas del análisis se generan conjuntos de datos intermedios que no son explorados y evaluados en términos de calidad. Por ejemplo, para realizar el análisis con las herramientas *Tuxedo* solo se debe ejecutar *TopHat* y posteriormente *Cuffdiff*. Este último utilizará internamente a *Cufflinks* para hacer la cuantificación de transcritos y genes. *Cufflinks* provee los resultados de expresión en unidades de *RPKM/FPKM* (Reads/Fragments Per Kilobase per Million mapped reads), dependiendo de si se utilizó un protocolo single o paired-end, respectivamente (Mortazavi et al., 2008). Los RPKMs (FPKMs) de una secuencia se calculan según la Ecuación 3.1, donde n refiere a la cantidad de lecturas (fragmentos) mapeadas a dicha secuencia, l es la cantidad de pb que componen dicha secuencia y s refiere a la cantidad de lecturas totales generadas en la secuenciación de la muestra, es decir el tamaño de la librería.

$$RPKM(FPKM) = \frac{n}{\frac{l}{1000}\frac{s}{1000000}} \tag{3.1}$$

Esta unidad pretende reflejar la concentración molar de un transcrito/gen en la muestra biológica, con el fin de hacer comparables los datos de expresión entre y dentro de las muestras, controlando los sesgos por longitud de gen y por tamaño de librería. Consecuentemente, *Cufflinks* asume que esto se cumple, por lo que no propone ningún control de calidad de los datos de expresión. Como alternativa a las *Tuxedo tools* se han desarrollado herramientas que intentan solventar algunos de los inconvenientes asociadas a ellas. En particular, las más utilizadas son el programa `Python HTSeq` (Anders et al., 2015) para la cuantificación de genes y los paquetes estadísticos de `R`: `edgeR` (Robinson et al., 2010), `DESeq` (Anders and Huber, 2010), `DESeq2` (Love et al., 2014) y `Limma` (Ritchie et al., 2015), para el análisis de expresión diferencial.

Por otro lado, el control de calidad del experimento como un todo tampoco es evaluado. Una contaminación en la extracción del ARN/ADN, un error en la asignación de los *barcodes*, una falla en la rotulación o simplemente una mala selección de los individuos considerados como réplicas puede alterar totalmente los resultados de un experimento. Por ejemplo, supongamos que dos muestras correspondientes a diferentes condiciones experimentales (**A** y **B**) son rotuladas en forma cruzada, es decir la muestra correspondiente a **A** es rotulada como **B**

y la muestra bajo la condición **B** es rotulada como **A**. Luego, la variabilidad que existe entre las réplicas de una misma condición se confundirá con la variabilidad biológica existente entre las condiciones **A** y **B**; consecuentemente, el número de genes identificados como diferencialmente expresados entre las condiciones **A** y **B** será bajo. Al ignorar el error que se ha cometido en el etiquetado, se concluirá erróneamente sobre la comparación de las condiciones **A** y **B**.

A los inconvenientes mencionados se les suma el hecho del escaso desarrollo de herramientas de visualización que provean una rápida exploración del alineamiento de las lecturas a regiones genómicas de interés. Los visualizadores más populares son `IGV` (Robinson et al., 2011) e `IGB` (Nicol et al., 2009). Ambos son herramientas `Java` diseñadas para explorar genomas completos. Sin embargo, en el caso de los experimentos de RNA-seq éstos resultan demasiado complejos, ya que sólo interesa explorar el pequeño grupo de genes o isoformas que modifican su expresión en respuesta a condiciones experimentales. Otras aplicaciones de las NGS, como la secuenciación dirigida de ADN o *Targeted Sequencing* también se focalizan en un pequeño grupo de regiones genómicas, por lo que contar con herramientas que permitan su rápida exploración beneficiaría también al análisis de experimentos realizados con esta técnica.

Con el fin de suplir las necesidades identificadas se *desarrolló* un **flujo estándar operativo** para el análisis de datos de RNA-seq dirigido a determinar qué genes evidencian DGE con el objetivo de **disminuir o eliminar sesgos** propios de los datos y así **optimizar** la calidad de los resultados obtenidos. El análisis propuesto se basó en técnicas del análisis de datos multivariados. Las herramientas utilizadas son, en su mayoría, programas existentes que se caracterizan por su versatilidad y por haber sido ampliamente utilizados (Merino et al., 2013, 2016). Adicionalmente, se desarrolló un paquete `R` llamado `TarSeqQC` para facilitar la exploración y el control de calidad en grupos reducidos de regiones genómicas (Merino et al., 2017b).

3.2. Métodos

3.2.1. Flujo de análisis

El *protocolo de operación estándar* diseñado con el fin de optimizar la calidad de los resultados obtenidos mediante el análisis de datos de RNA-seq fue la pri-

mera tarea realizada en el contexto de la investigación doctoral (Merino et al., 2013). El protocolo diseñado estableció un procedimiento de análisis basado en herramientas existentes, sentando una base para el desarrollo de trabajos futuros. Este protocolo está dirigido a:

- Asegurar la calidad de los alineamientos: Esto se logra mediante la evaluación de los valores de calidad disponibles en los archivos SAM.
- Evaluar la consistencia e integridad de los datos de expresión: Se debe asegurar que los valores se correspondan con números no negativos, que todos los genes/isoformas se hayan detectado en todas las muestras, que las escalas de expresión de las distintas muestras sean comparables, etc.
- Determinar la fiabilidad de las réplicas biológicas: En la mayoría de los análisis de RNA-seq, sólo un grupo de genes/isoformas son afectados por las condiciones experimentales. Por lo tanto, las diferencias entre éstas deben estar explicadas por los valores de expresión. Luego, un análisis multivariado puede ser utilizado para evaluar el comportamiento de las réplicas en términos globales.
- Corregir los sesgos propios de los datos de RNA-seq: Efecto de longitud de gen, de nivel de expresión y las diferencias entre los tamaños de librería deben ser controlados y, si es necesario, corregidos mediante estrategias de normalización adecuadas.

Siguiendo los objetivos previamente descritos, el flujo desarrollado propone:

1. **Alineamiento:** Mapeo de las lecturas de secuenciación de cada muestra contra el genoma de referencia, utilizando herramientas como `TopHat2` (Kim et al., 2013) o `STAR` (Dobin et al., 2013).
2. **Control de calidad y preparación de los datos:** Esta etapa se enfoca en el filtrado de genes. Se busca evaluar y remover los alineamientos de baja calidad, genes con baja expresión y que no son de interés del experimento. Las sub-etapas involucradas son:
   a) *Filtrado de lecturas:* Las lecturas con valor de *MAPQ* bajos y/o que mapean a más de una región genómica pueden resultar en estimaciones erróneas de los valores de expresión. Por este motivo, se propone

sólo conservar las lecturas con valor de *MAPQ* **mayor** a cierto umbral y mapeadas a una única región genómica. Un umbral aceptable de *MAPQ* puede ser 20, que significa que la probabilidad de que el alineamiento sea incorrecto es 0,01 (Cai et al., 2015; Kalari et al., 2012; Merino et al., 2016).

b) *Cuantificación de la expresión:* Las lecturas ya mapeadas deben ser asignadas a genes/isoformas. La elección de la herramienta de cuantificación muchas veces depende del programa que se utilice para analizar DE. Una vez obtenidos los perfiles de expresión de cada muestra, se utiliza `R` para construir la **matriz de expresión**, según se definió en la Sección 2.2.5. Es necesario destacar que en el caso de experimentos cuya matriz de expresión ha sido obtenida de un repositorio público, será fundamental controlar que los valores de expresión de todos los genes sean valores correctos, lo cual se puede hacer desde `R`.

c) *Filtrado de genes:* En el análisis de datos es necesario identificar y eliminar los datos ruidosos o poco confiables. En el contexto de RNA-seq, un dato ruidoso puede ser un gen con valor de expresión muy bajo en comparación con el resto de los genes, o incluso, con valor de expresión incorrecto. Por otro lado, el análisis transcriptómico intenta, en muchos casos, develar los cambios que ocurren en genes que *codifican a proteínas*, de modo que el resto de los genes o regiones genómicas anotadas constituyen una fuente de ruido ya que escapan del interés del estudio. La definición de *bajo valor de expresión* es relativa ya que depende directamente de la cantidad de lecturas se hayan generado en cada muestra. Para independizarse del tamaño de librería, esta definición se hace en términos de *conteos por millón* (CPM), definidos según lo indica la Ecuación 3.2, donde n indica los conteos de ese gen y s el tamaño de la librería, es decir, la suma de los conteos de todos los genes presentes en la muestra.

$$CPM = \frac{n \times 1000000}{s} \tag{3.2}$$

Diversos umbrales de CPM y criterios de baja expresión se han utilizado. Por ejemplo, se puede considerar que un gen tiene baja expresión si obtuvo, en *promedio*, menos de 1 CPM por condición; otro caso, sería

considerar baja expresión a menos de 1 CPM en todas las réplicas de una condición. La definición de genes de interés, en el caso del genoma humano, se sugiere hacer teniendo en cuenta los genes que tienen anotación funcional.

3. **Control de calidad global:** Una vez que se tiene la matriz de expresión para los genes expresados y que son de interés del estudio, es necesario controlar el comportamiento global de las muestras en el contexto del experimento.

   a) *Exploración de las distribuciones por muestras:* En general, la mayoría de los genes no modificarán su expresión en respuesta a un cambio en la condición experimental. Por lo tanto, las distribuciones de los valores de expresión de los genes en las muestras bajo estudio deben ser similares. Si esto no ocurre, será necesario un proceso de *normalización*. En el contexto de RNA-seq, uno de los sesgos más frecuentes que afectan a la comparabilidad de las muestras es la cantidad de lecturas generadas en cada una de ellas. Por lo general, este sesgo se corrige mediante la utilización de factores de escala obtenidos con funciones especificas implementadas en los paquetes `R` de análisis de expresión diferencial.

   b) *Separabilidad de las muestras:* Las muestras globalmente deben evidenciar la separación que existe entre réplicas de distintas condiciones. Con el fin de evaluar esta separabilidad y, en el caso que corresponda, determinar la presencia de *muestras atípicas o outliers* se propone la utilización del *PCA*. Éste permite explorar la variabilidad de las muestras mediante la definición de componentes ocultas de la matriz de expresión. Dado que, por lo general, un número pequeño de genes alterará su expresión como consecuencia del experimento, es de esperar que éstos sean los responsables de la variabilidad de las muestras. Luego, el diagrama de dispersión de las primeras dos componentes debería mostrar las réplicas biológicas agrupadas, mientras que las diferentes condiciones experimentales deberían aparecer distanciadas. Las muestras que *interrumpan* esta separabilidad, o se alejen de las réplicas de la misma condición experimental que representa, serán consideradas como *muestras outliers* y deberán eliminarse.

4. **Análisis de expresión diferencial:** Una vez que la **matriz de expresión** se ha "limpiado" es posible llevar a cabo el análisis de expresión diferencial. Para ello, se recomienda la utilización de los paquetes `R edgeR`, `Limma` o `DESeq2`, los cuales permiten el análisis de diferentes configuraciones experimentales y han sido ampliamente utilizados (Arrieta et al., 2015; Bailey et al., 2016; Ciriello et al., 2015; Melé et al., 2015; Nikolayeva and Robinson, 2014; Pelish et al., 2015; Thota et al., 2014).

5. **Exploración de los genes diferencialmente expresados:** Los perfiles de las lecturas sobre los genes detectados como diferencialmente expresados, así como también la cantidad de lecturas obtenidas en cada una de las muestras, pueden ser fácilmente exploradas utilizando la herramienta desarrollada, `TarSeqQC`. Ésta también provee la posibilidad de analizar la existencia de sesgos en los valores de expresión, ocasionados principalmente por factores como longitud o contenido en GC (Merino et al., 2017b).

6. **Análisis Funcional:** En el caso de que corresponda, realizar el análisis funcional del experimento utilizando herramientas como los paquetes `R GOSeq` (Young et al., 2010), `RDAVIDWebService` (Fresno and Fernández, 2013), `MIGSA` (Juan Cruz Rodríguez and Fernández) o la plataforma `DAVID` (Dennis Jr et al., 2003; Huang et al., 2007).

El protocolo hace uso tanto de funciones escritas en código `R` como de programas existentes. En la Seccion A.2.1 del Anexo Digital se describe el archivo `sourcePipeline.R`, el cual contiene las funciones implementadas. El archivo `pipeline.R` de la Sección A.2.2 de dicho anexo contiene las instrucciones necesarias para utilizar dichas funciones. El funcionamiento de los programas específicos utilizados puede ser consultado en la documentación y artículos científicos correspondientes. En particular, la herramienta desarrollada en el contexto de esta tesis, `TarSeqQC`, será presentada en la sección siguiente.

3.2.2. TarSeqQC

Tanto en experimentos de RNA-seq como otras aplicaciones de las NGS, sucede muchas veces que el investigador desea focalizar su atención en la exploración de genes/isoformas o regiones específicas determinadas como relevantes, ya sea por el análisis de DE o por su asociación con el experimento en cuestión. En

este contexto, `TarSeqQC` (Merino et al., 2017b) es una paquete de funciones que permite el control de calidad y la exploración de un grupo de regiones genómicas específicas a partir de los datos de alineamiento. La herramienta también comprende estrategias de visualización con resoluciones a nivel de nucleótido, lo cual en muchas oportunidades es un requisito fundamental, por ejemplo, para poder determinar alteraciones en una base nucleotídica que está afectando directamente la expresión de una determinada proteína. `TarSeqQC` se encuentra disponible para la comunidad científica en Bioconductor (`http://bioconductor.org/packages/TarSeqQC/`) y posee más de 3.000 descargas según las estadísticas del repositorio (`http://bioconductor.org/packages/stats/bioc/TarSeqQC/`) desde su primera versión, en septiembre de 2.015.

Descripción

`TarSeqQC` es un paquete `R` que utiliza para su funcionamiento tres tipos de archivos: *BED*, *BAM* y *FASTA*. Los archivos en formato *BED* son del tipo tabular en los que cada renglón representan una región genómica de interés, llamada *feature*. La cantidad de columnas en este archivo puede variar, pero al menos debe tener las siguientes: "chr" (cromosoma), "start" (posición genómica correspondiente al inicio de la *feature*), "end" (posición genómica correspondiente al final de la *feature*), "name" (nombre de la *feature*) y "gene" (nombre del gen que contiene a la *feature*). Por ejemplo, supongamos que las *features* son exones. Luego, cada renglón corresponderá a un exón y al menos deberá contener el cromosoma donde éste se ubica, la posición sobre tal cromosoma donde comienza y donde termina el exón, un nombre que lo identifique como único entre el resto de las *features* y finalmente, el nombre del gen que lo contiene. El archivo *BAM* contendrá los alineamientos de las muestras que se desean explorar, entretanto, el genoma de referencia donde están definidas las *features* se presentará en un archivo *FASTA* (ver Figura 3.2**A**).

El paquete implementa dos clases, `TargetExperiment` y `TargetExperimentList`, cada una de las cuales posee diferentes atributos y métodos. La primera de ellas está dirigida a almacenar los datos de alineamiento y cantidad de lecturas obtenidas para las *features* en una muestra. Consecuentemente, sus métodos permiten el control de calidad y exploración a nivel de muestra. La clase `TargetExperimentList` es útil para resumir los resultados de un experimento que involucra más de una muestra (ver Figura 3.2**B**). Cuando se construye un

objeto `TargetExperiment`, a partir de los archivos que se proveen, es posible computar el *coverage* (cantidad promedio de lecturas para una *feature*) o la mediana de las lecturas obtenidas para una *feature*. Cualquiera de estas dos medidas es indicadora de la cantidad de veces que se ha secuenciado la *feature* en cuestión por lo que puede ser utilizada para identificar *features* de baja expresión o débilmente captadas por la secuenciación.

Características

La herramienta desarrollada permite explorar los siguientes aspectos (Figura 3.2**C**):

- *Resultados a nivel de nucleótido*: Se proveen herramientas para indagar los *perfiles de lecturas o expresión* con resolución de nucleótidos. Es decir, gráficos de barras que ilustran la cantidad de lecturas que se han obtenido para cada nucleótido de una región genómica. En estos gráficos se destaca la cantidad de lecturas contradicen a la referencia, lo que podría estar indicando la presencia de una variante nucleotídica. También es posible explorar los porcentajes observados de bases nucleotídicas, diferentes a las de referencia, en cada posición genómica.

- *Desempeño a nivel de feature*: Se logra a través de la inspección simultánea de todas las *features* a etapas tempranas del análisis, para así poder detectar regiones no secuenciadas y/o de baja expresión. Los métodos implementados permiten incorporar umbrales de expresión predefinidos por el usuario. Por ejemplo, se podría utilizar los umbrales

  - $(0, 1)$: *feature* no secuenciada
  - $[1, 100)$: *feature* de baja expresión/coverage
  - $[100, 500)$: *feature* de expresión/coverage media
  - $[500, Inf)$: *feature* de expresión/coverage alta

- *Desempeño a nivel de muestra*: Se evalúa utilizando métodos gráficos y analíticos que analizan el porcentaje de lecturas alineadas a las *features*, la distribución empírica del coverage/expresión promedio, la comparación de los valores de coverage/expresión entre *features*, etc.

- *Desempeño del experimento*: Se incluyen métodos que integran los resultados de todas las muestras involucradas en un experimento, para así poder evaluar su consistencia.

De esta manera, el control de calidad y la exploración, a diferentes niveles, pueden ser llevados a cabo utilizando métodos gráficos y analíticos que incorporan intervalos y/o umbrales de coverage/expresión definidos por el usuario para así asistir el análisis. La Figura 3.2**D** ilustra esquemáticamente algunos de los resultados gráficos que se pueden obtener utilizando `TarSeqQC`.

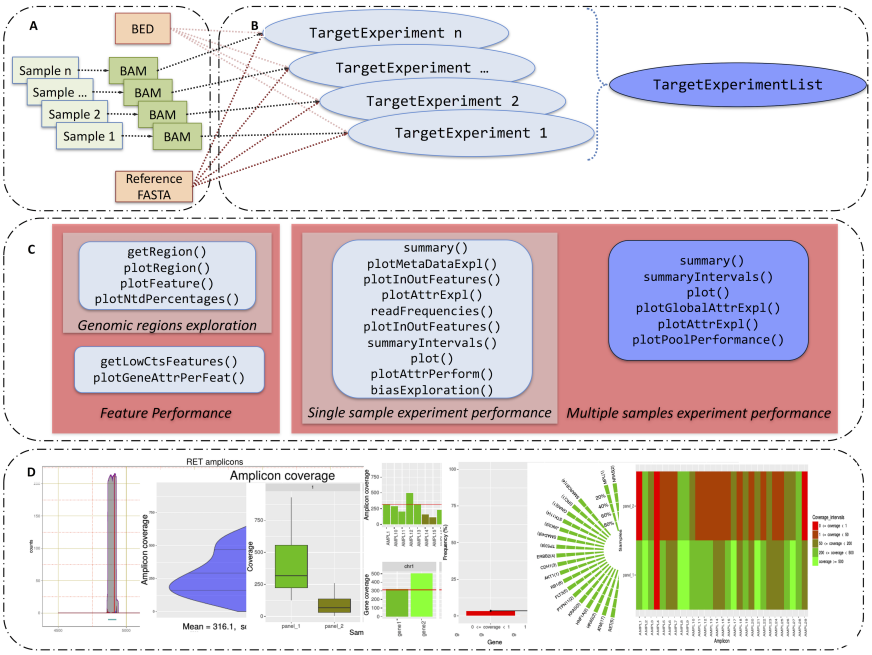

Figura 3.2: Esquema ilustrativo de `TarSeqQC`. **A**: Archivos *BAM*, *BED* y *FASTA* listos para usar. **B**: Creación de los objetos `TargetExperiment`, uno para cada muestra y posterior construcción del objeto `TargetExperimentList` resumiendo los resultados de todas las muestras del experimento. **C**: Uso de los métodos correspondientes a cada clase para explorar y visualizar los resultados a nivel de experimento, muestra y *feature*. **D**: Ejemplo de las gráficas generadas mediante el uso de `TarSeqQC`. Figura extraída de Merino et al. (2017b)

La principal limitación de `TarSeqQC` es que los recursos computacionales que requiere para poder llevar a cabo algunas de sus funcionalidades así como también los tiempos de ejecución dependen del tamaño de panel de *features* y de la cantidad de lecturas generadas en la secuenciación. La demanda de recursos es en especial alta para aquellas funciones que indagan a nivel de nucleótido, por ejem-

plo: `buildTargetExperiment`, `plotFeature` y `pileupCounts`. Como ejemplo, la Tabla 3.1 muestra los tiempos de ejecución al analizar una muestra típica de *Targeted Sequencing* (accesible en el repositorio SRA: SRR948511), conteniendo 2.866.654 lecturas pareadas, secuenciadas a partir de un panel de 1.736 *features* de 47 genes. El procesamiento se realizó en una computadora de escritorio equipada con un procesador Intel I5 con 8 Gb. de memoria RAM y dos procesadores físicos y cuatro hilos.

Tabla 3.1: Tiempos de ejecución de `TarSeqQC` sobre un conjunto de datos de 2.866.654 lecturas paired-end, obtenidas con un panel de 1.736 *features* de 47 genes.

Función	Núcleos usados	Tiempo de ejecución
Constructor `TargetExperiment`	4	1,43 min
Constructor `TargetExperiment`	2	1,38 min
`plotRegion` gen *JAK3* (20.000pb)∗	4	1,59 min
`plotRegion` gen *JAK3* (20.000pb)∗	2	2,78 min
`plotRegion` gen *JAK3* (10.000pb)∗	4	29,76 seg
`plotRegion` gen *JAK3* (10.000pb)∗	2	47,37 seg
`plotRegion` gen *JAK3* (5.000pb)∗	4	11,82 seg
`plotRegion` gen *JAK3* (5.000pb)∗	2	15,7 seg
`pileupCounts` gen *JAK3* (20.000pb)∗	4	1,55 min
`pileupCounts` gen *JAK3* (20.000pb)∗	2	2,75 min
`pileupCounts` gen *JAK3* (10.000pb)∗	4	28,44 seg
`pileupCounts` gen *JAK3* (10.000pb)∗	2	46,82 seg
`pileupCounts` gen *JAK3* (5.000pb)∗	4	9,95 seg
`pileupCounts` gen *JAK3* (5.000pb)∗	2	12,10 seg

∗Entre paréntesis se indica el número de pares de bases contenidos en las regiones exploradas.

3.3. Aplicación

3.3.1. Control de calidad multivariado en RNA-seq

A continuación se describe los resultados de la aplicación del flujo de trabajo propuesto en la Sección 3.2.1 sobre un conjunto real de datos de RNA-seq. Este trabajo surgió como una colaboración con el grupo de trabajo del Dr. Emmanuel Dias Neto, miembro de la Fundación Antonio Prudente y director del Laboratorio de Genómica Médica en el Centro Internacional de Investigaciones A.C. Camargo

Cancer Center, Brasil. El grupo del Dr Dias Neto realizó un experimento de secuenciación de RNA-seq, obteniendo un conjunto de datos que le fueron cedidos al Dr Fernández, Director de esta tesis, con el fin de realizar el análisis DEG. Cuando éste se realizó, siguiendo el enfoque clásico propuesto en la literatura y descrito en la Sección 3.1 se encontró que prácticamente no habían genes que evidenciaran DE, lo cual era poco probable por las condiciones experimentales involucradas. Es así que, con el fin de determinar el o los posibles sesgos, fuentes de errores o factores que se estaban ignorando se diseñó y utilizó el flujo de trabajo propuesto en esta tesis. Los resultados descritos en la siguientes secciones han sido presentados en el *XX Congreso Argentino de Bioingeniería*, realizado los días 28, 29 y 30 de octubre del año 2015 en San Nicolás de los Arroyos, Argentina. El manuscrito correspondiente ha sido publicado en el *Journal of Physics: Conference Series* (Merino et al., 2016).

El objetivo de este trabajo fue demostrar el efecto de la aplicación de los pasos de control de calidad propuestos y el impacto de incluirlos o no sobre los resultados finales del análisis.

Base de datos

La base de datos que se analiza se generó en el contexto de un proyecto dirigido al estudio de las alteraciones transcripcionales relacionadas con la ocurrencia de *metástasis* en pacientes con *cáncer de células escamosas orales* (por sus siglas del inglés *OSCC*). Para el experimento se consideró un total de 10 hombres con *OSCC*, cinco de los cuales presentaron *metástasis linfonodal*. Dado que el objetivo del estudio era determinar las alteraciones transcripcionales asociadas a la *metástasis*, esta condición se definió como el *tratamiento*, mientras que la ausencia de *metástasis* se consideró como condición *control*.

Los pacientes fueron reclutados en el Hospital A.C. Camargo en la ciudad de San Pablo, Brasil. A todos ellos se les extrajo una muestra biológica donde se aisló el ARNm para posteriormente realizar la secuenciación utilizando la plataforma *ABI5500 SOLiD* con un protocolo *paired-end* 75-35. Cabe aclarar que la definición 75-35 indica que la lectura que se realiza en sentido *hacia adelante* (o *forward*) tiene longitud de 75 pb, mientras, la que se realiza *hacia atrás* (o *reverse*) es de 35 pb. El diseño del experimento estuvo dirigido a eliminar cualquier efecto de corrida y/o *lane* sobre las muestras. Se realizaron dos corridas de secuenciación, cada una de las cuales involucró seis *lanes*. Los 10 pacientes se dividieron

en dos grupos de secuenciación, cada uno de los cuales tuvo representantes de las dos condiciones experimentales. De este modo, el experimento involucró dos condiciones experimentales, *sin metástasis* (Non-Met) y *con metástasis* (Met) y dos corridas de secuenciación, I y II, caracterizado por la matriz de diseño de la Ecuación 3.3.

$$MD = \begin{array}{c} \\ R1C1 \\ R2C1 \\ R3C1 \\ R4C1 \\ R5C1 \\ R1C2 \\ R2C2 \\ R3C2 \\ R4C2 \\ R5C2 \end{array} \begin{array}{c} \begin{array}{cc} Condicion & Corrida \end{array} \\ \left(\begin{array}{cc} Non-Met & II \\ Non-Met & I \\ Non-Met & II \\ Non-Met & II \\ Non-Met & I \\ Met & II \\ Met & I \\ Met & I \\ Met & II \\ Met & I \end{array} \right) \end{array} \tag{3.3}$$

En cada uno de estos grupos de secuenciación las muestras de sus integrantes se dividieron en seis alícuotas, una para cada *lane*. Es así que se utilizaron *barcodes* para identificar a las muestras de cada paciente. Los archivos de secuenciación de cada *lane*, para cada muestra, se alinearon contra el genoma humano (versión GRCh37/hg19 de *Ensembl*) utilizando el software provisto por *SOLiD*, llamado *LifeScope*. Posteriormente, los alineamientos de cada muestra se combinaron para obtener un archivo *BAM* por cada paciente. Estos archivos son los que se recibieron para realizar el estudio, por lo que constituyeron el punto de partida del análisis transcriptómico que aquí se presenta.

Estrategia de análisis

El tipo de análisis seleccionado para abordar el objetivo del estudio es la *expresión diferencial de genes*, previamente definida y nombrada como *DEG*. Para ello, se utilizó un *flujo de trabajo* basado en el protocolo de operación propuesto en la Sección 3.2.1. Las especificaciones correspondientes a cada etapa aquí utilizadas se detallan a continuación.

1. **Control de calidad y preparación de los datos**: se filtraron tanto alineamientos como genes según los criterios establecidos anteriormente.

a) *Filtrado de lecturas:* Las lecturas con *MAPQ* mayor a 20 y únicamente alineadas a una región genómica se conservaron mediante la utilización de la herramienta de cuantificación `HTSeq`.

b) *Construcción de la matriz de expresión:* El programa `HTSeq` se utilizó para cuantificar las lecturas a nivel de genes en cada una de las muestras. En particular, el modo de cuantificación utilizado fue aquel que considera solo las lecturas que intersectan completamente con un gen (modo `intersection-strict`). Luego se construyó la **matriz de expresión** en `R`.

c) *Filtrado de genes:* Se estableció como criterio conservar aquellos genes que tuvieron como mínimo 10 conteos en todas las muestra. Además, se filtraron los genes de longitud menor a 200 pb, ya que corresponden a ARNnp, y éstos no son objeto de estudio. También, como el interés sólo reside en los genes codificantes de proteínas, se eliminaron aquellos que no tuviesen anotación funcional en la plataforma *DAVID*.

2. **Control de calidad global**: se llevó a cabo en `R` utilizando funciones que éste provee.

   a) *Exploración de las distribuciones y normalización:* Los métodos `R boxplot` del paquete `graphics` y `density`, del paquete `stats`, se utilizaron para explorar los diagramas de cajas y estimar la función de densidad de los valores de expresión, respectivamente (R Core Team, 2017). Con el fin de seleccionar el método de normalización adecuado se compararon los provistos por `edgeR` y `DESeq2`.

   b) *Separabilidad de las muestras:* Se realizó el *PCA* de la **matriz de expresión** mediante la función `prcomp` del paquete `stats` de `R` (R Core Team, 2017).

3. **Análisis de expresión diferencial:** se llevó a cabo mediante el paquete `DESeq2` de `R`. Éste asume que los conteos que representan la expresión del i-ésimo gen en la k-ésima muestra, y_{ik}, siguen una distribución NB según la Ecuación 3.4, con media μ_{ik} y dispersión ϕ_i. En particular, `DESeq2` asume que la media es proporcional a la concentración de fragmentos de ADNc que había originalmente en la muestra, p_{ik}, escalada por el factor de nor-

malización s_k, asociado al tamaño de la librería de la k-ésima muestra.

$$y_{ik} \sim NB(\mu_{ik} = p_{ik}s_k, \phi_i) \tag{3.4}$$

Si bien ambas cantidades, p_{ik} y s_k son variables aleatorias, s_k es estimado a partir de las muestras. Luego `DESeq2` ajusta, para cada gen, un GLM con distribución NB según la Ecuación 3.5 con los elementos $x_k r$ de la matriz de diseño y los coeficientes β_{ir}.

$$log_2(q_{ik}) = \sum_r x_{kr}\beta_{ir} \tag{3.5}$$

En este trabajo, los elementos de la matriz de diseño (Ecuación 3.3) son dos, la condición y la corrida, por lo que $r = 1, 2$. Los coeficientes estimados del GLM indican el fold change, es decir el cambio en la expresión, entre dos niveles de dichos factores. Los fold changes estimados son luego corregidos mediante un procedimiento bayesiano (Love et al., 2014). Luego del ajuste de los modelos para todos los genes, esta herramienta utiliza la distancia de Cook para identificar datos atípicos, los cuales en su mayoría ocurren por efectos técnicos más que biológicos (Love et al., 2014). Los outliers son reemplazados por la media recortada al 20 % y los modelos son nuevamente ajustados. Posteriormente, se utiliza el test de Wald (Cordeiro and Demétrio, 2008) para evaluar si los coeficientes, o algún contraste, son nulos. En este caso, no se consideró interacción entre los efectos condición y corrida, ya que el diseño experimental cruzó estos dos efectos.

Resultados y discusión

Filtrado de datos

Los archivos de alineamiento correspondientes a las 10 muestras involucradas en el experimento bajo análisis consistieron, en promedio, de 150 millones de lecturas alineadas por muestra. Luego del filtrado por *MAPQ*, se obtuvo un promedio de 19 millones de pares de lecturas alineadas por individuo. Como resultado de la cuantificación se obtuvo una **matriz de expresión** de dimensiones n =54.664 (genes) y $p = 10$ (muestras). Esta matriz fue sometida a los filtrados descritos anteriormente identificando:

- 34.657 genes sin anotación funcional o no codificantes a proteínas
- 828 ARNnp
- 5.988 genes de baja expresión

De este modo, la **matriz de expresión**, CM_1, quedó conformada por 13.191 filas y 10 columnas.

Normalización

Las características distribucionales de los valores de expresión *crudos* se ilustran en la Figura 3.3**a-b**. En particular, los valores se han expresado en escala logarítmica en base 2. La inspección de los diagramas de caja indicó diferencias tanto en la mediana como en el ancho de las cajas de las distribuciones de las distintas muestras. En términos de densidad, se apreció que las muestras **no estaban centradas** en la misma moda. Luego, se requirió del uso de un método de normalización que permitió equiparar las diferencias encontradas entre las distintas muestras.

Con el fin de contar con un criterio **objetivo** que soporte la elección de un método de normalización sobre otro, se realizó una comparación de las estrategias disponibles focalizando en el impacto que éstas tienen sobre el número de genes diferencialmente expresados y el número de genes identificados como *outliers* por el método de DEG. Los métodos evaluados fueron *Trimmed Mean of M values* (TMM), *Upper Quartile* (UQ) y *Relative Log Normalization* (RLE) (Maza et al., 2013). Estos métodos se encuentran implementados en el paquete `edgeR` y comparten la hipótesis de que sólo un pequeño número de genes son los que alteran su expresión como consecuencia del cambio en la condición experimental. Adicionalmente, *RLE* también se encuentra disponible en el paquete `DESeq2`.

La evaluación se realizó normalizando la matriz CM_1 con las distintas estrategias. Luego se ajustaron los modelos con `DESeq2` y se determinó la cantidad de datos atípicos que se identificaron en cada una de las matrices normalizadas. Posteriormente, se reemplazaron los *outliers* y se reajustaron los modelos. Los resultados obtenidos se resumen en la Tabla 3.2, específicamente la segunda columna contiene los outliers detectados antes de la corrección y la tercera columna, los detectados luego de ella. Como se puede apreciar, la cantidad de outliers detectados tanto antes como después de la corrección resultó **menor** para la matriz CM_1 normalizada con la estrategia *RLE* de `edgeR`.

Tabla 3.2: Cantidad de outliers detectados antes y luego de la corrección establecida por DESeq2 sobre la matriz de expresión normalizada mediante diferentes métodos.

Método	*Outliers detectados*	*Outliers luego de la corrección*
TMM	700	89
Upper Quartile	688	89
RLE (edgeR)	689	57
RLE (DESeq2)	712	59

También se exploró el número de genes detectados por `DESeq2` como diferencialmente expresados en la comparación *Metastásico, No Metastásico*, utilizando la matriz de expresión normalizada mediante las diferentes estrategias. La Tabla 3.3 resume los resultados obtenidos, presentando la cantidad de: genes detectados con DE, genes sobre-expresados en la condición *Met*, genes sub-expresados en la misma condición respecto del *Non-met*. Los resultados revelaron que el número de genes con DE fue **mayor** cuando se utilizó la normalización *RLE* de `edgeR`, entretanto, la relación sobre/sub expresado fue similar para todas las metodologías.

Tabla 3.3: Cantidad de genes detectados como diferencialmente expresados por DESeq2 sobre la matriz de expresión normalizada mediante diferentes estrategias.

Método	*Genes con DE*	*Genes sobre-expresados*	*Genes sub-expresados*
TMM	212	99	113
Upper Quartile	199	91	108
RLE (edgeR)	207	93	114
RLE (DESeq2)	204	91	113

Los resultados anteriores sugirieron entonces la utilización de la estrategia de normalización *RLE* propuesta por `edgeR` para calcular los factores de escala. Luego de su aplicación sobre la matriz CM_1, las diferencias en distribución y en densidad se redujeron, como se aprecia en la Figura 3.3**c-d**.

Control de calidad global

Si bien, según los resultados anteriormente descritos, no se encontró evidencia de alguna muestra *outlier*, posteriormente se realizó el *PCA* sobre la matriz CM_1 *cruda* y normalizada. Los diagramas de dispersión de las primeras dos componentes de cada *PCA* se ilustran en la Figura 3.4, donde las réplicas de la condición

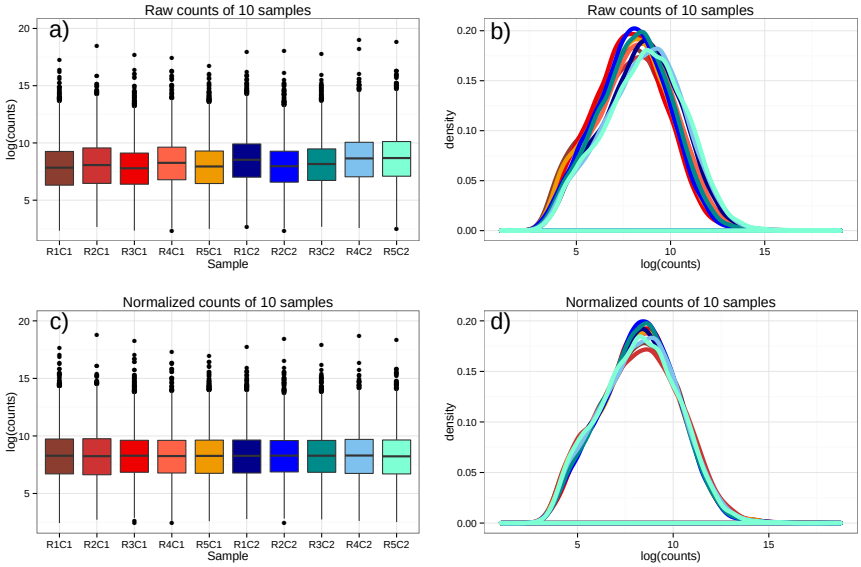

Figura 3.3: Diagramas de caja (izquierda) y gráfico de densidad estimada (derecha) para los valores de expresión. Los paneles **a)** y **b)** corresponden a los conteos sin normalizar y los paneles **c)** y **d)** luego de la normalización mediante la estrategia propuesta por `DESeq2`. La notación $RXCY$ refiere a: RX, réplica X; CY, condición Y donde $C1/C2$ es no-metástasis/metástasis, respectivamente. Figura extraída de Merino et al. (2016).

Met se identificaron con triángulos y las de la condición *Non-Met*, con círculos. Los diagramas obtenidos revelaron que las muestras no eran completamente separables entre condiciones, como se esperaba que sucediera. Específicamente, las flechas señalan una muestra de cada condición, identificadas como *R2C1* y *R2C2*, que se ubicaron más próximas a las réplicas de la condición opuesta. Este comportamiento indicó la **falta de separabilidad**, guiada por la condición experimental, en el espacio de las componentes principales. La normalización no corrigió la separabilidad de las muestras, lo que reveló que el efecto observado no es un efecto técnico sino que las dos muestras identificadas son **outliers**, por lo que debieron ser removidas del análisis.

La remoción de las muestras *outliers* implicó la re-ejecución de las etapas de análisis realizadas hasta el momento. Es así que la matriz de expresión original, de dimensiones 54.664 $\times$10 primero se redujo a una matriz de 54.664 $\times$8. Posteriormente, se realizó el filtrado de datos anteriormente aplicado para así obtener una nueva matriz de expresión, CM_2, conteniendo información de 13.482 genes identificados en 8 muestras. Cabe destacar que esta matriz tuvo un 2,2 % de **in-**

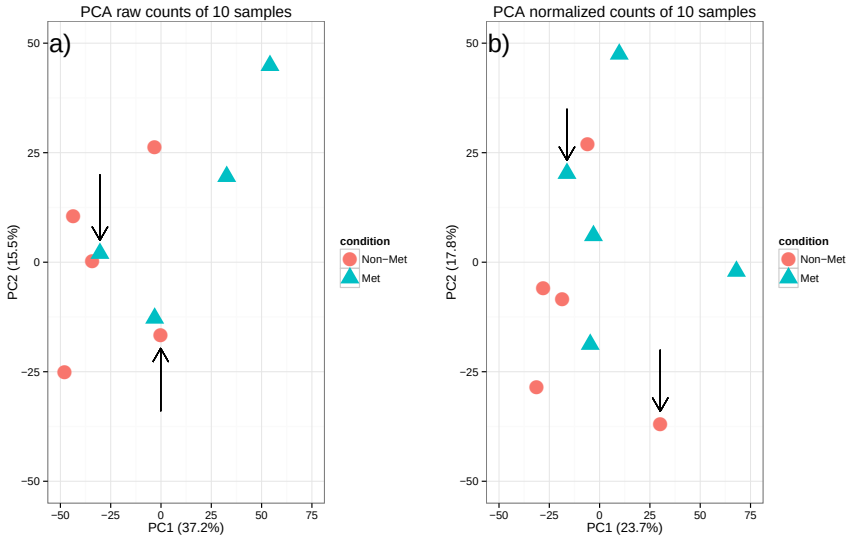

Figura 3.4: Diagramas de dispersión de las primeras dos componentes principales sobre la matriz CM_1: **a)** *cruda* o sin normalizar y **b)** normalizada. Figura extraída de Merino et al. (2016).

cremento en la cantidad de genes expresados contenidos en la matriz CM_1. En la Figura 3.5 se ilustran los diagramas de dispersión de las primeras dos componentes del *PCA* sobre la matriz CM_2 *cruda* (panel **a**) y normalizada (panel **b**). En ambos paneles se incorporó una línea entrecortada que indica la separabilidad que ahora sí se observa entre las muestras de las diferentes condiciones experimentales.

Impacto del control de calidad

Ambas matrices, CM_1 y CM_2, se analizaron con el fin de evaluar el impacto del control de calidad sobre el análisis. En primera instancia, se determinó el número de valores de expresión *outliers* detectados al analizar ambas matrices con `DESeq2`. Luego de la corrección de éstos, se determinó la DEG, considerando un nivel de significancia de 0,05 para el *FDR*. En la Tabla 3.4 se resumen los resultados obtenidos para las dos matrices de expresión. Por un lado, se ha encontrado que el número de *outliers* identificados usando la matriz CM_1 resultó **superior** al encontrado para la matriz CM_2. En el primer caso, el 6,35 % de los genes analizados presentaron valores atípicos, recuperando aproximadamente un 88 % de éstos mediante la estrategia de imputación utilizada por `DESeq2`. En el caso de la matriz CM_2, el 5,5 % de sus genes presentaron valores extremos. Luego de la corrección, se recuperó un 95 % de ellos, lo que representa un incremento del 7 %

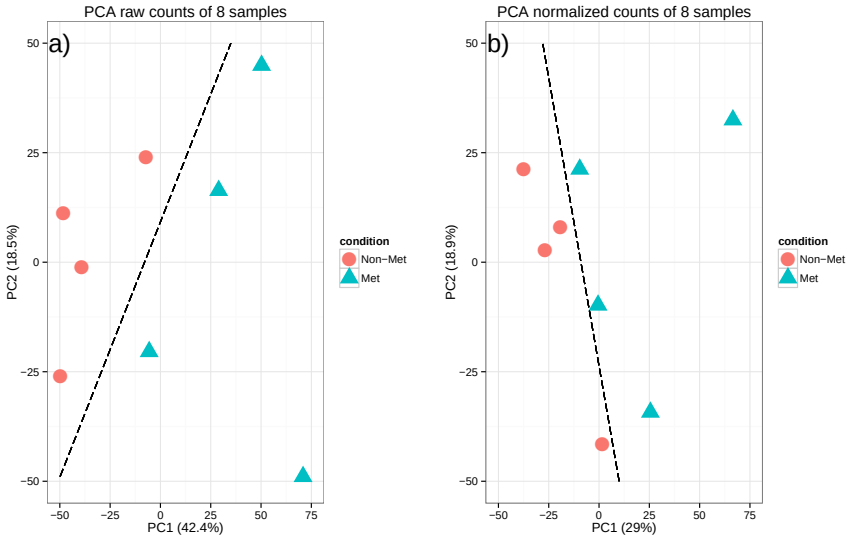

Figura 3.5: Diagramas de dispersión de las primeras dos componentes principales sobre la matriz CM_2: **a)** *cruda* o sin normalizar y **b)** normalizada. Figura extraída de Merino et al. (2016).

en términos de genes recuperados, revelando una **ganancia de información** al utilizar la matriz CM_2.

La última columna de la Tabla 3.4 contiene la cantidad de genes detectados como diferencialmente expresados. Es notable que la presencia de muestras *outliers* en la matriz CM_1 impactó directamente en el número de genes con DE ya que sólo 91 genes se identificaron como tales. La eliminación de las muestras *outliers* determinó un **incremento** de más del 300 % en la cantidad de genes detectados como diferencialmente expresados. Más precisamente, al analizar CM_2 se detectó 424 genes con DE. La comparación de los genes diferencialmente expresados encontrados en ambos casos determinó que el 88 % de los genes identificados en el primer caso también se encontraron al analizar CM_2. Los resultados obtenidos sugieren que la inclusión de las muestras *outliers* derivó en una pérdida de información que pudo ser recuperada al eliminarlas del análisis.

Finalmente, se compararon los fold changes y los valores p ajustados estimados para los 81 genes identificados como diferencialmente expresados al analizar las dos matrices. El análisis del diagrama de dispersión comparando los valores de *fold change* (Figura 3.6**a**) indicó que cuando se eliminaron las muestras *outliers* los valores absolutos de *fold change* se **incrementaron**. La prueba de Wilcoxon (Wilcoxon, 1945) reveló la existencia de evidencia significativa soportando este

Tabla 3.4: Resultados del análisis de expresión diferencial sobre las matrices de expresión CM_1 y CM_2.

Matriz de expresión	*Outliers detectados*	*Outliers post-corrección*	*Genes con DE*
CM_1	837	93	91
CM_2	737	38	424

incremento (valor p= $4{,}957e-7$) cuando se eliminaron las muestras *outliers*, tanto para los genes sobre-expresados (puntos rosa sobre la línea identidad) como para los genes sub-expresados (puntos celestes debajo de la línea identidad). Concordantemente, los valores p ajustados resultaron menores luego de la eliminación de las muestras atípicas (Figura 3.6**b**).

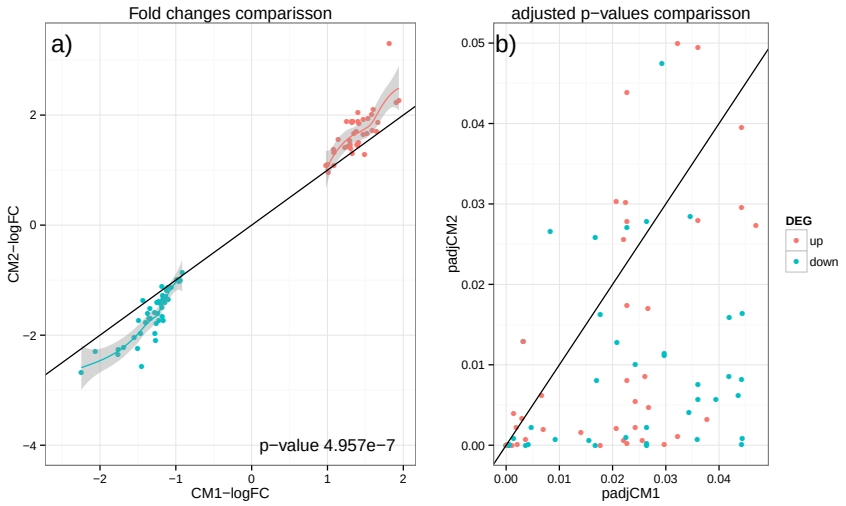

Figura 3.6: Diagramas de dispersión para los 81 genes detectados como diferencialmente expresados utilizando las dos matrices de expresión CM_1 y CM_2. **a)** Valores de fold change en escala logarítmica. **b)** Valores p ajustados por FDR. Figura extraída de Merino et al. (2016)

Conclusiones

La utilización de un *flujo de análisis estándar* focalizado en el *control de calidad* es fundamental en el estudio de experimentos que analizan los cambios de expresión de genes utilizando experimentos de RNA-seq. Con la aplicación aquí presentada se ha demostrado que el uso de herramientas del análisis multivariado, como el *PCA*, permitió la identificación de muestras atípicas en el conjunto de datos que estaban **enmascarando** información biológica relevante. La utilidad

del proceso de control de calidad global se demostró mediante el análisis de un experimento real de RNA-seq. Se encontró que, ignorando la posible existencia de muestras atípicas, sólo 91 genes pudieron ser detectados como diferencialmente expresados. En cambio, al incorporar la estrategia propuesta, este número se incrementó más de un 300 %. En este contexto, se ha demostrado que la presencia de muestras *outliers* impacta directamente en los análisis posteriores por lo que el **control de calidad global** debe ser un paso **fundamental** a realizar en cualquier flujo de análisis de datos de RNA-seq.

3.3.2. Control de calidad en Targeted Sequencing

Aquí se presenta una aplicación directa de la herramienta desarrollada, `TarSeqQC` sobre un conjunto real de datos de secuenciación dirigida o *targeted sequencing*. Este trabajo surgió como una colaboración con el grupo de trabajo del Dr. Osvaldo Podhajcer, Jefe del Laboratorio de Terapia Molecular y Celular de la Fundación Instituto Leloir, Argentina. La herramienta desarrollada y la aplicación que se detalla a continuación han sido publicadas en la revista *Human Mutation* de la editorial *Wiley* (Merino et al., 2017b).

Base de datos

El conjunto de datos con el que se trabajó en esta aplicación se generó mediante un protocolo de secuenciación dirigida o *targeted sequencing*. Éste permite explorar en forma simultánea un conjunto de regiones genómicas o *features* específicas (el *panel*) pertenecientes a un pequeño grupo de genes. Por lo general, la búsqueda se dirige a *features*, también llamadas amplicones, donde se conoce o se sospecha que pueden ocurrir variaciones genéticas asociadas a una enfermedad específica (Meldrum et al., 2011). El protocolo de secuenciación se inicia a partir de una muestra de ADN extraída, de la cual se seleccionan las *features* de interés utilizando *primers* diseñados específicamente para tal fin.

El experimento analizado tuvo como objetivo la caracterización del perfil molecular del cáncer colorrectal en un paciente con poliposis adenomatosa. Los trabajos de extracción y procesamiento de las muestras se llevaron a cabo en el Hospital Udaondo y en el Instituto Leloir de la ciudad de Buenos Aires, con consentimiento informado del paciente. Las muestras involucradas fueron una biopsia de tejido tumoral, una muestra de tejido sano y una muestra de ADN circulante

(o cell free DNA). Respectivamente, se las denotó como *Tumor*, *Normal* y *cfDNA*. En el contexto del experimento, la comparación de la composición del ADN de la muestra *Tumor* con la *Normal* permite determinar la presencia de variaciones asociadas a la patología; mientras que la comparación de la muestra *cfDNA* con las otras dos devela la existencia de posibles metástasis. Particularmente, un gen que se seleccionó para su secuenciación es el *APC*, un gen supresor de tumor en el que se han reportado numerosas variaciones genéticas asociadas a la patología bajo estudio (Nieuwenhuis and Vasen, 2007; Segditsas and Tomlinson, 2006).

La secuenciación de las muestras se realizó utilizando el kit comercial *Ion Ampliseq^{TM} Comprehensive Cancer panel* (`http://www.thermofisher.com/`) y el secuenciador *Ion Proton*. El *panel* utilizado se ha diseñado para secuenciar 15.991 *features* pertenecientes a 409 genes relacionados con algún tipo de cáncer. La selección de las *features* se realizó mediante la utilización de cuatro pools de primers de PCR. La secuenciación se realizó en dos corridas; en la primera se secuenció las muestras *Tumor* y *Normal*, a una cobertura promedio de 1.000X, y en la segunda, se secuenció la muestra *cfDNA* a una profundidad de 10.000X. Las lecturas de secuenciación se alinearon contra el genoma de referencia (hg19, GRCh37) con el software `TMAP4`, incluido en la plataforma de análisis (*Torrent Suite* verión 5.0) provista con el secuenciador. Los archivos *BAM* se procesaron con *Samtools* (versión 1.2, Li et al., 2009) para filtrar las lecturas no mapeadas y ordenar los alineamientos.

Estrategia de análisis

En el contexto de los experimentos de *targeted sequencing*, el control de calidad debe asegurar que todas las muestras involucradas en un experimento tengan cobertura promedio acorde con lo estipulado en su diseño y, fundamentalmente, suficiente para los análisis posteriores de descubrimiento de variantes genómicas. En esta aplicación, el control de calidad del experimento se realizó utilizando `TarSeqQC`. Para ello, se emplearon los archivos de alineamiento, el genoma de referencia y un archivo *BED*, conteniendo la definición de las 15.991 *features* contenidas en el panel. En primera instancia, se construyó un objeto de la clase `TargetExperiment` para cada una de las muestras analizadas y luego se resumieron los resultados de éstas en un objeto de la clase `TargetExperimentList`. Posteriormente, se utilizaron los métodos de ambas clases para llevar a cabo las tareas de control de calidad anteriormente descritas. El análisis comenzó con una

evaluación global del desempeño del experimento, en términos de eficiencia a la hora de leer las *features*. Se evaluó la cobertura promedio lograda en todas las *featuares* y por pool de primers. Para ello se definió un conjunto de intervalos de cobertura delimitados por los valores 0, 1, 100, 1.000, 5.000 y 10.000. Luego, se identificó las *features* de baja cobertura o desempeño y finalmente, se analizó el gen *APC*. Los archivos de alineamiento también se analizaron con `TEQC`, la única herramienta de control de calidad disponible para este tipo de experimentos hasta el momento (Hummel et al., 2011). Todos los archivos con el código fuente (del ingles, *scripts*) utilizados para realizar estos procesamientos se encuentras disponibles como material suplementario de Merino et al. (2017b).

Resultados y discusión

Desempeño del experimento

Como primer herramienta, se utilizó el método `summary` para obtener las medidas resumen de la cobertura a nivel de *features* y de *pool*. En el primer caso, se obtuvo un valor de cobertura promedio de 3.585 y 2.417 para las muestras *Tumor* y *Normal*, respectivamente, entretanto, para la muestra *cfDNA* se determinó un valor promedio de 4.219 (ver Tabla 3.5). Estos valores sugieren un buen **desempeño global** para las dos primeras muestras, por el contrario, en el caso de *cfDNA*, el valor de cobertura promedio resultó ser menor al esperado considerando que en su diseño se fijó un valor de cobertura de 10.000. Esta muestra también fue la que evidenció mayor variabilidad al comparar los coeficientes de variación encontrados para la cobertura. Adicionalmente, se determinó que en las tres muestras hubo al menos una *feature* que no se logró secuenciar. Similares resultados se obtuvieron mediante el análisis con `TEQC`. La exploración de la cobertura por *pool*, sólo posible con `TarSeqQC`, se hizo mediante la función `plotPoolPerformance`, la cual evidenció gran variabilidad de los valores de cobertura logrados en cada uno de ellos. Particularmente, el *pool 1* logró la cobertura promedio más alta, 5.201, mientras que los otros tres alcanzaron valores entre 2.500 y 3.000.

Las frecuencias absoluta y relativa y las correspondientes frecuencias acumuladas de *features* encontradas en los intervalos de cobertura definidos previamente se resumen en la Tabla 3.6. Se encontró que más del 64 % de las *features*, en las tres muestras, tuvieron cobertura **superior** a 1.000. En el caso de la muestra *Tumor*, más del 90 % de las *features* tuvieron cobertura mayor a este valor. Sólo

Tabla 3.5: Medidas resumen obtenidas para la cobertura lograda en las muestras *Tumor*, *Normal* y *cfDNA* con `TarSeqQC` y `TEQC`.

Muestra	*Tumor*		*Normal*		*cfDNA*	
Herramienta/ Medida	**TarSeqQC**	**TEQC**	**TarSeqQC**	**TEQC**	**TarSeqQC**	**TEQC**
Promedio	3.585	3.519	2.417	2.396	4.219	3.989
Desvío	1.759	1.901	2.296	2.409	5.331	5.413
Coeficiente de variación (%)	49,07	54,02	95	100,54	126,42	135,7
Mínimo	0	0	0	0	0	0
1º cuartil	2.437	2.278	412	162	719	226
Mediana	3.560	3.425	1.960	1.936	2.271	1.928
3º cuartil	4.686	4.588	3.768	3.769	5.852	5.831
Máximo	35.650	35.798	44.660	44.829	97.000	141.514

Tabla 3.6: Frecuencias absoluta y relativa y las correspondientes frecuencias acumuladas de *features* encontradas en los intervalos de cobertura definidos para las muestras *Tumor*, *Normal* y *cfDNA*.

Intervalo de cobertura	**Muestra**					
	Tumor		*Normal*		*cfDNA*	
	Abs	Rel	Abs	Rel	Abs	Rel
	(cum)	(cum)	(cum)	(cum)	(cum)	(cum)
[0;1)	8	0,1	829	5,2	130	0,8
	(8)	(0,1)	(829)	(5,2)	(130)	(0,8)
[1;100)	168	1,1	1.232	7,7	943	5,9
	(176)	(1,2)	(2.061)	(12,9)	(1.073)	(6,7)
[100; 1.000)	1.063	6,6	3.661	22,9	3.942	24,7
	(1.239)	(7,8)	(5.722)	(35.8)	(5.015)	(31,4)
[1.000; 5.000)	11.637	72,8	8.187	51,2	6.287	39,3
	(12.876)	(80,6)	(13.909)	(87)	(11.302)	(70,7)
[5.000; 10.000)	3.079	19,3	1.661	12,5	2.877	18
	(15.955)	(99,9)	(15.900)	(99,5)	(14.179)	(88,7)
[10.000; Inf)	36	0,2	91	0,5	1.812	11.3
	(15.991)	(100)	(15.991)	(100)	(15.991)	(100)

ocho *features* resultaron con **cobertura nula** en esta muestra, es decir, no se secuenciaron; las muestras *Normal* y *cfDNA* tuvieron 829 y 130 *features* no se-

cuenciadas, respectivamente. Tanto en la muestra *Normal* como en la *cfDNA*, se encontró un porcentaje cercano al 30 % de *features* con cobertura inferior a 1.000. Más aún, en la muestra *Normal*, casi un 13 % de las *features* exhibieron cobertura menor a 100, revelando un **bajo desempeño** de los primers utilizados para su captura y/o un problema en su secuenciación. En el caso de la muestra *cfDNA*, este porcentaje fue de 6,7 %. Estos resultados sugirieron una posible relación entre la variabilidad observada a nivel de pooles y las *features* de **bajo desempeño**. Cabe destacar que esta evidencia no fue encontrada con `TEQC` ya que esta herramienta no considera información de pool.

El bajo desempeño observado en un conjunto de *features* de las muestras *Normal* y *cfDNA* se verificó utilizando el método `plot`, el cual genera una gráfica bidimensional donde cada color está determinado por el valor de cada celda en dicha matriz. En la implementación de `TarSeqQC`, la matriz contiene *features* (1, ..., n) en filas, y muestras en columnas (1, ..., p), de manera que la celda_{ij} de la matriz tiene el intervalo de cobertura registrado para la i-ésima *feature* en la j-ésima muestra. La Figura 3.7**A** ilustra los resultados encontrados cuando los amplicones se ordenaron en la matriz según su posición genómica. A grandes rasgos, es de notar que en la muestra *Tumor* es donde prevalecen las celdas color verde, que indican cobertura mayor a 1.000. Cuando las *features* (amplicones) se agruparon según el pool de PCR (Figura 3.7**B**), al especificar el parámetro `pool` en el método `plot`, la gráfica se tornó mucho más clara, indicando que específicamente los pooles 2 de la muestra *Normal* y 3 de la muestra *cfDNA* agruparon las *features* de menor cobertura. Estos resultados indicaron la ocurrencia de **problemas técnicos** durante la preparación de la librería, lo cual derivó en **bajo desempeño** de algunos de los pooles de PCR involucrados en cada muestra.

El complemento cuantitativo de los resultados gráficos anteriores se obtuvo mediante la función `plotAttrPerform`. La gráfica generada por ésta se muestra en la Figura 3.7**C**. En ella, se ilustran las frecuencias relativas acumuladas de *features* en cada uno de los intervalos de cobertura y coloreadas por pool, para las muestras *Tumor* (panel **I**), *Normal* (panel **II**) y *cfDNA* (panel **III**). Idealmente, las líneas y puntos que identifican a cada pool deberían confundirse, como sucede en el caso de la muestra *Tumor*. Se destaca además que, en dicha muestra, las *features* se acumularon en los intervalos de cobertura alta. Por ejemplo, más del 70 % de las *features* en todos los pooles de esta muestra tuvieron cobertura mayor a 1.000. En el caso de las otras dos muestras, ambas evidenciaron el sesgo en uno de los

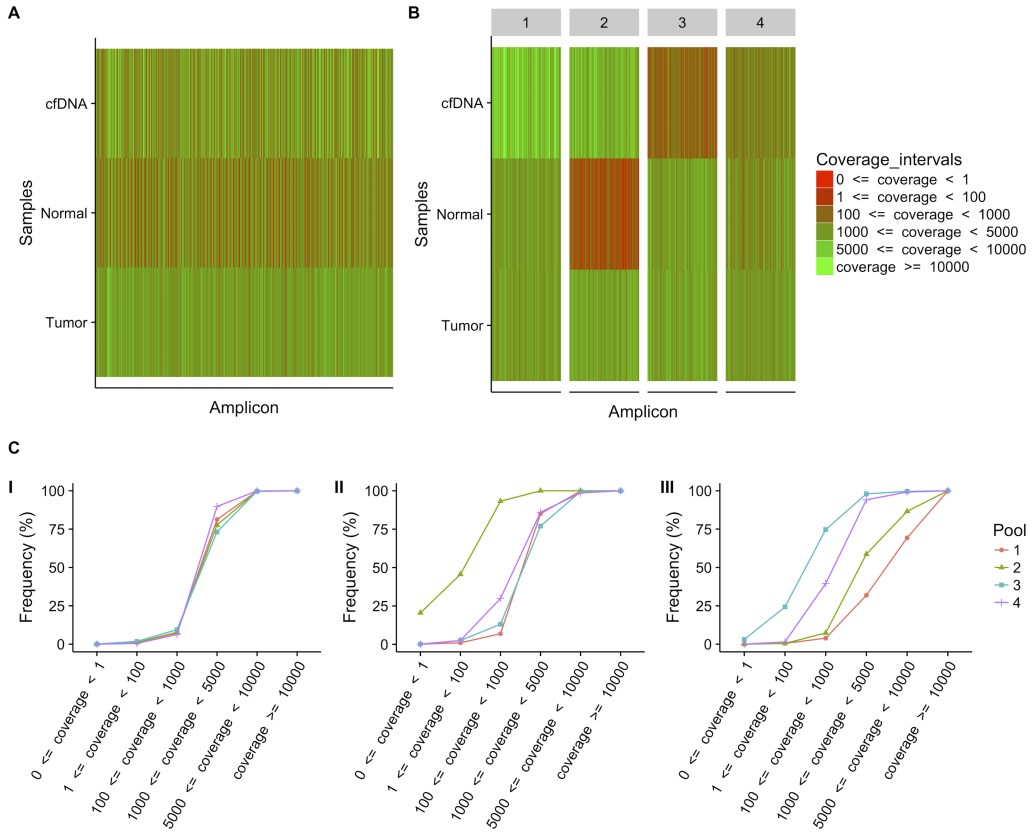

Figura 3.7: Resultados de cobertura por muestra y por pool, coloreados de acuerdo al intervalo de cobertura donde cae cada *feature*. **A**: Exploración conservando el orden genómico de las *features*. **B**: Exploración agrupando las *features* por pool de primers. **C**: Frecuencia relativa acumulada de *features* encontradas en cada intervalo de cobertura para las muestras **I**) *Tumor*, **II**) *Normal* y **III**) *cfDNA*. Figura extraída de Merino et al. (2017b).

pooles de PCR, observado previamente al analizar la Figura 3.7**B**. En el caso de la muestra *Normal*, se encontró que un 93 % de las *features* secuenciadas en el pool 2 tuvieron cobertura menor a 1.000 incluso, con el 45 % de éstas (1.824 *features*) con cobertura menor a 100. En la muestra *cfDNA* se encontró una situación más compleja, con dos pooles, el 3 y el 4, de **bajo desempeño**. Específicamente, el pool 3 fue el de peor desempeño, con el 75 % de sus *features* logrando cobertura menor a 1.000, mientras que para el pool 4 este porcentaje fue de 40 %.

Features de bajo desempeño

El análisis anterior reveló que las muestras analizadas involucraron un conjunto de *features* que no fueron secuenciadas a una profundidad suficiente, las cuales se llamaron *features* de bajo desempeño ya que ellas no podrán ser utilizadas **confiablemente** para detectar variantes genómicas. La exploración de la identidad de estas *features* se realizó mediante el método `getLowCtsFeatures`, el cual retorna aquéllas cuya cobertura ha sido inferior a cierto umbral, especificado por el usuario. En este caso, como la cobertura del diseño del experimento fue 1.000, se tomó como valor de baja cobertura a 100. Luego, se identificaron 3.037 *features*, provenientes de 396 genes, con cobertura menor a este umbral en al menos una de las tres muestras. Estas *features* fueron mayoritariamente del pool 2 (1.828) y del pool 3 (1.007). Específicamente, de las 1.828 *features* identificadas en el pool 2, 1.772 tuvieron baja cobertura sólo en la muestra *Normal*, mientras que 901 de las 1.007 *features* identificadas en el pool 3, sólo tuvieron bajo desempeño en la muestra *cfDNA*. Estos casos pueden explicarse, por ejemplo, por una **inhibición** a la técnica de PCR, diferente en cada una de las muestras. Adicionalmente, se determinó que 91 *features* tuvieron baja cobertura en todas las muestras, lo cual podría indicar una falla o deficiencia en el **diseño** de los primers de PCR utilizados para su lectura. Finalmente, se detectaron dos *features* que no se secuenciaron en ninguna muestra. Una de ellas es la de nombre *240390110* del gen *TAL1*, el cual tiene 12 *features* en el panel; y la otra es la *234444604*, la cual es parte del gen *MAP2K4*, que tiene otras 20 *features* en el panel.

Desempeño del gen APC

La exploración en detalle del gen *APC* reveló que sus 94 amplicones lograron una cobertura promedio superior a 1.900 en la muestra *Normal* y a 3.000 en las otras dos muestras analizadas. Específicamente, la cobertura promedio de cada *feature* de este gen se ha resumido en el gráfico de barras mostrado en la Figura 3.8 , obtenida con el método `plotGeneAttrPerFeat`. Se determinó que el

bajo desempeño encontrado en el pool 2 (barras verdes) de la muestra *Normal* (panel **B**), pool 3 (barras celestes) y pool 3 (barras violetas) de la muestra *cfDNA* (panel **C**) también estuvo presente en el gen *APC*. En el caso de esta última muestra, también se encontró que la mayoría de las *features* del pool 1 tuvieron cobertura muy alta (mayor a 5.000), en cambio, las del pool 3 y 4 fueron las de menor cobertura. A modo de ejemplo, la *feature 224537542* de este gen tuvo una cobertura superior a 4.500 en las muestras *Tumor* y *Normal* (círculos rojos en la Figura 3.8**A** y **B**), mientras que en la muestra *cfDNA* la cobertura fue inferior a 360 (circulo rojo en la Figura 3.8**C**). Similarmente, la última *feature* de los gráficos, destacada con círculos azules, evidenció cobertura mayor a 900 en las muestras *Tumor* y *cfDNA* y cercana a cero para la muestra *Normal*.

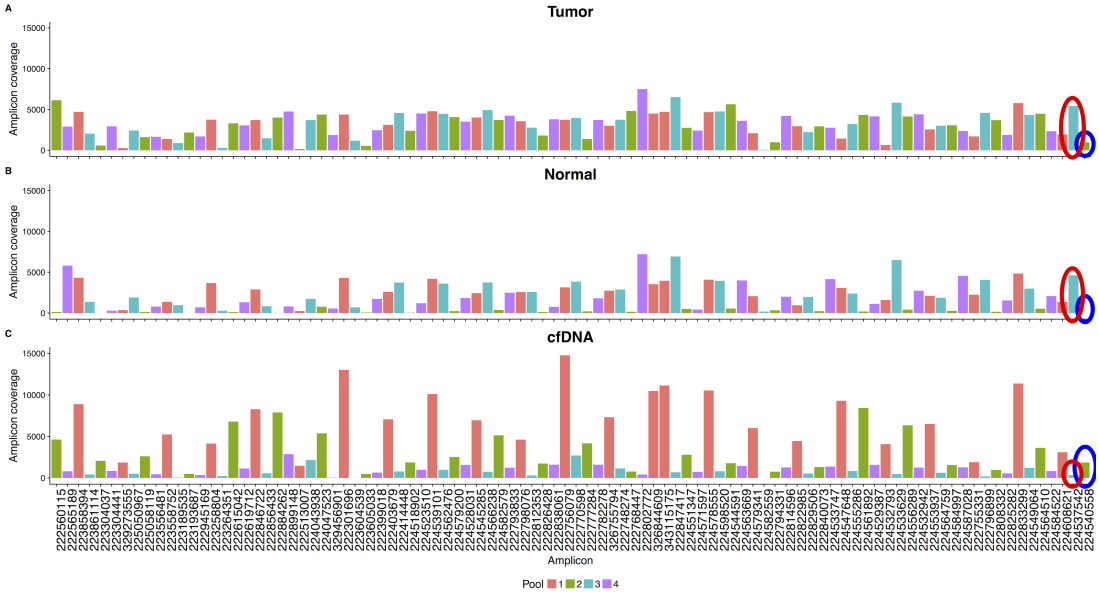

Figura 3.8: Cobertura por *feature* (amplicon) del gen *APC* en las muestras **A)** *Tumor*, **B)** *Normal* y **C)** *cfDNA*. El color de cada barra representa el pool donde fue secuenciado cada *feature*. Figura extraída de Merino et al. (2017b).

En particular, cuando se analizó el solapamiento entre *features* del gen *APC*, se encontró que la última *feature* no solapa con ninguna de las otras del panel, por lo que cualquier variación genética que exista en la muestra *cfDNA* dentro de ella **no podrá detectarse**.

Por último, el perfil de lecturas de la *feature 224537542* del gen *APC* se exploró con el método `plotFeature`. Las gráficas resultantes se muestran en la Figura 3.9. Coincidentemente con lo encontrado anteriormente, se aprecia que

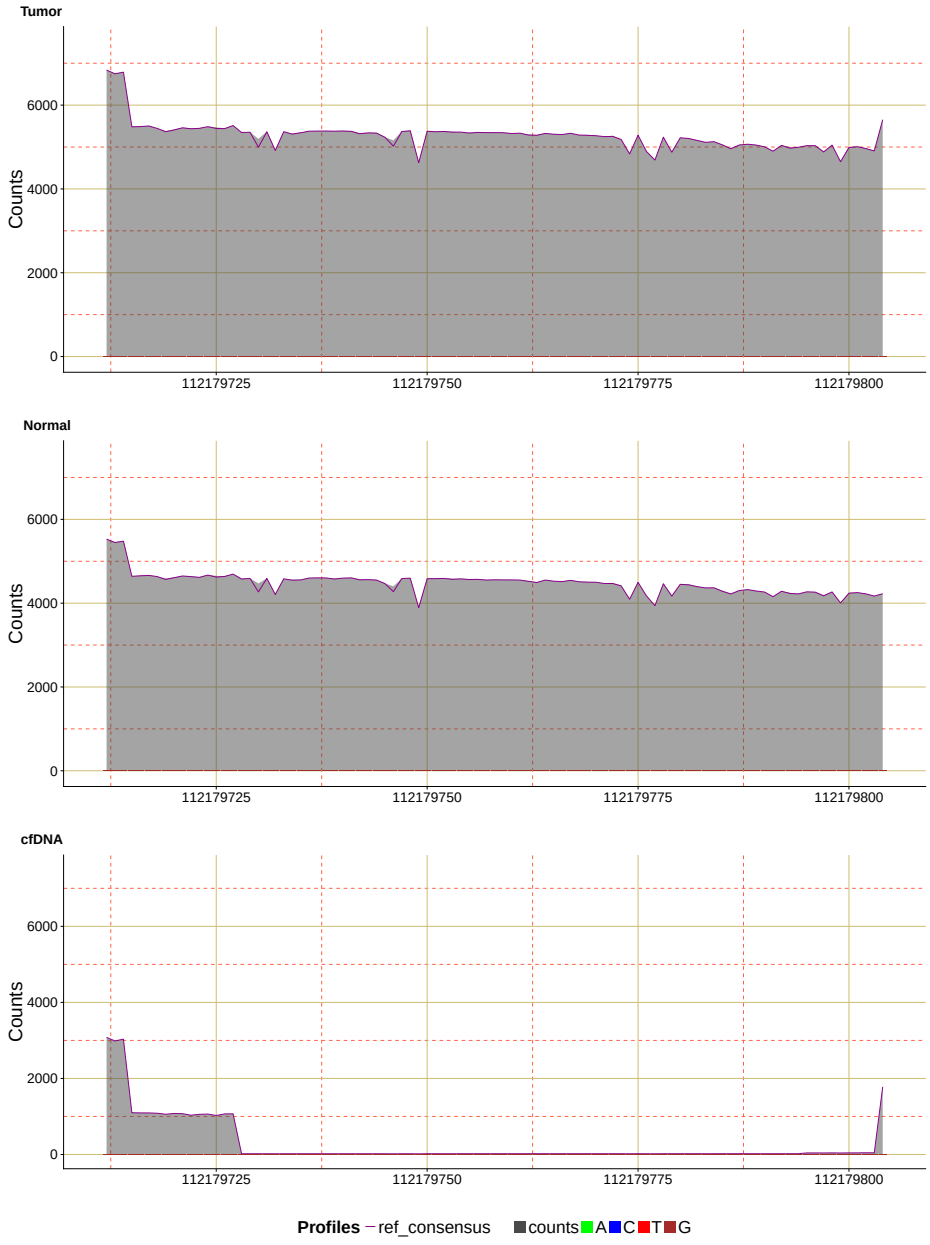

Figura 3.9: Perfil de lecturas de la *feature 224537542* en las muestras **A)** *Tumor*, **B)** *Normal* y **C)** *cfDNA*. El eje x representa la posición genómica, y el eje y muestra la cantidad de lecturas que se contaron para cada posición. La curva violeta indica las lecturas que coinciden con la referencia, mientras que las variantes nucleotídicas que superan una frecuencia mínima de 0,05 y en posiciones que por lo menos han sido leídas 10 veces se identifican con barras coloreadas según la leyenda indicada. El sombreado gris indica el total de lecturas por cada posición.

la *feature* ha sido secuenciada en las muestras *Tumor* y *Normal* con cobertura prácticamente uniforme, superior a 4.000. Mientras tanto, en la muestra *cfDNA*, sólo las primeras bases se han secuenciado, con cobertura superior a 1.000, y el resto no han sido leídas. Es así que el experimento no resulta útil para detectar variantes nucleotídicas en dicha región en la muestra *cfDNA*.

3.4. Conclusiones

El control de calidad es un proceso crucial que debe ser realizado en etapas tempranas del análisis de datos de secuenciación. Dependiendo de la aplicación que se esté estudiando, el control de calidad estará enfocado a evaluar diferentes aspectos y resultados por lo que resulta fundamental contar con herramientas específicas que permitan indagarlos.

En la primera aplicación aquí presentada se utilizó un conjunto de datos reales de RNA-seq. Se demostró que la utilización de herramientas del análisis estadístico de datos multivariados, en etapas tempranas, permitió develar la existencia de muestras atípicas o *outliers* que debieron ser removidas. Adicionalmente, se demostró que la **ausencia del control** impacta directamente en los resultados de las etapas posteriores de análisis de expresión diferencial. Por lo tanto, quedó en evidencia la necesidad y utilidad de incorporar controles dirigidos a asegurar y/o optimizar la calidad de los resultados intermedios de las distintas etapas de análisis con el fin de optimizar los resultados finales del experimento y otorgar confiabilidad a las conclusiones abordadas.

La segunda aplicación desarrollada evidenció la utilidad de la herramienta desarrollada, `TarSeqQC`, en el **control de calidad** no sólo de experimentos de RNA-seq sino en cualquier aplicación de las NGS donde se pueda definir una relación *feature*-gen. En particular, se ilustró el uso de esta herramienta utilizando un conjunto de datos de secuenciación dirigida. Se demostró que el control de calidad en etapas tempranas del análisis evita **reportes incorrectos** involucrando *features* incluso pooles de PCR de bajo rendimiento. `TarSeqQC` permite conocer la identidad de los genes/*features* bajo rendimiento, lo que implica conocer en qué regiones no se reportarán variantes por ineficacia del experimento, lo cual no implica que éstas no puedan existir.

En las dos aplicaciones aquí presentadas se ha dejado en evidencia la importancia de contar tanto con un flujo de análisis ordenado y estructurado como con

herramientas de control de calidad apropiadas para la exploración y visualización de resultados intermedios.

Capítulo 4

Comparación de métodos de análisis de splicing alternativo

4.1. Introducción

RNA-seq es la técnica más utilizada para analizar la dinámica del transcriptoma, incluidos los cambios en el SA. Como se explicó anteriormente en el Capítulo 2, en el análisis de SA se pueden estudiar dos *tipos de cambios* que involucran modificaciones en la expresión de las isoformas: DIE y DS. En tales casos, este análisis se realiza utilizando cuantificación de expresión a nivel de isoformas y/o exones, en vez de genes como se hace en DGE (Mortazavi et al., 2008). En este contexto, se han publicado diversos trabajos evaluando y comparando los distintos métodos y estrategias de cuantificación existentes, para lo cual se han utilizado tanto datos *sintéticos* (simulados) de RNA-seq como datos reales (Kanitz et al., 2015; Soneson et al., 2016; Teng et al., 2016). Adicionalmente, se han desarrollado numerosas herramientas para estudiar DIE o DS, generalmente en forma separada (Anders et al., 2012; Aschoff et al., 2013; Leng et al., 2013; Trapnell et al., 2012). Sin embargo, predomina la existencia de herramientas específicas para DS, mientras que para DIE se han utilizado las diseñadas para DGE. Si bien éstas son ampliamente conocidas y empleadas para el análisis de DGE, su desempeño al utilizar cuantificación a nivel de isoformas cuya expresión está correlacionada, no ha sido evaluada en profundidad. De forma complementaria, si bien se han desarrollado trabajos reportando comparaciones de metodologías para la detección y el estudio del SA, éstos se basan fundamentalmente en caracterizaciones *descriptivas* y *cualitativas* como el enfoque que utilizan, el uso o no de anotación, el

manejo de lecturas paired-end, el nivel de cuantificación, entre otros (Alamancos et al., 2014; Hooper, 2014; Wang et al., 2015). La falta de evaluaciones *objetivas* de desempeño, basadas en medidas específicas estimadas sobre conjuntos de datos, sumado al creciente número de herramientas disponibles dificultan al investigador la tarea de selección de las herramientas adecuadas para analizar sus datos. En este contexto, hasta donde la autora conoce, sólo se ha reportado un trabajo evaluando *sistemáticamente* ocho programas utilizando datos simulados y reales de RNA-seq (Liu et al., 2014). No obstante, este trabajo sólo se enfocó en análisis de DS, sin consideración de la ocurrencia de DIE, y en datos de plantas, organismos cuyo genoma difiere bastante de los humanos. Además, hasta donde la autora conoce, no se han reportado trabajos evaluando desempeños de *flujos de procesamiento* como un todo, lo cual resulta fundamental a la hora de decidir las herramientas que se utilizarán para analizar nuestros datos.

En este capítulo se presenta una evaluación sistemática y comparación objetiva de nueve flujos de procesamiento (*pipelines* o *workflows*) para la detección de DIE y DS en datos de RNA-seq sintéticos y reales. Para evaluar el desempeño de los *pipelines* se diseñaron distintos escenarios experimentales donde se varió el número de réplicas disponibles así como también la proporción de genes/isoformas con DS/DIE. La evaluación y la comparación que aquí se presenta se basó en medidas de desempeño bien conocidas para tal fin. Con los resultados obtenidos se establecieron guías prácticas para asistir la selección del *pipeline* más apropiado para el análisis según el número de replicas disponible, el balance entre sensibilidad y precisión deseado, entre otros. Los resultados de este trabajo han sido recientemente publicados en la revista *Briefings in Bioinformatics* de la editorial *Oxford* (Merino et al., 2017a).

4.2. Materiales y Métodos

4.2.1. Flujos de análisis

Los *pipelines* evaluados en este trabajo se basaron en siete métodos comúnmente utilizados para analizar cambios de expresión: `Cuffdiff2` (Trapnell et al., 2012), y los paquetes `R` `EBSeq` (Leng et al., 2013), `Limma` (Ritchie et al., 2015), `DESeq2` (Love et al., 2014), `NOISeq` (Tarazona et al., 2015), `SplicingCompass` (Aschoff et al., 2013) y `DEXSeq`. Con ellos se diseñaron nueve flujos de trabajos

para analizar DIE y DS, ilustrados en la Figura 4.1. La denominación utilizada para la identificación de los *pipelines* fue: Cufflinks, DESeq2, EBSeq, Limma y NOISeq, en el caso de los flujos DIE (flechas y líneas sólidas), y CufflinksDS, DEXSeq, LimmaDS y SplicingCompass, para los flujos que analizan DS (flechas y líneas entrecortadas). Es menester destacar que de las herramientas seleccionadas, sólo dos de ellas, `Cuffdiff2` y `Limma`, son capaces de analizar los cambios de expresión de ambos tipos, DIE y DS.

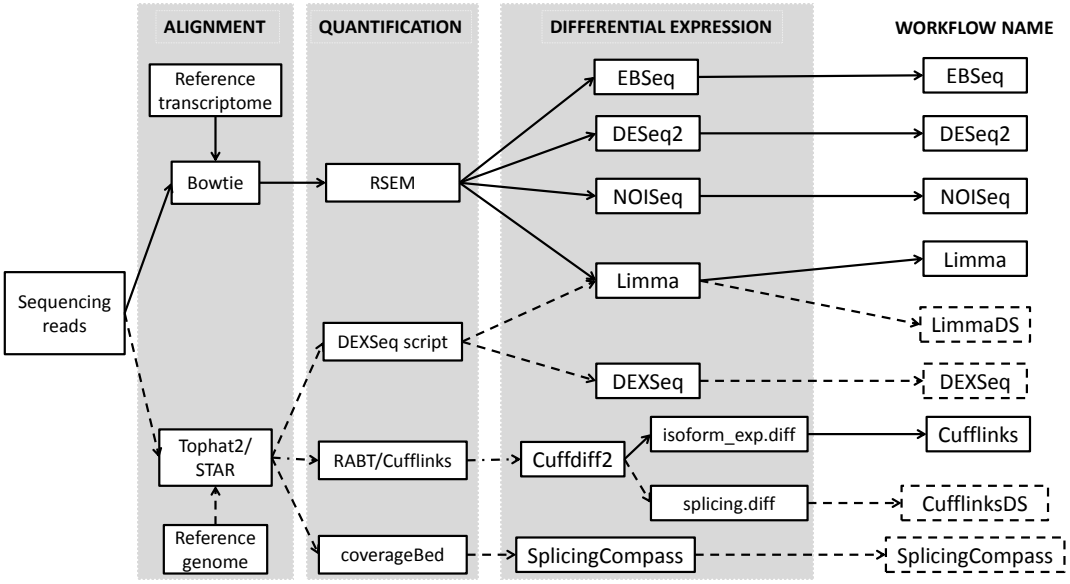

Figura 4.1: Esquema de los nueve *pipelines* evaluados. Cinco de ellos se utilizaron para analizar expresión diferencial de isoformas (flechas y líneas sólidas), mientras que los cuatro *pipelines* restantes se emplearon para analizar splicing diferencial (flechas y líneas a trazos). Todos los *pipelines* comienzan su análisis a partir de lecturas de secuenciación y finalizan generando una lista de isoformas con expresión diferencial o genes con splicing diferencial. Figura extraída de Merino et al. (2017a).

Flujos del tipo DIE

Los flujos del tipo DIE, o simplemente flujos DIE, son aquéllos dirigidos a la identificación de cambios en la expresión absoluta de las isoformas. Este grupo de *pipelines* utilizó métodos de cuantificación basados en modelos probabilísticos a nivel de isoformas. Estos modelos tratan de asignar las lecturas a los **transcritos** o **fragmentos** de ellos de donde se supone han sido originadas. Para ello, modelan la incertidumbre derivada del solapamiento de las isoformas de un mismo gen (Liu et al., 2014). En particular, excepto para el *workflow* Cufflinks, se utilizó la herra-

mienta `RSEM` (v1.2.30, Li and Dewey, 2011) para generar las matrices de expresión a nivel de isoformas. Se tomó como punto de partida a las lecturas de secuenciación y se las alineó contra el transcriptoma humano de referencia utilizando el software `Bowtie` (v1.0.0, Langmead et al., 2009), siguiendo las sugerencias de Teng et al. (2016) y Liu et al. (2014). Luego, se utilizaron los paquetes `R` correspondientes a cada *pipeline* para evaluar DIE. De ellos, `EBSEq` (v.1.10.0) y `DESeq2` (v.1.10.1) asumen que los valores de expresión crudos han sido generados de una distribución NB, mientras que `Limma` (v3.269.9) asume que los datos transformados mediante la función logaritmo siguen una distribución normal. Para inferir los cambios de expresión al comparar diferentes condiciones experimentales, `EBSeq` usa modelos bayesianos jerárquicos (Leng et al., 2013), mientras que `DESeq2` combina modelos bayesianos con GLMs para obtener estimaciones de los coeficientes correspondientes que posteriormente serán evaluados utilizando el estadístico de Wald (Love et al., 2014). En el caso de `Limma`, éste utiliza una transformación, denominada *voom*, que deriva en datos con distribución normal heterocedástica, por lo que luego aplica el enfoque de mínimos cuadrados generalizados utilizando pesos de precisión para modelar la relación media-varianza. Los cambios en la expresión son luego detectados utilizando la función de `Limma`, *eBayes*, ampliamente utilizada en el análisis de datos de microarreglos de ADN (Ritchie et al., 2015). En el caso de `NOISeq`, se utilizó la herramienta `NOISeqbio` (v2.14.1) para analizar DIE. En particular, ésta no asume distribución alguna para los datos de expresión ya que es un método no paramétrico que se adapta a los datos y utiliza los fold-changes y las diferencias absolutas en los valores de expresión para así definir un estadístico para cada isoforma. Este estadístico será comparado con la distribución de ruido que `NOISeq` estima en un proceso de permutaciones de los datos, obteniendo para cada isoforma una estimación de la probabilidad de que dicha isoforma haya estado diferencialmente expresada (P_{de}), de manera que su complemento representa un análogo a un p-valor ajustado. Cabe destacar que el resultado de la ejecución de cada flujo es una lista de isoformas significativamente detectadas como diferencialmente expresadas (DI).

Flujos del tipo DS

Los *pipelines* DS tienen como objetivo la identificación de genes que han evidenciado alteraciones en los patrones de SA. Para ello, utilizan matrices de expresión a distintos niveles, los cuales dependen del enfoque en que se basa la herramienta del análisis diferencial. Pese a esto, todos ellos comprenden como

primer paso el alineamiento contra el genoma humano de referencia. En particular, en este trabajo, este paso se realizó utilizando el software `TopHat2` (v2.0.9, Kim et al., 2013). Cada *workflow* utilizó un método de cuantificación diferente, siguiendo las recomendaciones de los autores de los software de DS. En el caso del flujo SplicingCompass, se utilizó la herramienta `coverageBed` (v2.17.0, Quinlan and Hall, 2010) para obtener matrices de expresión mediante un modelo de unión de transcritos donde todos los exones de todas las isoformas de un gen son considerados. Con estas matrices a nivel de exones y sitios de unión (*junctions*), `SplicingCompass` (v1.0.1) construye vectores para cada gen de cada muestra. Los ángulos entre vectores de dos muestras son calculados y comparados usando un test t para determinar DS (Aschoff et al., 2013). Las matrices de expresión utilizadas en los *pipelines* DEXSeq y LimmaDS fueron las mismas y se generaron utilizando el script de `python` que provee el paquete `DEXSeq` (v1.16.10) para tal fin, siguiendo las sugerencias de Soneson et al. (2016). A diferencia de `Limma`, `DEXSeq` asume que los conteos que determinan la expresión han sido generados de una distribución NB, como `DESeq2`. Pese a esto, el enfoque que siguen las dos herramientas es incorporar un término de interacción al modelo entre el exón que se analiza y el factor que indica la condición experimental para así poder evaluar cambios en el uso de exones entre condiciones como medida indirecta del DS. Específicamente, las funciones que se utilizaron para determinar DS son `initSigGenesFromResults` (SplicingCompass), `perGeneQValue` (DEXSeq) y `diffSplice` (LimmaDS). Las tres funciones permiten obtener valores p ajustados a nivel de gen que indican la significancia del cambio detectado. Como resultado de la ejecución de cada *pipeline* se obtiene una lista de genes cuyos cambios en el SA resultaron significativos (ASG). En el caso de los *pipelines* basados en las *Tuxedo tools*, `Cufflinks2` (v2.1.1) se usó primero para obtener expresión a nivel de isoformas en escala de FPKMs a partir de las lecturas alineadas al genoma mediante TopHat2. Posteriormente, `Cuffdiff2` se empleó para realizar los análisis de DE a nivel de isoformas y de SA, generando las salidas de los dos flujos evaluados.

4.2.2. Bases de datos

Experimento real de RNA-seq

Como punto de partida de la construcción de las bases de datos a analizar se consideró un conjunto de 30 muestras reales de datos de RNA-seq, de un experimento que comparó muestras de tejido de pacientes con cáncer de próstata con muestras de tejido normal (Número de acceso GEO: GSE22260, Kannan et al., 2011). En particular, este conjunto de datos comprende 10 pacientes con cáncer de próstata a los cuales se ha extraído sólo tejido tumoral y 10 pacientes a los que se les extrajo tanto tejido sano o normal como tumoral. Es decir que la condición normal tiene 10 muestras y la condición tumor tiene 20, 10 de las cuales están pareadas con las muestras normales. Para evitar el efecto sujeto, estas 10 muestras tumorales se descartaron. Posteriormente, se alinearon las lecturas de todas las muestras y se cuantificaron con el fin de analizar la presencia de muestras *outliers* siguiendo el procedimiento descrito en el Capítulo 3. De las 20 muestras, se encontró que cuatro de ellas, dos de cada condición, presentaban comportamiento atípico con respecto al resto, por lo que se decidió eliminarlas. Es así que el conjunto de datos utilizado finalmente consistió de 16 muestras de RNA-seq de un experimento de dos condiciones, normal o control (C) y tumor (T), cada una de las cuales tuvo ocho réplicas. Con este conjunto de datos se realizó por un lado la evaluación de los *pipelines* propuestos y por otro, se generaron diferentes conjuntos de datos sintéticos donde los perfiles de expresión se controlaron para así simular DIE/DS.

Características de los datos reales

La base de datos reales se caracterizó con el objetivo de determinar parámetros importantes para la posterior simulación de datos sintéticos. En este contexto, se exploró la distribución del número de isoformas por gen que presenta el genoma humano de referencia que se utilizó para los análisis. La Figura 4.2 muestra la frecuencia absoluta (panel **A**) y la frecuencia relativa acumulada (panel **B**) estimadas a partir de los archivos de anotación. Se determinaron tres grupos de genes, los cuales se identificaron en las gráficas mediante un código de colores. En particular, los grupos definidos representaron, cada uno, un tercio del total de genes con más de una isoforma. Tales grupos son: entre dos y cuatro isoformas ("2-4"), entre cinco y nueve isoformas ("5-9") y más de nueve isoformas ("> 9").

Dichos grupos de genes deberán ser considerados durante la simulación con el fin de respetar la naturaleza de los datos reales.

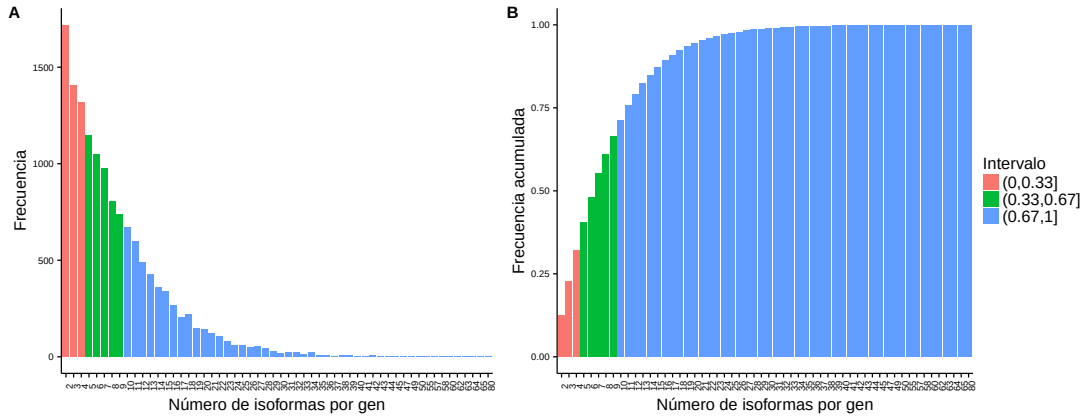

Figura 4.2: Gráficos de barras ilustrando la frecuencia: **A)** absoluta y **B)** relativa acumulada del número de isoformas por gen. En color se destacan los intervalos de frecuencia estimados.

Luego, se exploró el valor de fold change observado en el experimento real, en las isoformas expresadas, agrupadas según el número de isoformas por gen. Los diagramas de cajas correspondientes se ilustran en la Figura 4.3**A**, donde se ha representado al fold change en escala logarítmica en base 2 (log_2). Cabe destacar que a los tres grupos definidos en el párrafo anterior se ha agregado el grupo de genes que tiene sólo una isoforma ("1"). Los gráficos de caja, en los cuatro grupos, están centrados en el valor 0, en escala log_2, lo que se corresponde con el valor 1 en escala decimal, es decir que el cociente de los valores de expresión en la condición T y C vale 1, reflejando el mismo valor para ambas condiciones. Esto es de esperar en casi todos los experimentos de RNA-seq, ya que la mayoría de las isoformas/genes no modifican su expresión ante un cambio en la condición sino que un pequeño porcentaje de ellos se espera que resulten diferencialmente expresados.

Finalmente, se exploraron las proporciones de las isoformas de mayor expresión de cada gen (*isoforma mayor*, M), en la condición C, que fue tomada luego como referencia en la simulación. La Figura 4.3**B** ilustra los diagramas de cajas para dichas proporciones, agrupados según el grupo al que pertenece el gen del que proviene cada isoforma mayor. Se determinó que a medida que el número de isoformas por gen aumenta, la proporción de la isoforma mayor disminuye, lo cual es razonable ya que el gen tiene más alternativas de splicing. Se identificó que en el

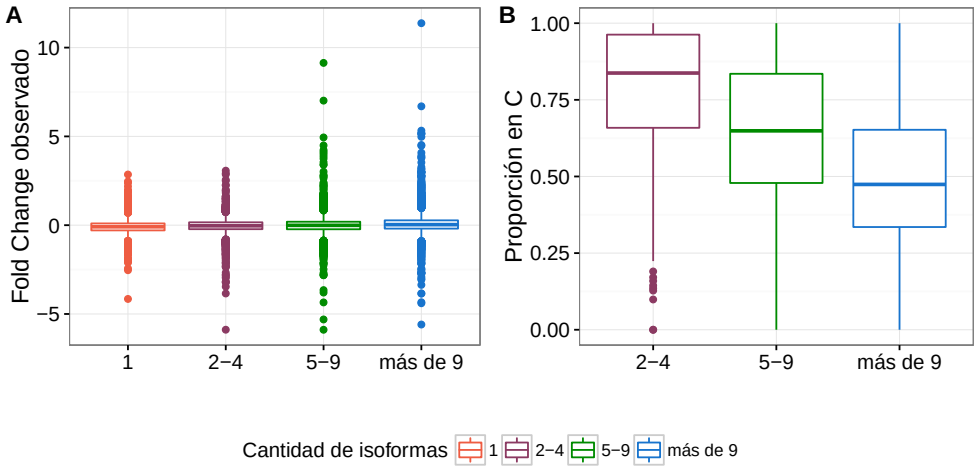

Figura 4.3: Diagramas de cajas de: **A)** Fold change, en escala logartímica en base dos, de la expresión de las isoformas, observado en la condición de referencia (control, C) y **B)** proporción de las isoformas de mayor expresión de cada gen observada en la condición de referencia. Cada diagrama se corresponde con un grupo de genes, categorizados según el número de isoformas que éstos poseen.

grupo "2-4" isoformas, la proporción media fue 0,793 (±0,19), mientras que en el caso del grupo "5-9" resultó de 0,647 (±0,22) y en el grupo "> 9", 0,499 (±0,213).

Simulación

Los datos sintéticos se obtuvieron en tres configuraciones o *escenarios* experimentales con diseño balanceado que se llamaron *S1*, *S2* y *S3*. Los escenarios *S1* y *S2* involucraron ocho muestras en total, 4 réplicas por condición, diferenciándose entre sí en el porcentaje de genes a simular con expresión diferencial: 5 % en *S1* y 10 % en *S2*. El tercer escenario simulado abarcó el mismo porcentaje de genes con DE que *S2* pero consideró ocho réplicas por condición. De esta manera la comparación *S1* vs *S2* se utilizó para develar el efecto de la incremento de la heterogeneidad entre condiciones, mientras que la comparación *S2* vs *S3* se empleó para determinar el efecto del incremento en el número de réplicas en experimentos balanceados.

El proceso para generar las lecturas de secuenciación de las muestras con perfiles de expresión controlados de cada escenario se ilustra en la Figura 4.4. Como se puede apreciar, éste consistió de tres etapas o pasos. En la primera de ellas, cada uno de los 16 archivos de secuenciación que componen la base de datos real se

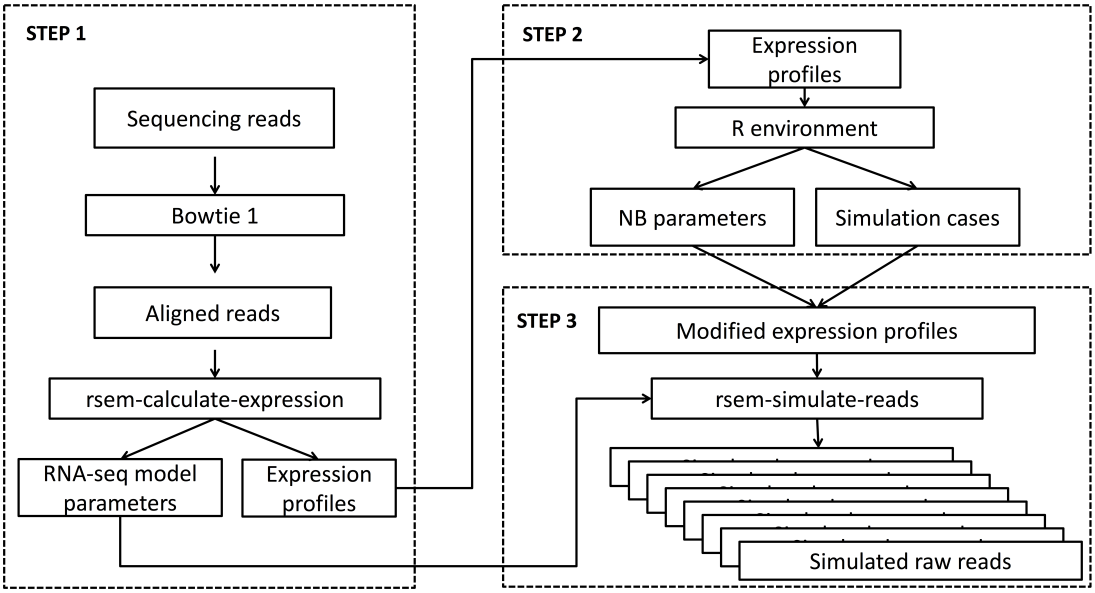

Figura 4.4: Esquema del proceso de simulación diseñado para generar archivos de lecturas de RNA-seq donde se ha controlado los perfiles de expresión de genes e isoformas para simular expresión diferencial de isoformas y splicing diferencial. El proceso se dividió en tres etapas. La primera de ellas se llevó a cabo en todas las muestras de la base de datos real, de a una por vez, y tuvo como objetivo obtener los perfiles de expresión reales así como también parámetros específicos del modelo de secuenciación que aplicará más tarde el simulador. La segunda etapa se realizó una vez para cada escenario experimental simulado con el fin de obtener los parámetros de las distribuciones de las cuales se asume que los conteos de cada gen/isoforma han sido generados. Estos parámetros se obtuvieron por condición y fueron modificados para simular cambios en la expresión absoluta y/o relativa de las isoformas. El tercer paso se ejecutó diez veces por escenario, al fin de generar repeticiones de los mismos. A partir de los perfiles de expresión modificados se generaron valores de conteos para cada isoforma en cada muestra. Éstos, junto con los archivos conteniendo los parámetros del modelo de RNA-seq, se proveyeron al simulador para así generar las lecturas de las muestras correspondientes a cada repetición de cada escenario experimental. Figura extraída de Merino et al. (2017a).

alineó contra el transcriptoma de referencia usando `Bowtie1` y luego se **cuantificó** dichos alineamientos utilizando `RSEM`. La cuantificación generó, por un lado, un archivo con los perfiles de expresión de las isoformas en la muestra y, por el otro, un archivo con parámetros del modelo de RNA-seq que aplica el simulador. Una vez que este primer paso se ejecutó para todas las muestras, se procedió a la ejecución de la segunda etapa. Ésta se llevó a cabo en `R`, donde se conformó la **matriz de expresión** del experimento. El primer procesamiento realizado fue el **control de calidad** de dicha matriz, en el cual se removieron los genes que no tuvieron conteos en al menos una muestra. Posteriormente, se identificó un conjunto de genes de expresión elevada (20 CPMs en al menos una réplica de la condición C), entre los cuales se seleccionó *aleatoriamente* el porcentaje de genes a simular con DE correspondiente a cada escenario. Los conteos que miden la expresión de la i-ésima isoforma del g-ésimo gen en la k-ésima condición, y_{igk}, se asumieron como generados por una distribución NB, $y_{igk} \sim NB(\mu_{igk}, \phi_{igk})$. Los parámetros de las distribuciones NB, μ_{igk} y ϕ_{igk}, se estimaron a partir de los *perfiles de expresión* observados en la condición de referencia (C) para luego **simular** sobre ellos los cambios de expresión. Paralelamente se definió los casos de expresión diferencial que se simularon posteriormente. Luego se procedió a la **modificación** de los perfiles de expresión. En el caso de los genes que se simularon sin cambios en la expresión, se asignó a ambas condiciones los mismos parámetros distribucionales, tomados de la condición C. Los genes seleccionados para ilustrar la expresión diferencial fueron divididos *aleatoriamente* en **subgrupos de simulación** y los parámetros distribucionales se modificaron de acuerdo con dichos subgrupos, que se explican en el siguiente apartado. Notablemente, este paso se ejecutó una sola vez para cada uno de los tres escenarios experimentales. Posteriormente, se ejecutó el tercer paso que consistió en la simulación propiamente dicha de las *lecturas* de secuenciación a partir de los perfiles de expresión alterados y de los parámetros del modelo de RNA-seq estimados para cada muestra en el primer paso. Este tercer paso se realizó diez veces por cada escenario para tener así repeticiones de los mismos y poder otorgar validez estadísticas a las comparaciones posteriormente realizadas. El proceso de simulación en sí fue realizado mediante la herramienta `rsem-simulate-reads` de `RSEM`, la cual permitió generar, de a uno por vez, los archivos FASTQ con las lecturas de secuenciación correspondientes a los perfiles de expresión deseados (Li and Dewey, 2011).

Grupos y subgrupos simulados

Los genes aleatoriamente seleccionados para ser simulados con algún tipo de expresión diferencial fueron divididos en cuatro grandes **grupos**, los cuales a su vez fueron divididos en **subgrupos** acorde con a la magnitud del cambio en la expresión que se simuló. Los cambios en la expresión se reflejaron en alteraciones en los parámetros de las distribuciones de las que se asumió que los datos de expresión provienen. Cabe recordar que las modificaciones se realizaron sólo en una de las condiciones, tomando como referencia a la condición C. Los cuatro grandes grupos considerados fueron:

- **DIE:** En este grupo se consideraron genes que tenían *más de un* transcrito anotado en la referencia. Todas las isoformas de los genes en este grupo, que originalmente se encontraron expresadas en la condición C, fueron simuladas con cambios en su **expresión absoluta**, es decir del tipo DIE. Los cambios simulados involucraron fold changes de 2, 3, 4 y 5, lo que dividió a este grupo en cuatro subgrupos de simulación. Cabe destacar que dichos cambios se consideraron en ambas direcciones, es decir, algunos genes incrementaron la expresión de todas sus isoformas en T al doble que en C (fold change de 2), mientras que otros tuvieron todas sus isoformas expresadas a la mitad en T respecto de C. Dado un gen g, n_g (> 1) indica el número de isoformas de dicho gen. Si g fue seleccionado para ser simulado como DIE, con un fold change a, entonces cada una de sus isoformas (i) tendrá un fold change a (igual a 2, 3, 4 o 5) en la comparación T vs C. Para lograr lo dicho anteriormente, los parámetros media (μ) y varianza (σ^2) de las condiciones C y T se determinaron según la Ecuaciones 4.1 y 4.2, respectivamente. Por el contrario, para simular los fold changes 0,5, 0,33, 0,25 y 0,2 correspondientes a los valores 2, 3, 4 y 5 de la comparación CvsT se utilizaron las Ecuaciones 4.3 y 4.4 para simular los parámetros de las distribuciones correspondientes a la condición C y T, respectivamente:

$$\mu_{Ci} = \mu_{C0i}, \quad \sigma^2_{Ci} = \sigma^2_{C0i} \quad \forall i = 1, ..., n_g \tag{4.1}$$

$$\mu_{Ti} = a\mu_{C0i}, \quad \sigma^2_{Ti} = a^2\sigma^2_{C0i} \quad \forall i = 1, ..., n_g \tag{4.2}$$

$$\mu_{Ci} = a\mu_{C0i}, \quad \sigma^2_{Ci} = a^2\sigma^2_{C0i} \quad \forall i = 1, ..., n_g \tag{4.3}$$

$$\mu_{Ti} = \mu_{C0i}, \quad \sigma^2_{Ti} = \sigma^2_{C0i} \quad \forall i = 1, ..., n_g \tag{4.4}$$

donde:
μ_{Ci} y σ^2_{Ci} son los parámetros distribucionales media y varianza de la expre-

sión de la i-ésima isoforma del gen g en la condición C, modificados para simular cambios de expresión,
μ_{Ti} y σ^2_{Ti} son los parámetros distribucionales media y varianza de la expresión de la i-ésima isoforma del gen g en la condición T, modificados para simular cambios de expresión,
μ_{C0i} y σ^2_{C0i} son la media y varianza muestral de la expresión de la i-ésima isoforma del gen g, respectivamente, observadas para la condición C en los datos reales de RNA-seq,
a es el fold change simulado para el cambio de expresión.

- **DE:** En este grupo se consideraron los genes aleatoriamente seleccionados para simular DE que *sólo* poseen una isoforma anotada, es decir, que no evidencian SA. Luego, cada gen fue simulado con DE considerando fold changes de 2 y 4, en ambas direcciones. Es decir, que los genes dentro de este grupo se dividieron en dos subgrupos. Para llevar a cabo las simulaciones se utilizaron las Ecuaciones 4.1, 4.2, 4.3 y 4.4 con la particularidad de que en todos los casos $n_g = 1$.

- **DS:** Este grupo involucró genes que tuvieron *más de una isoforma* anotada para así poder simular la ocurrencia de **cambios en el SA**. En particular, los cambios simulados fueron tales que la expresión total del gen en las condiciones C y T fue la misma, alterando las proporciones de las isoformas correspondientes. El control del cambio simulado se ejerció sobre la isoforma de mayor expresión (M), identificada en la condición C observada. La proporción (p) de M se cambió en varios niveles para así determinar cuatro subgrupos: $0-0{,}7$, $0{,}1-0{,}4$, $0{,}3-0{,}6$ y $0{,}5-0{,}8$, en ambas direcciones. La notación $X-Y$ indica que en la condición C la proporción simulada de M fue X, mientras que en T fue Y. El resto de las isoformas del gen se simuló con la misma expresión, igual a la proporción remanente $(1-p)$ dividido por $n_g - 1$. Cabe destacar que la asignación de genes a cada subgrupo se realizó considerando las proporciones observadas en los datos reales (ver Figura 4.3**B**) de manera de simular situaciones reales. Es así que sólo el grupo de genes $2-4$ se consideró para simular el subgrupo de cambios $0-0{,}7$, mientras que dicho grupo de genes no se consideró para el caso $0{,}1-0{,}4$. Luego, para el cálculo de los parámetros distribucionales de la expresión de la isoforma M se utilizaron las Ecuaciones 4.5, para C, y 4.6, para T.

Mientras, los parámetros distribucionales del resto de las isoformas en C y T se calcularon con las Ecuaciones 4.8 y 4.8, respectivamente:

$$\mu_{CMg} = p_C \mu_{C0g}, \quad \sigma^2_{CMg} = p^2_C \sigma^2_{C0g} \tag{4.5}$$

$$\mu_{TMg} = p_T \mu_{C0g}, \quad \sigma^2_{TMg} = p^2_T \sigma^2_{C0g} \tag{4.6}$$

$$\mu_{Ci} = p_{Ci}\mu_{C0g}, \quad \sigma^2_{Ci} = p^2_{Ci}\sigma^2_{C0g}, \quad p_{Ci} = \frac{1 - p_C}{n_g - 1} \;\; \forall i = 1, ..., n_g \;\wedge\; i \neq M \tag{4.7}$$

$$\mu_{Ti} = p_{Ti}\mu_{C0g}, \quad \sigma^2_{Ti} = p^2_{Ti}\sigma^2_{C0g}, \quad p_{Ti} = \frac{1 - p_T}{n_g - 1} \;\; \forall i = 1, ..., n_g \;\wedge\; i \neq M \tag{4.8}$$

donde:
μ_{CMg} y σ^2_{CMg} son los parámetros distribucionales media y varianza de la expresión de la isoforma mayor (M) del gen g en la condición C, modificados para simular cambios de expresión relativa,
μ_{TMg} y σ^2_{TMg} son los parámetros distribucionales media y varianza de la expresión de la isoforma mayor (M) del gen g en la condición T, modificados para simular cambios de expresión relativa,
μ_{C0g} y σ^2_{C0g} son la media y varianza muestral de la expresión del gen g, respectivamente, observadas para la condición C en los datos reales de RNA-seq,
p_C y p_T son las proporciones de la isoforma mayor simuladas en las condiciones C y T, respectivamente,
μ_{Ci} y σ^2_{Ci} son los parámetros distribucionales media y varianza de la expresión de la i-ésima isoforma del gen g, distinta de la isoforma M, en la condición C, modificados para simular cambios de expresión relativa,
μ_{Ti} y σ^2_{Ti} son los parámetros distribucionales media y varianza de la expresión de la i-ésima isoforma del gen g, distinta de la isoforma M, en la condición T, modificados para simular cambios de expresión relativa.

- **DIEDS:** Este grupo abarcó genes con *más de una isoforma* anotada a los que se les simuló **combinada** y **controladamente** cambios del tipo DIE y DS. Para ello, se modificó la expresión absoluta del gen y la expresión relativa de sus isoformas entre las condiciones de interés. En particular, se consideraron cinco subgrupos de simulación: $0{,}5 - 0{,}8 - 0{,}5$, $2 - 0{,}8 - 0{,}5$, $4 - 0{,}8 - 0{,}5$, $2 - 0{,}8 - 0{,}3$ y $4 - 0{,}8 - 0{,}3$. El nombre de cada subgrupo

contiene tres números indicando: el fold change (a) de TvsC, la proporción de la isoforma M en C (p_C) y la proporción de ésta en T (p_T), respectivamente. Para simular los cambios descritos se utilizaron las Ecuaciones 4.9 y 4.10 para la isoforma M en las condiciones C y T, respectivamente; mientras que las Ecuaciones 4.11 y 4.12 se utilizaron para modificar los parámetros distribucionales de el resto de las isoformas en las condiciones C y T, respectivamente.

$$\mu_{CMg} = p_C \mu_{C0g}, \quad \sigma^2_{CMg} = p_C^2 \sigma^2_{C0g} \tag{4.9}$$

$$\mu_{TMg} = a p_T \mu_{C0g}, \quad \sigma^2_{TMg} = a^2 p_T^2 \sigma^2_{C0g} \tag{4.10}$$

$$\mu_{Ci} = p_{Ci}\mu_{C0g},\ \sigma^2_{Ci} = p^2_{Ci}\sigma^2_{C0g},\ p_{Ci} = \frac{1-p_C}{n_g-1}\ \forall i = 1,...,n_g\ \wedge\ i \neq M \tag{4.11}$$

$$\mu_{Ti} = a p_{Ti}\mu_{C0g},\ \sigma^2_{Ti} = a^2 p^2_{Ti}\sigma^2_{C0g},\ p_{Ti} = \frac{1-p_T}{n_g-1}\ \forall i = 1,...,n_g\ \wedge\ i \neq M \tag{4.12}$$

donde:

μ_{CMg} y σ^2_{CMg} son los parámetros distribucionales media y varianza de la expresión de la isoforma mayor (M) del gen g en la condición C, modificados para simular cambios de expresión relativa,

μ_{TMg} y σ^2_{TMg} son los parámetros distribucionales media y varianza de la expresión de la isoforma mayor (M) del gen g en la condición T, modificados para simular cambios de expresión relativa,

μ_{C0g} y σ^2_{C0g} son la media y varianza muestral de la expresión del gen g, respectivamente, observadas para la condición C en los datos reales de RNA-seq,

p_C y p_T son las proporciones de la isoforma mayor simuladas en las condiciones C y T, respectivamente,

μ_{Ci} y σ^2_{Ci} son los parámetros distribucionales media y varianza de la expresión de la i-ésima isoforma del gen g, distinta de la isoforma M, en la condición C, modificados para simular cambios de expresión relativa,

μ_{Ti} y σ^2_{Ti} son los parámetros distribucionales media y varianza de la expresión de la i-ésima isoforma del gen g, distinta de la isoforma M, en la condición T, modificados para simular cambios de expresión relativa.

a es el fold change del cambio en la expresión.

A modo de resumen, la Tabla 4.1 contiene la definición de los grupos y subgrupos de simulación y la denominación que se ha utilizado para su identificación.

Tabla 4.1: Grupos y subgrupos simulados, definidos a partir de la combinación de cambios en expresión absoluta y relativa de las isoformas y/o genes entre las dos condiciones experimentales bajo estudio.

Grupo de simulación	Fold change de expresión absoluta	Cambio en la proporción de la isoforma más expresa	Subgrupo de simulación
DIE	2	Sin cambio	DIE-2
	3	Sin cambio	DIE-3
	4	Sin cambio	DIE-4
	5	Sin cambio	DIE-5
DE	2	Sin cambio	DE-2
	4	Sin cambio	DE-4
DS	Sin cambio	0-0,7	DS-0-0,7
	Sin cambio	0,1-0,4	DS-0,1-0,4
	Sin cambio	0,3-0,6	DS-0,3-0,6
	Sin cambio	0,5-0,8	DS-0,5-0,8
DIEDS	0,5	0,8-0,5	DIEDS-0,5-0,8-0,5
	2	0,8-0,3	DIEDS-2-0,8-0,3
	2	0,8-0,5	DIEDS-2-0,8-0,5
	4	0,8-0,3	DIEDS-4-0,8-0,3
	4	0,8-0,5	DIEDS-4-0,8-0,5

4.2.3. Análisis de expresión diferencial

El análisis de los datos simulados comenzó con la *preparación* de los archivos de anotación y su *indexación*, necesario para los pasos posteriores. Primero se utilizó `Bowtie1` para generar los índices del genoma y del transcriptoma de referencia (hg19/GrCh37 v.75 extraído de Ensembl). Luego, el archivo de anotación de *features* (archivo *gtf*) se formateó usando el script de `pyhton dexseq_prepare_annotation.py` (con la opción `--aggregate='no'`) provisto con el paquete `DEXSeq`, y el script provisto por `SplicingCompass`. Las lecturas se **alinearon** contra el transcriptoma, con `Bowtie1`. Los alineamientos resultantes se **cuantificaron** luego con la herramienta `rsem-calculate-expression` de `RSEM` para así obtener los perfiles de expresión a nivel de isoformas necesarios para el análisis con los flujos DIE. Por otro lado, se utilizó `Tophat2` (combinado con `Bowtie1`) para obtener los **alineamientos** contra el genoma, que posteriormente fueron **cuantificados** según se especificó en la definición de los flujos de análisis DS DEXSeq, LimmaDS y SplicingCompass. Paralelamente, se procesó las lectu-

ras de secuenciación simuladas con los flujos Cufflinks y CufflinksDS obteniendo así los archivos *isoform_exp.diff* y *splicing.diff*, respectivamente.

En el caso de los flujos DIE, excepto Cufflinks, una vez obtenidos los perfiles de expresión, éstos se cargaron en `R` para así construir la **matriz de expresión**, una por cada repetición de cada escenario. Las matrices de expresión fueron preprocesadas para **filtrar** las isoformas de baja expresión. El criterio utilizado para su identificación fue que tuviesen menos de 4 CPMs en al menos una condición. También se eliminaron del análisis aquellas isoformas que no se simularon como expresadas. Para DESeq2 se utilizó el valor-p ajustado por FDR generado por la función `nbinomWaldTest` para determinar las DI. En el caso de Limma, esta tarea se realizó utilizando la función de `R` `fdr` para ajustar los valores p generados mediante la función `eBayes`, una vez aplicada la transformación `voom`. En el caso de EBSeq y NOISeq, los valores de probabilidad a posteriori de estar diferencialmente expresados que ambos generan (PPDE) se utilizaron para obtener la lista de DI, siguiendo las recomendaciones de los autores que sugieren que $1 - PPDE$ es equivalente a un valor p ajustado por FDR (Aschoff et al., 2013; Tarazona et al., 2015). En todos los casos, la identificación de las DI se hizo considerando un umbral de significancia de 0,05.

Las matrices de expresión de los flujos DS DEXSeq, LimmaDS y SplicingCompass también fueron construidas en `R` a partir de los perfiles de expresión. En este último *pipeline*, la función `initSigGenesFromResults` se utilizó para identificar los ASG, mientras que en el caso de `DEXSeq` se utilizó el método `testForDEU` para evaluar el uso diferencial de exones y seguidamente, la función `perGeneQValue` para computar un valor p ajustado para cada gen. Con éstos se procedió luego a la identificación de los ASG para dicho *pipeline*. En el caso de LimmaDS, siguiendo las sugerencias de los autores, la matriz de expresión fue previamente procesada para filtrar exones de baja expresión. Para ello se consideró como suficiente que un exón tuviese más de 1 CPM en al menos tres muestras. Luego sí se aplicó la transformación `voom`, se ajustó los modelos con la función `lmFit` y se analizó DS con `diffSplice`, obteniendo así un valor p ajustado para cada gen analizado. Al igual que en los *pipelines* DIE, se consideró un umbral de significancia de 0,05 para identificar los ASG.

Los *scripts* utilizados para analizar los datos se encuentran disponibles como material suplementario de Merino et al. (2017a) y también en un proyecto de-

positado en el repositorio GitHub[1] donde además se proporciona un instructivo para guiar tanto el análisis como la replicación del proceso de simulación.

4.2.4. Evaluación del desempeño

El desempeño de los nueve *pipelines* fue *evaluado* mediante medidas **objetivas** comúnmente utilizadas para tal fin (Fernandez et al., 2005; Liu et al., 2014; Soneson et al., 2016), las cuales se listan en la Tabla 4.2. En el contexto del análisis de expresión diferencial, cada gen/isoforma de la lista de ASG/DI obtenida con la ejecución de cada *pipeline*, en cada una de las replicaciones de los diferentes escenarios, fue llamado un *positivo* (P). A su vez, éstos se clasificaron como *verdaderos positivos* (TPs, del inglés *true positives*) o *falsos positivos* (FPs, del inglés *false positives*) dependiendo de si realmente fueron o no simulados como diferencialmente expresado, respectivamente. Cada gen/isoforma cuyo cambio en el SA/expresión se detectó como no significativo fue llamado *negativo* (N) y clasificado en *verdadero negativo* (TN, del inglés *true negative*), cuando verdaderamente se simuló sin cambios en la expresión, y *falso negativo* (FN, del inglés *false negative*) cuando se simuló como diferencialmente expresado, pero no se detectó como tal. Específicamente, para los *pipelines* DIE se consideró como TPs todas aquellas isoformas identificadas como DI, por el flujo correspondiente, que se simularon dentro de los grupos **DIE**, **DE** y **DIEDS** pero también se incluyó a las del grupo **DS**. Esto se debe a que cambios del tipo DS pueden causar cambios DIE y como en ese grupo no se controló dicho efecto, éste pudo o no haber existido. Sin embargo, en el cálculo de los FNs, las isoformas pertenecientes en el grupo **DS** no se consideraron, ya que no necesariamente el DS causará DIE. Una vez que todos los genes/isoformas expresos se clasificaron en TP, FP, FN y VN se calcularon las medidas listadas en la Tabla 4.2, obteniendo una medida por repetición de cada uno de los escenarios evaluados.

Una vez evaluadas las medidas de desempeño, se determinó la **relación** entre éstas y los valores de expresión así como también la longitud de los genes/isoformas para determinar posibles sesgos. Para ello se definieron grupos de expresión/longitud sobre los cuales se computaron las medidas de desempeño. Posteriormente, se determinó el efecto del subgrupo de simulación, por ejemplo **DIE**-2, sobre el desempeño de los *pipelines* así como también el efecto que ejerce sobre éstos el número de isoformas que tiene un gen. La primera de estas tareas

[1]https://github.com/gamerino/benchmarkingDiffExprAndSpl

Tabla 4.2: Medidas utilizadas para evaluar el desempeño de los flujos de análisis bajo estudio. En el contexto de la expresión diferencial los resultados se pueden clasificar en: diferencialmente expresados (positivos, P) y no diferencialmente expresados (negativos, N). Luego, una isoforma/gen será un *verdadero positivo* (TP) o un *verdadero negativo* (TN) si la clasificación asignada por el *pipeline* se corresponde con la simulada. En caso contrario, será un *falso positivo* (FP) o *falso negativo* (FN) según corresponda.

Medida	**Fórmula de cálculo**
Exactitud (ACC)	$ACC = \frac{TP + TN}{TP + TN + FP + FN}$
Sensibilidad (TPR)	$TPR = \frac{TP}{TP + FN}$
Precisión (PPV)	$PPV = \frac{TP}{TP + FP}$
Tasa de falsos positivos (FPR)	$FPR = \frac{FP}{TN + FP}$
F-score	$F = \frac{2TP}{2TP + FP + FN}$

TP: True Positive; FP: False Positive; TN: True Negative; FN: False Negative

se realizó considerando la división de los genes/isoformas simulados como diferencialmente expresados provista por la Tabla 4.1. En el segundo caso, los genes simulados como diferencialmente expresados se agruparon en cuatro grupos de acuerdo al número de isoformas que tenían. Los cuatro grupos fueron: **1**, **2-4**, **5-9**, $>$ **9**, los cuales se definieron teniendo en cuenta la distribución del número de isoformas por gen observado en la base de datos de RNA-seq real.

4.3. Resultados

4.3.1. Evaluación de la simulación

La primera tarea que se realizó consistió en evaluar si el proceso de simulación diseñado generó los resultados deseados. Para ello, en cada repetición de cada escenario se evaluaron aspectos tales como la separabilidad de las muestras, la correlación entre el valor medio de expresión simulado y el valor observado, la correspondencia entre los fold changes/ratios simulados y los observados, entre otras cosas. Estas tareas se realizaron utilizando los datos de cuantificación

obtenidos a partir de las lecturas simuladas con el flujo `Bowtie1-RSEM`. En este contexto, la Figura 4.5 ilustra el gráfico de dispersión de las primeras dos componentes principales del *PCA* realizado sobre la expresión de las isoformas en las muestras correspondientes a una de las repeticiones del escenario S1 (Figura 4.5**A**) y sobre el valor medio de la expresión de las mismas a lo largo de las diez repeticiones de dicho escenario (Figura 4.5**B**). Similarmente, las Figura 4.6 y Figura 4.7 ilustran los mismos gráficos para los escenarios S2 y S3, respectivamente. Se puede comprobar que en todos los casos se ha logrado la **separabilidad** entre muestras debida a la condición que representa cada muestra experimental.

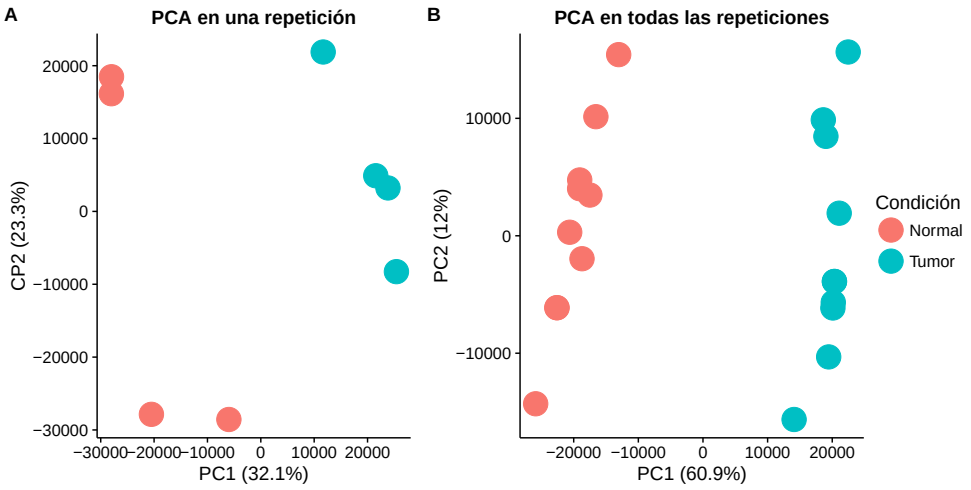

Figura 4.5: Diagrama de dispersión de las primeras dos componentes del *PCA* realizado sobre: **A)** los valores de expresión de las isoformas en una repetición del escenario S1; **B)** los valores medios de expresión de las isoformas en las diez repeticiones de dicho escenario.

El siguiente paso fue determinar la correlación de Spearman (Spearman, 1904) existente entre los valores medios de expresión simulados y los logrados en las distintas repeticiones de cada escenario. Esto se realizó utilizando la función `cor.test` de `R`. Los valores promedio de correlación encontrados fueron 0,914 para la condición C y 0,92 para T en S1, 0,766 (C) y 0,797 (T) en S2 y 0,809 y 0,836 en las condiciones C y T de S3, respectivamente. En todos los casos, se utilizó el test de Wilcoxon para determinar si la correlación entre valores de expresión en la condición C fue diferente a la de los valores de expresión en la condición T. En los tres escenarios se encontraron diferencias significativas (valores p menores a 0,05) entre dichas correlaciones, resultando mayores las respectivas a

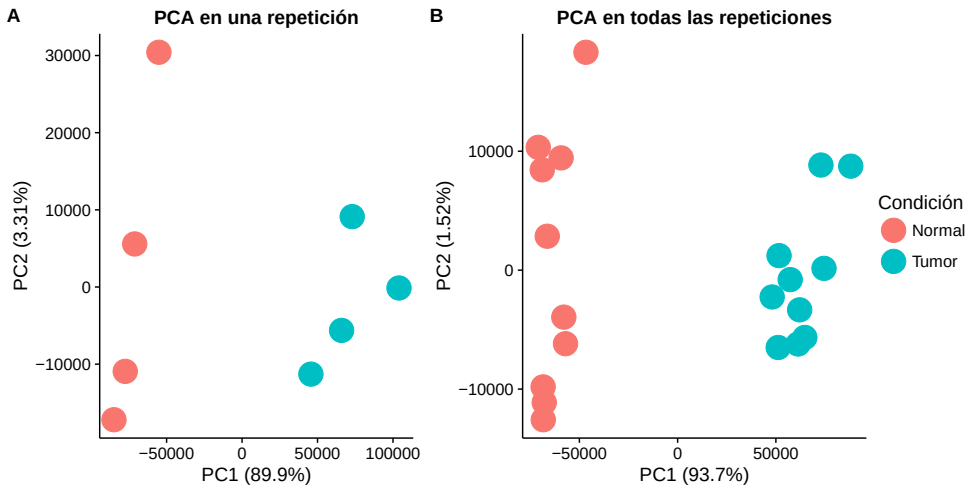

Figura 4.6: Diagrama de dispersión de las primeras dos componentes del *PCA* realizado sobre: **A)** los valores de expresión de las isoformas en una repetición del escenario S2; **B)** los valores medios de expresión de las isoformas en las diez repeticiones de dicho escenario.

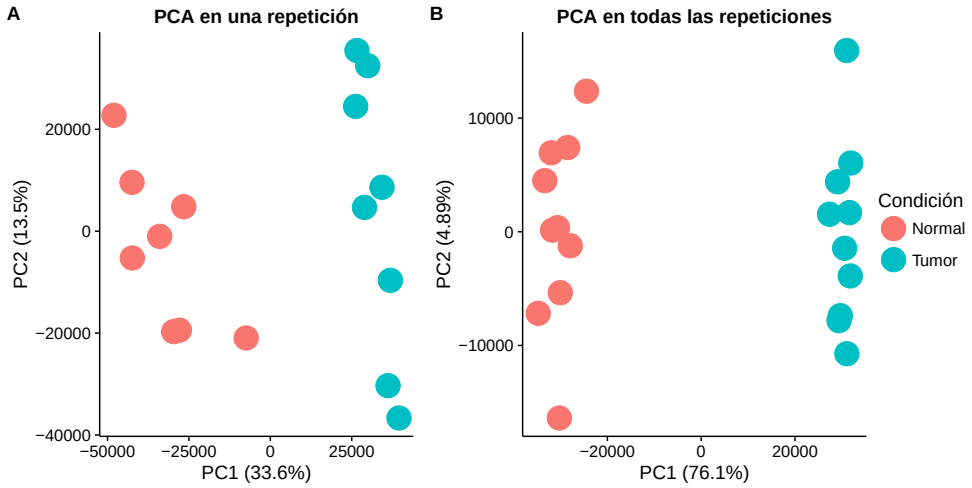

Figura 4.7: Diagrama de dispersión de las primeras dos componentes del *PCA* realizado sobre: **A)** los valores de expresión de las isoformas en una repetición del escenario S3; **B)** los valores medios de expresión de las isoformas en las diez repeticiones de dicho escenario.

la condición T. Posteriormente se evaluó la correlación de los valores de expresión pero sólo de las isoformas simuladas como diferencialmente expresadas. En este caso, los valores medios de correlación encontrados fueron 0,937 y 0,964 para las condiciones C y T, respectivamente, de S1; en el caso de S2 se observaron valores medios de correlación de 0,803 para C y 0,868 para T, mientras que los hallados en S3 fueron 0,834 y 0,891. Al comparar las correlaciones encontradas para todas las isoformas con las halladas sólo para las isoformas diferencialmente expresadas, el test de Wilcoxon reveló diferencias significativas (valores p menores a 0,05) entre ellas, siendo las últimas las de mayor valor en los tres escenarios.

Posteriormente, se exploraron los valores de fold change, en escala log_2, logrados en cada uno de los grupos simulados que involucraron cambios en la expresión absoluta de las isoformas: DIE, DE y DIEDS. Los diagramas de caja de los fold changes encontrados en los escenarios S1, S2 y S3 se muestran en las Figura 4.8, Figura 4.9 y Figura 4.10, respectivamente. En cada una de ellas, dichos diagramas se agruparon en paneles según los grupos de simulación considerados. A su vez, en cada panel, se han incluido líneas a trazos para indicar los valores de fold change deseados en cada uno de los subgrupos de simulación. Mediante el análisis de dichas figuras, se determinó que en los grupos **DIE** y **DE** los fold changes

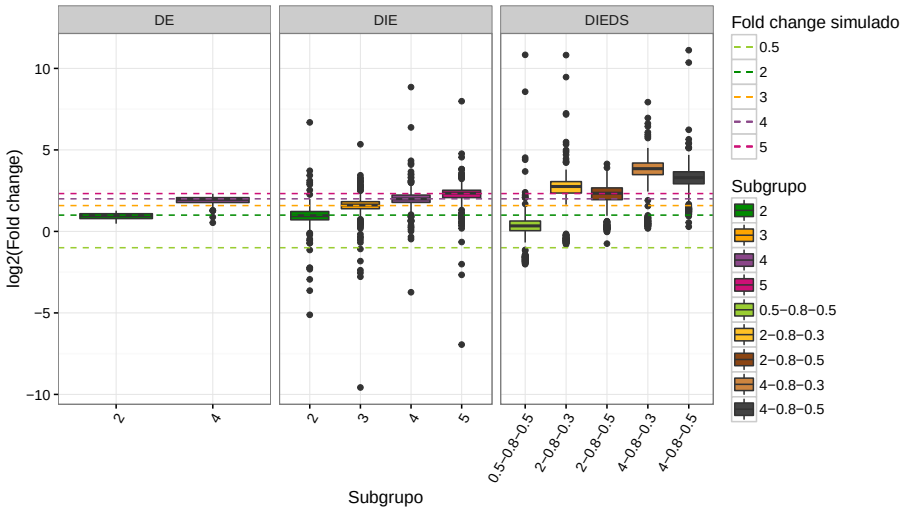

Figura 4.8: Diagramas de caja del fold change, en escala logarítmica en base dos, obtenido en las diez repeticiones del escenario S1. Cada panel representa uno de los tres grupos simulando cambios de expresión absoluta de isoformas y los correspondientes subgrupos. Las líneas de trazos representan los fold changes simulados.

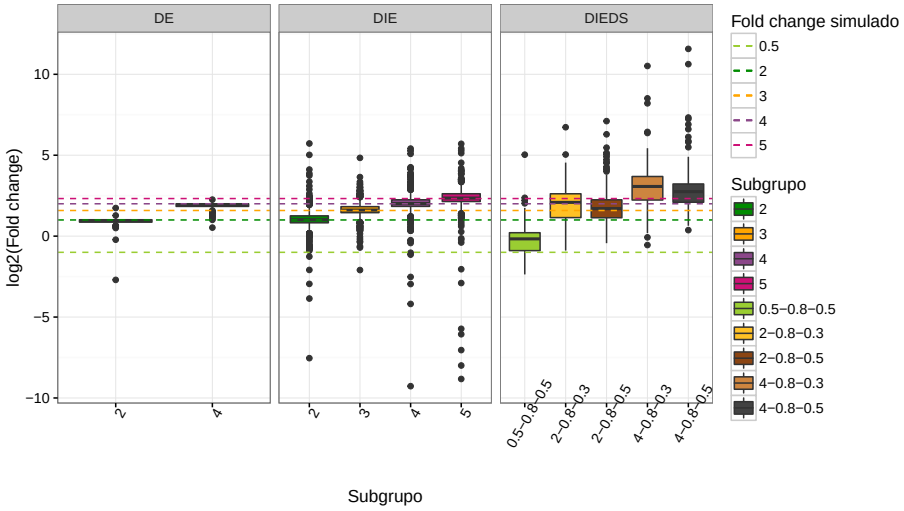

Figura 4.9: Diagramas de caja del fold change, en escala logarítmica en base dos, obtenido en las diez repeticiones del escenario S2. Cada panel representa uno de los tres grupos simulando cambios de expresión absoluta de isoformas y los correspondientes subgrupos. Las líneas de trazos representan los fold changes simulados.

observados en las isoformas a lo largo de las repeticiones de cada escenario se correspondieron con los valores deseados, impuestos por la simulación. En el caso del grupo **DIEDS**, la correspondencia no fue tan clara, probablemente por la influencia del splicing diferencial sobre los cambios de expresión absoluta.

Se evaluaron también los valores de proporción de las isoformas más expresadas de cada gen, logrados en los dos grupos simulados que involucraron cambios en la expresión relativa (**DS** y **DIEDS**). Los diagramas de caja de dichas proporciones para los escenarios S1, S2 y S3 se muestran en las Figura 4.11, Figura 4.12 y Figura 4.13, respectivamente. Dichas figuras contienen un panel para cada grupo de simulación, en el cual se incluyeron líneas a trazos para indicar los valores de proporciones deseados. Además, dada la dependencia de las proporciones con el número de isoformas por gen, se analizaron las proporciones en forma separada. Así, cada una de las figuras contiene tres paneles:**A)** isoformas mayores de los genes del grupo "**2-4**", **B)** isoformas mayores de los genes del grupo "**5-9**" y **C)** isoformas mayores provenientes de genes con más de nueve transcritos anotados. Se determinó que en el caso del escenario S1, las proporciones observadas en las isoformas a lo largo de las repeticiones de cada escenario se correspondieron con los valores deseados impuestos por la simulación, tanto en el grupo DS como el DIEDS. En el caso de los escenarios S2 y S3, las proporciones fueron levemente

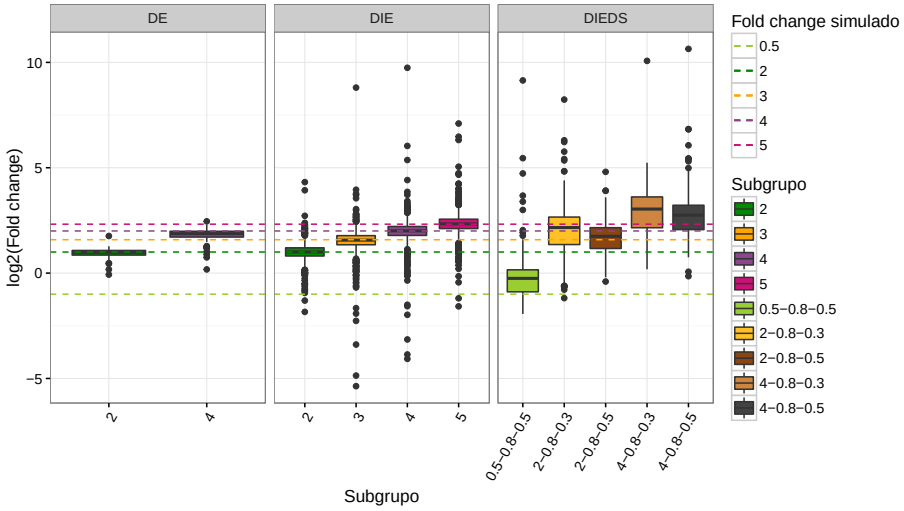

Figura 4.10: Diagramas de caja del fold change, en escala logarítmica en base dos, obtenido en las diez repeticiones del escenario S3. Cada panel representa uno de los tres grupos simulando cambios de expresión absoluta de isoformas y los correspondientes subgrupos. Las líneas de trazos representan los fold changes simulados.

inferiores, acrecentándose este comportamiento para las isoformas provenientes de genes con más de nueve transcritos anotados .

Una vez que se comprobó que la simulación obtenida fue acorde a lo planificado se procedió a la evaluación y comparación de los métodos utilizando las bases de datos simuladas de RNA-seq, cuyos resultados se presentan a continuación.

4.3.2. Concordancia en la detección

Los resultados del análisis de expresión diferencial de todos los *pipelines* en los tres escenarios simulados se analizaron para determinar la cantidad de DI/ASG y la concordancia en la detección. En particular, la concordancia de los resultados se evaluó considerando el porcentaje de DI/ASG y de TPs/FPs encontrados en todas las repeticiones de cada uno de los escenarios. La Tabla 4.3 contiene los resultados obtenidos para los cinco flujos de análisis que estudian DIE. Su análisis determinó que EBSeq detectó el **mayor número** de DI (>8.500) en los tres escenarios evaluados, mientras que Cufflinks fue el flujo de análisis que menos DI halló, siendo sus resultados **tres veces menores** que los de EBSeq. No obstante, este último tuvo la **menor concordancia** en las detecciones de DI en los tres escenarios. Esto se dedujo al analizar su bajo porcentaje de DI encontrados en

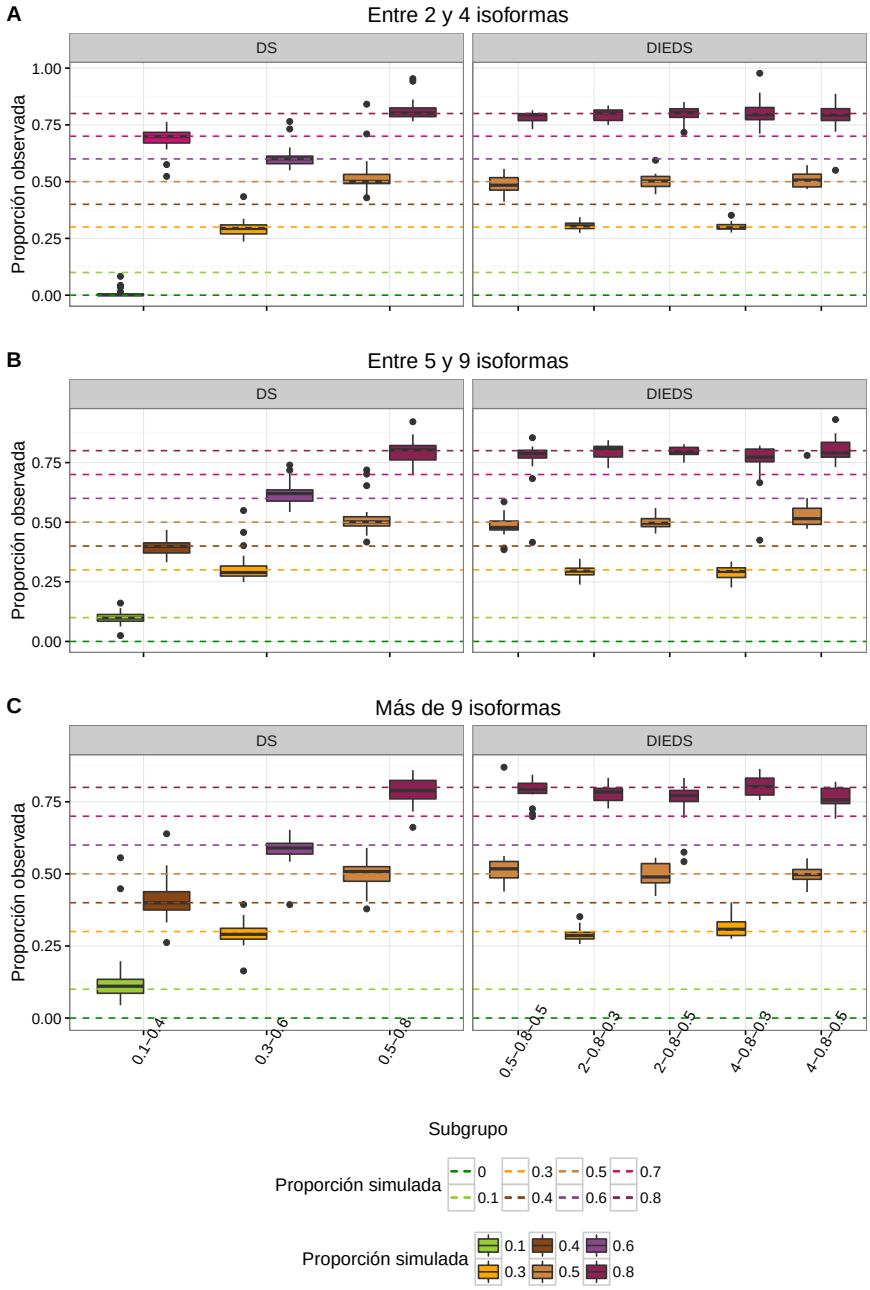

Figura 4.11: Diagramas de cajas de la proporción de las isoformas, sobre las diez repeticiones del escenario S1, separados según el número de isoformas por gen: **A)** Entre dos y cuatro, **B)** Entre cinco y nueve y **C)** Más de nueve. Se muestran los resultados separados en paneles correspondientes al grupo de simulación, en cada uno de los cuales se ha ubicado a los respectivos subgrupos. Las líneas de trazos representan las proporciones simuladas.

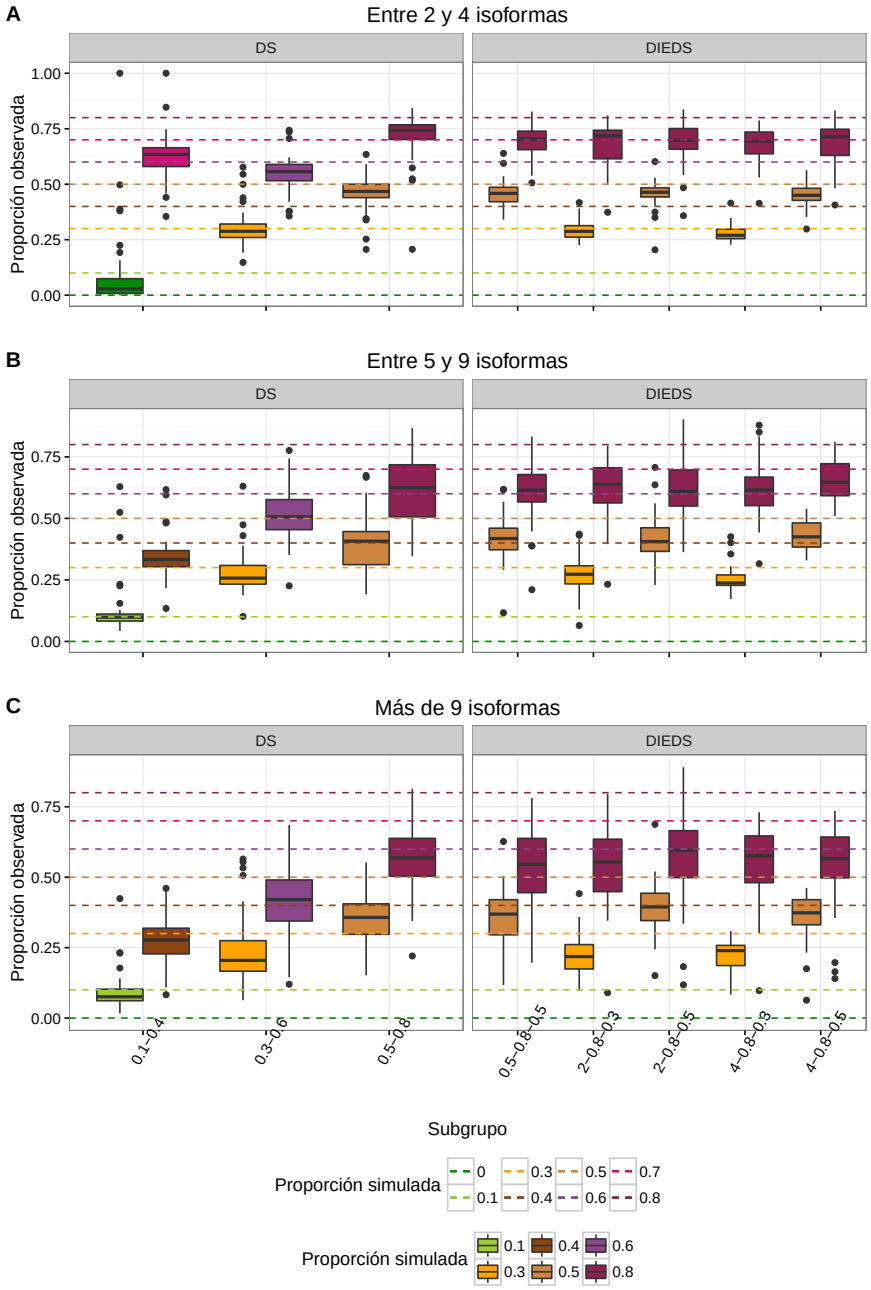

Figura 4.12: Diagramas de cajas de la proporción de las isoformas, sobre las diez repeticiones del escenario S2, separados según el número de isoformas por gen: **A)** Entre dos y cuatro, **B)** Entre cinco y nueve y **C)** Más de nueve. Se muestran los resultados separados en paneles correspondientes al grupo de simulación, en cada uno de los cuales se ha ubicado a los respectivos subgrupos. Las líneas de trazos representan las proporciones simuladas.

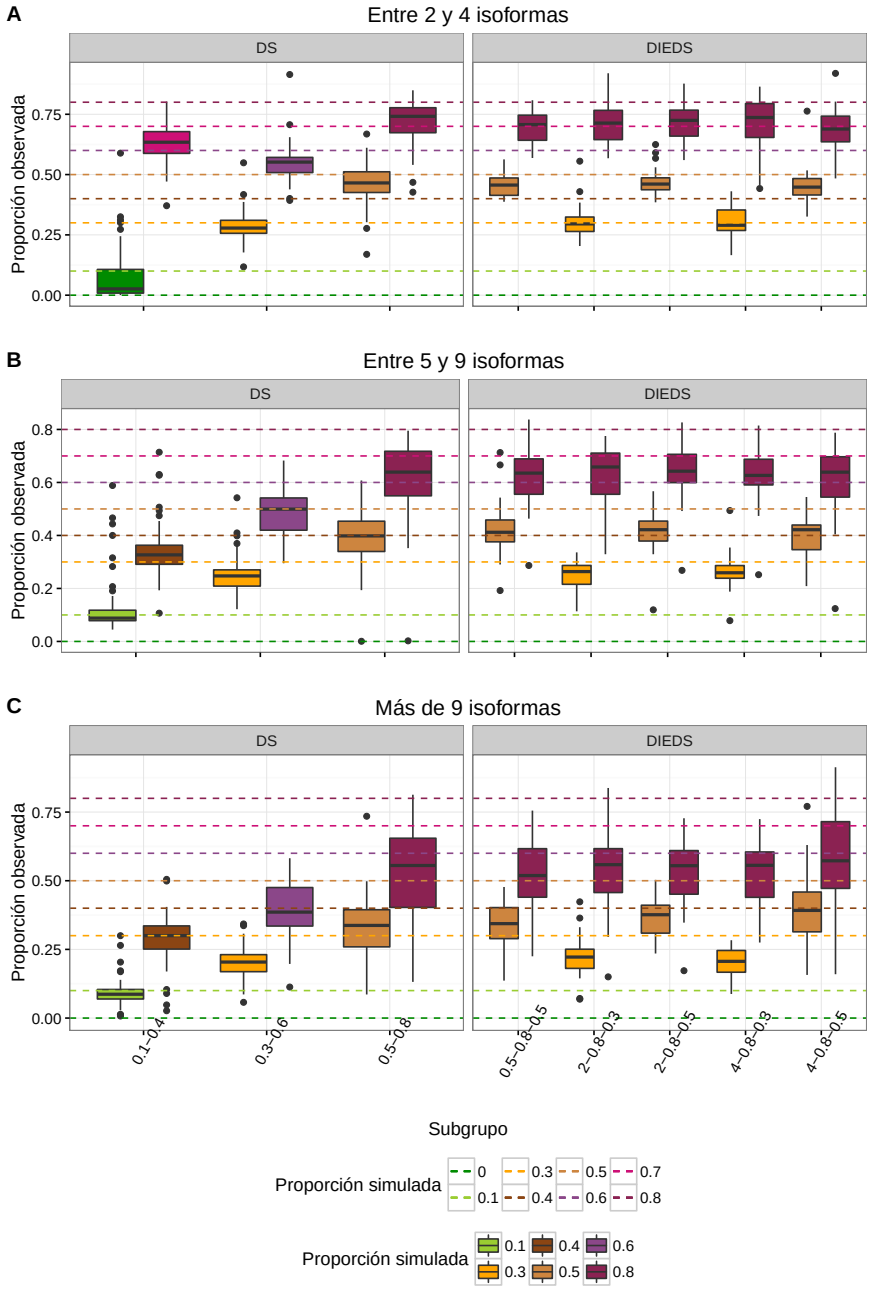

Figura 4.13: Diagramas de cajas de la proporción de las isoformas, sobre las diez repeticiones del escenario S3, separados según el número de isoformas por gen: **A)** Entre dos y cuatro, **B)** Entre cinco y nueve y **C)** Más de nueve. Se muestran los resultados separados en paneles correspondientes al grupo de simulación, en cada uno de los cuales se ha ubicado a los respectivos subgrupos. Las líneas de trazos representan las proporciones simuladas.

todas las repeticiones de un mismo escenario. Por otro lado, DESeq2 y Limma fueron los *pipelines* **más concordantes**, principalmente en S2 y S3, en términos de DI y TP alcanzando porcentajes superiores a 17 % y 30 %, respectivamente. Al comparar los escenarios S1 y S2 se encontró que EBSeq y Cufflinks **no** mostraron diferencias en el porcentaje de TPs concordantes, mientras que DESeq2, Limma y NOISeq **incrementaron** el porcentaje de concordancia de TPs en S2 aproximadamente en un 5 %. Dicho porcentaje resultó **aumentado** en un 10 %, aproximadamente, para EBSeq and DESeq2 y sólo un 1 % para Cufflinks y Limma, en S3 respecto de S2. Consistentemente, todos los *workflows* mostraron un porcentaje de FPs concordantes menor al 5 %, lo que indicó la **efectividad** del proceso de simulación utilizado.

Tabla 4.3: Resultados del análisis de expresión diferencial de isoformas con los cinco *pipelines* evaluados en los tres escenarios simulados.

Variable		*DI medio*	*Total DI*	*Concordancia en DI (%)*	*TP*	*Concordancia en TP (%)*	*FP*	*Concordancia en FP (%)*
EBSeq	S1	2.679	8.852	10,8	3.155	29,6	5.697	0,4
	S2	6.086	16.553	14,6	8.037	29,7	8.516	0,5
	S3	6.305	13.875	23,5	8.155	39,4	5.720	0,7
DESeq2	S1	1.948	4.937	17,7	2.739	31,1	2.198	1
	S2	5.577	13.179	21,1	7.483	36,5	5.696	1,1
	S3	6.054	11.916	29,7	7.770	44,7	4.146	1,5
NOISeq	S1	1.961	4.996	19	2.556	36,6	2.440	0,7
	S2	6.388	15.956	19,7	7.742	40	8.214	0,9
	S3	2.882	4.908	32,4	4.342	36,4	566	1,8
Cufflinks	S1	414	845	25,2	584	35,8	261	1,5
	S2	1.257	2.460	27,2	1.816	36,3	644	1,6
	S3	1.030	1.860	29,9	1.452	37,7	408	1,5
Limma	S1	838	1.468	29,2	1.351	31,4	117	4,3
	S2	3.077	5.451	31,05	4.590	36,2	861	3,8
	S3	5.238	9.927	28,3	7.177	38,5	2.750	1,6

DI, isoformas detectadas como diferencialmente expresadas; TP, verdaderos positivos; FP, falsos positivos

La Tabla 4.4 lista los resultados de concordancia obtenidos en el caso de los *workflows* que analizaron DS. De éstos, se determinó que DEXSeq fue el que encontró, en promedio, **más** cantidad de ASG (más de 423 en todos los escenarios) mientras que CufflinksDS fue consistentemente el que menos genes identificó con DS (menos de 303). En términos del porcentaje de ASG concordantes a lo largo de las diez repeticiones de cada escenario, SplicingCompass fue el de **menor concordancia** ($< 20\,\%$), mientras que LimmaDS mostró los valores más altos, superando en los tres escenarios, el 25 %. Adicionalmente, SplicingCompass y CufflinksDS son los que mostraron menor concordancia en término de TPs, mientras que DEXSeq logró los valores mayores, seguido de LimmaDS. Al comparar los escenarios evaluados, se encontró que todos los *pipelines*, a excepción de Cuf-

flinksDS, **incrementaron** el porcentaje de concordancia de ASG y TP en S3 respecto a S2 y en S2 respecto a S1. En términos de FP, la concordancia resultó mayor que la observada para los flujos DIE, aunque no superó el 10 %.

Tabla 4.4: Resultados del análisis de splicing diferencial utilizando los cuatro *pipelines* evaluados en los tres escenarios simulados.

Variable		*ASG medio*	*Total ASG*	*Concordancia en ASG (%)*	*TP*	*Concordancia en TP (%)*	*FP*	*Concordancia en FP (%)*
CufflinksDS	S1	106	318	11,3	185	19,5	133	0
	S2	303	779	13	527	19,4	252	0,4
	S3	226	468	17,7	437	18,8	31	3,2
DEXSeq	S1	423	1.134	16,4	372	44,9	762	2,5
	S2	913	2.027	22,9	868	46,8	1.159	5
	S3	1.147	2.630	22,8	908	58,7	1.722	3,9
LimmaDS	S1	233	431	26,2	292	34,2	139	9,4
	S2	615	1.060	28,6	723	37,6	337	15,7
	S3	758	1.290	30,7	842	41,4	448	10,5
SplicingCompass	S1	149	536	6,2	234	11,5	302	2
	S2	349	923	9,5	557	13,5	366	1,6
	S3	600	1.256	17,5	714	24,6	542	8,1

ASG, genes detectados con splicing diferencial; TP, verdaderos positivos; FP, falsos positivos

4.3.3. Desempeño general

Las medidas de desempeño listadas en la Tabla 4.2 se calcularon para los flujos de análisis evaluados, en los tres escenarios bajo estudio. Los nueve *pipelines* evidenciaron elevados valores de exactitud, comprendidos en el rango $0{,}868 - 0{,}981$, motivo por el cual esta medida **no** se utilizó con fines comparativos. Los valores medios ($\pm$ desvío estándar) de las medidas sensibilidad, precisión y F-score se ilustran en la Figura 4.14. Específicamente, los paneles **A-C** muestran los resultados de los *pipelines* DIE y los paneles **D-E**, de los flujos DS. En el caso de los flujos DIE, en términos de **sensibilidad** (Figura 4.14**A**), los cinco *pipelines* evaluados alcanzaron valores superiores a 0,65; el valor más alto de sensibilidad promedio fue logrado por EBSeq (en S1 y S3), NOISeq (en S2) y DESeq2 (en S3), mientras que los valores más pequeños fueron los de Cufflinks. En términos de **precisión** (Figura 4.14**B**), el peor desempeño lo tuvieron EBSeq y NOISeq (en S1 y S2), mientras que Limma, Cufflinks (en S2 y S3) y NOISeq (en S3) tuvieron el **mejor desempeño**, logrando valores medios de precisión superiores a 0,9. En este aspecto, se debe mencionar que sólo Limma en S1 y NOISeq en S3 fueron capaces de **controlar el FDR** (1-precisión), logrando valores menores al impuesto (0,05). Al comparar los tres escenarios simulados, se determinó un incremento de S1 a S3 de todas las medidas en todos los *pipelines* a excepción de la sensibilidad

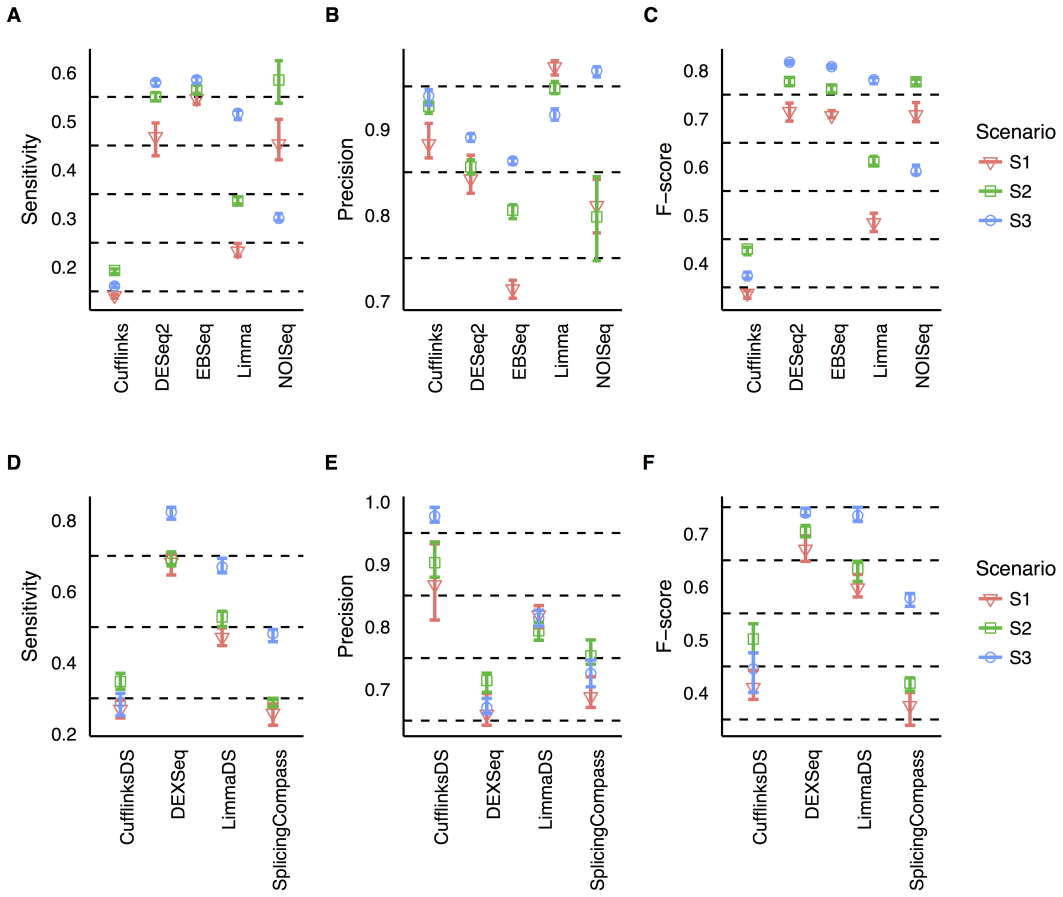

Figura 4.14: Medidas de desempeño global de los nueve flujos de análisis evaluados, obtenidas a lo largo de diez repeticiones de tres escenarios experimentales: S1, S2 y S3. Los paneles **A-C** contienen los diagramas de cajas de dichas medidas para los flujos DIE, mientras que los paneles **D-E** muestran los diagramas de cajas de los *pipelines* DS. Figura extraída de Merino et al. (2017a)

de NOISeq y la precisión de Limma, que evidenciaron comportamientos opuestos. En términos del F-score (Figura 4.14**C**) los valores más elevados (mayores a 0,7) se encontraron para DESeq2, EBSeq, Limma (S3) y NOISeq (S1 y S2), revelando que estos *pipelines* tuvieron el mejor **balance** entre sensibilidad y precisión. Luego, en función de todos los resultados descritos, EBSeq y DESeq2 serían los *pipelines* más **adecuados** para el análisis DIE aunque, si la prioridad es la precisión, Limma y NOISeq serían los más indicados, fundamentalmente cuando el número de muestras es elevado.

En el caso de los flujos de análisis DS, la exploración de la **sensibilidad** promedio (Figura 4.14 **D**) develó que el mejor desempeño fue logrado por DEXSeq y LimmaDS. Éstos alcanzaron valores próximos o superiores a 0,5 en todos los escenarios, mientras que los valores más bajos fueron los obtenidos por CufflinksDS y SplicingCompass. Pese a esto, CufflinksDS mostró los valores más elevados de **precisión** ($> 0{,}8$), en todos los escenarios, incluso **controlando el FDR** en S3 (Figura 4.14 **E**). Aunque DEXSeq fue el que tuvo los menores valores de precisión, inferiores a 0,75, este *workflow* junto con LimmaDS lograron los valores de F-score (Figura 4.14 **F**) más elevados en todos los escenarios, superando a 0,55. Luego, los flujos más **adecuados** para el análisis DS son DEXSeq y LimmaDS, los cuales lograron los mejores valores de sensibilidad y precisión. En particular, LimmaDS logró menor sensibilidad pero mayor precisión que DEXSeq a la hora de reportar ASG.

La habilidad de cada flujo de análisis de manejar los FPs se evaluó mediante el FPR. Los diagramas de cajas de dicha medida correspondientes a los *pipelines* DIE, en los tres escenarios evaluados, se muestran en la Figura 4.15**A-C**, mientras que los correspondientes a los flujos DS se ilustran en la Figura 4.15**D-F**. En el primer tipo de *pipelines*, se destaca que Cufflinks es el flujo de análisis que logró los menores valores de FPR en todos los escenarios. Contrariamente, EBSeq (en los tres escenarios) y NOISeq (en S1 y S2) evidenciaron las tasas más elevadas. En el caso de los flujos DS, los valores más bajos de FPR también se observaron para CufflinksDS, mientras que el peor en términos de esta medida, resultó ser DEXSeq. En general, los valores de FPR no excedieron a 0,05; particularmente, en S2 se encontraron valores superiores a los registrados en S1, mientras que en S3 la mayoría de los *pipelines* DIE tuvieron menor FPR que en S2, excepto Limma. En el caso de los flujos DS, el FPR se incrementó de S1 a S2 y de S2 a S3 para SplicingCompass y DEXSeq, mientras que CufflinksDS resultó el más estable y

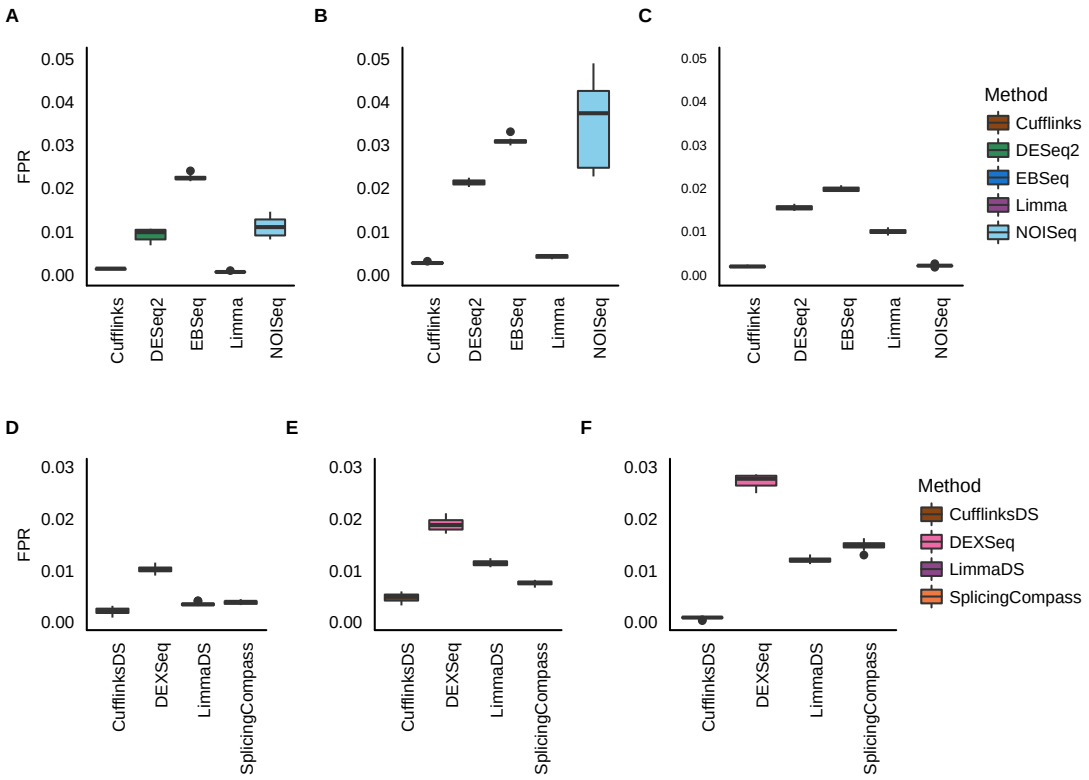

Figura 4.15: Tasa de falsos positivos (FPR) de los nueve *pipelines* evaluados, obtenidas en las diez repeticiones de cada escenario simulado. Los paneles representan el escenario: **A)** S1, **B)** S2 y **C)** S3 para los flujos DIE y **D)** S1, **E)** S2 y **F)** S3 para los *workflows* DS.

LimmaDS no exibió cambios entre S2 y S3.

Las medidas de desempeño obtenidas para cada *pipeline* se **relacionaron** con el nivel de expresión de las isoformas, en el caso de los flujos DIE, y de los genes, para los DS *pipelines* y con su longitud efectiva. Para ello, primero las isoformas/genes fueron agrupados en categorías definidas por el nivel de expresión y luego se calcularon dichas medidas en cada uno de estos grupos. Las Figura 4.16, Figura 4.17 y Figura 4.18 ilustran los valores medios de sensibilidad, precisión y F-score, respectivamente, observados en cada grupo de expresión, promediado a lo largo de las repeticiones de cada escenario simulado. En términos de la relación

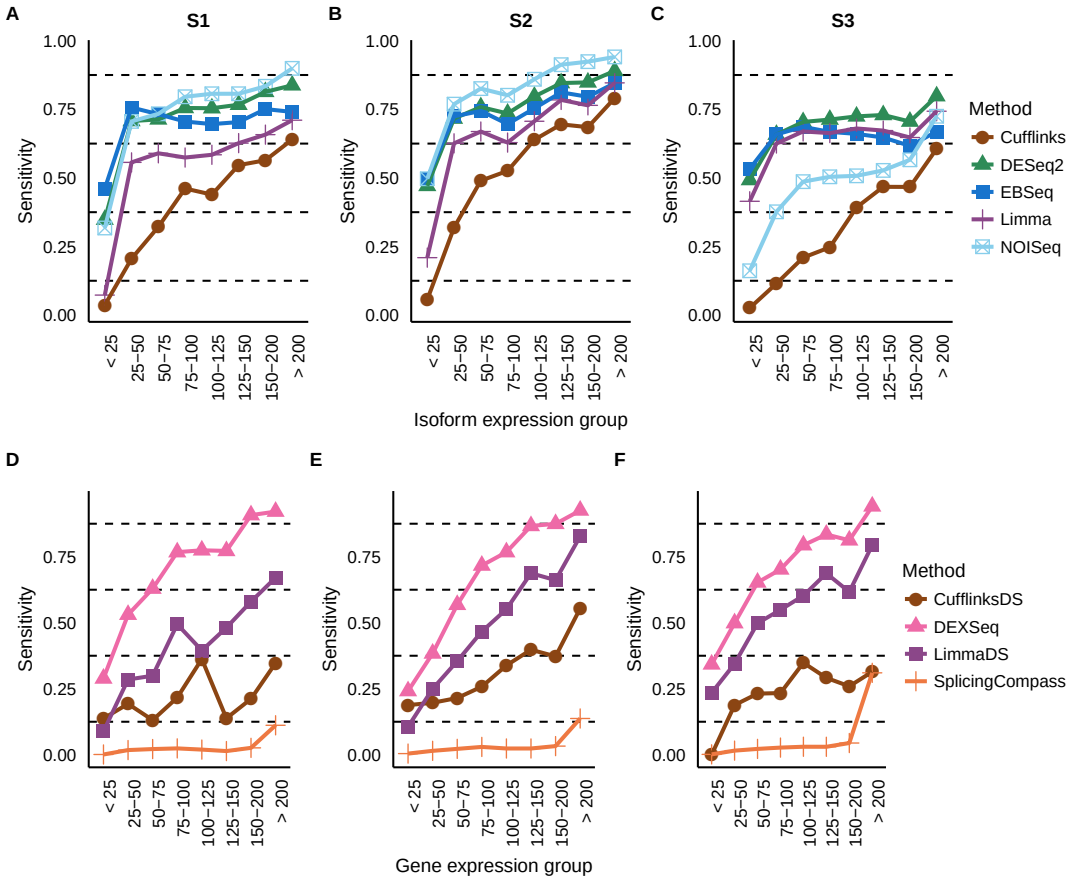

Figura 4.16: Sensibilidad promedio evaluada en cada uno de los grupos de isoformas/genes definidos según el nivel de expresión. Los paneles **A-C** representan los resultados de los flujos DIE, mientras que los paneles **D-F** ilustran los resultados de los *pipelines* DS.

entre la sensibilidad y el valor de expresión se apreció que de los nueve *pipelines* evaluados, SplicingCompass mostró ser el más **robusto**, mostrando prácticamente

los mismos valores de esta medida en todos los intervalos de expresión en todos los escenarios evaluados. El resto de flujos evaluados evidenciaron una correlación positiva entre sensibilidad y valor de expresión, la cual fue más alta para los grupos de menor expresión principalmente en los escenarios S1 y S2.

La correlación entre el valor de expresión y la precisión (Figura 4.17) resultó menor que en el caso anterior, incluso con tendencia a la **independencia** en el escenario

S3 para los flujos DIE (Figura 4.17**C**). Los flujos DS mostraron patrones de

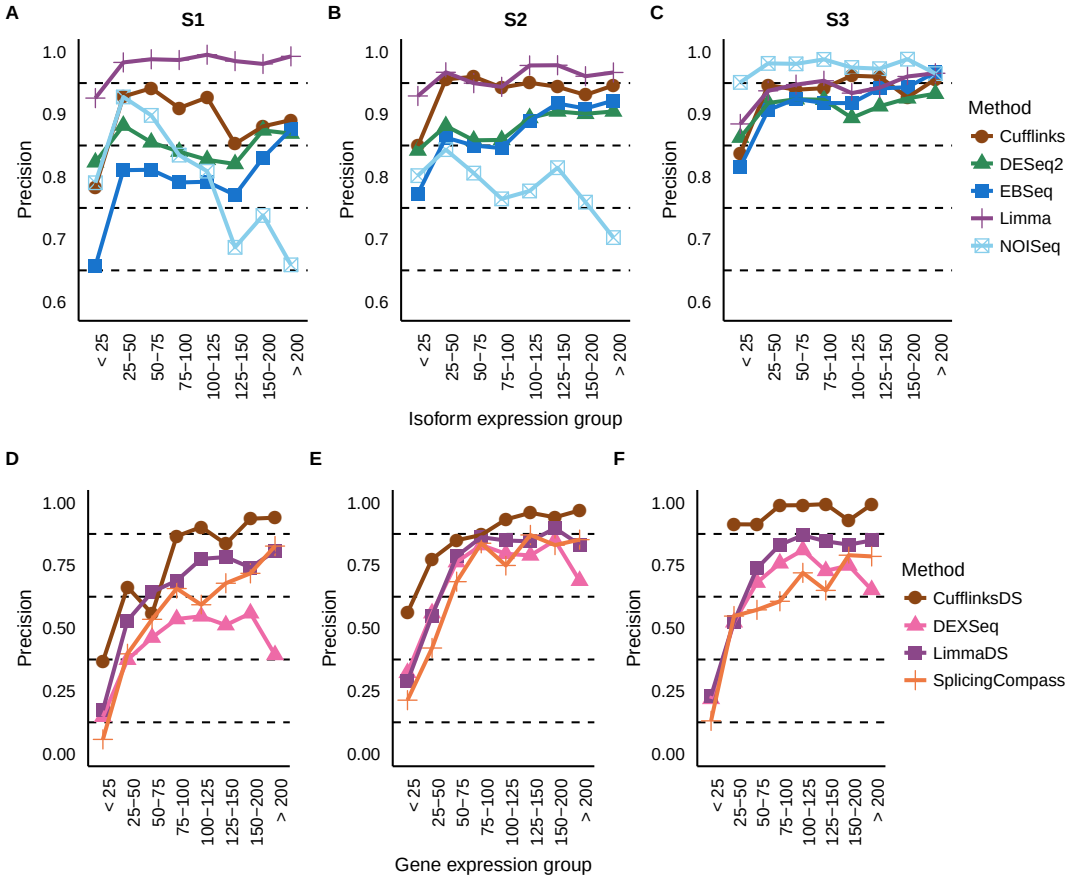

Figura 4.17: Precisión promedio evaluada en cada uno de los grupos de isoformas/genes definidos según el nivel de expresión. Los paneles **A-C** representan los resultados de los flujos DIE, mientras que los paneles **D-F** ilustran los resultados de los *pipelines* DS.

correlación más claros que los DIE, con **aumento** de las medidas de desempeño a medida que **aumentó** el valor de expresión de los genes.

Al analizar el F-score (Figura 4.18), se encontró que en los flujos DIE esta

medida resultó prácticamente **no correlacionada** con el valor de expresión, contrario a lo observado para los *pipelines* DS, los cuales evidenciaron incremento de este indicador con el aumento del valor de expresión.

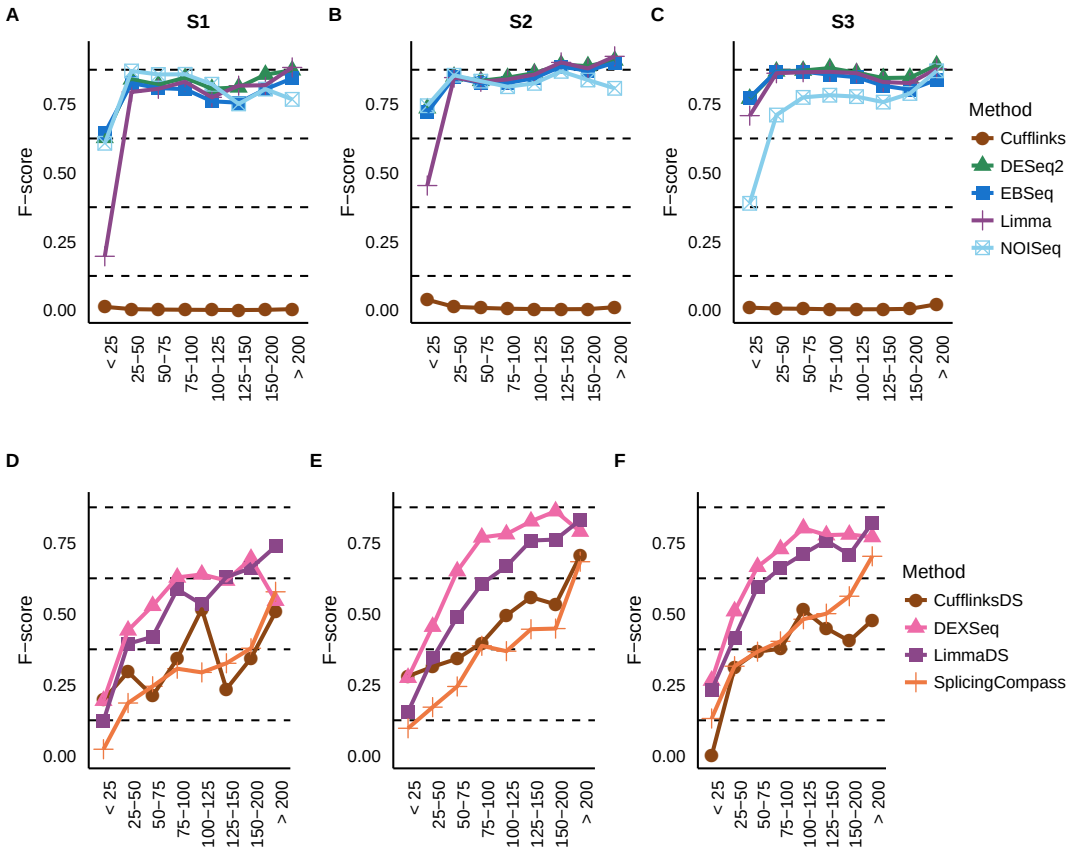

Figura 4.18: F-score promedio evaluada en cada uno de los grupos de isoformas/genes definidos según el nivel de expresión. Los paneles **A-C** representan los resultados de los flujos DIE, mientras que los paneles **D-F** ilustran los resultados de los *pipelines* DS.

Análogamente, se agrupó las isoformas/genes según grupos definidos por su longitud. Las Figura 4.19, Figura 4.20 y Figura 4.21 muestran los valores medios de sensibilidad, precisión y F-score, respectivamente, observados en cada grupo de longitud, promediado a lo largo de las repeticiones de cada escenario simulado. En términos de sensibilidad se encontró que el comportamiento de esta medida respecto de la longitud, fue similar en los escenarios S1 y S2 para todos los flujos evaluados, mientras que los comportamientos tanto de los *pipelines* DIE como DS en S3 resultaron diferentes a los anteriores, evidenciando que los *pipelines* DES-

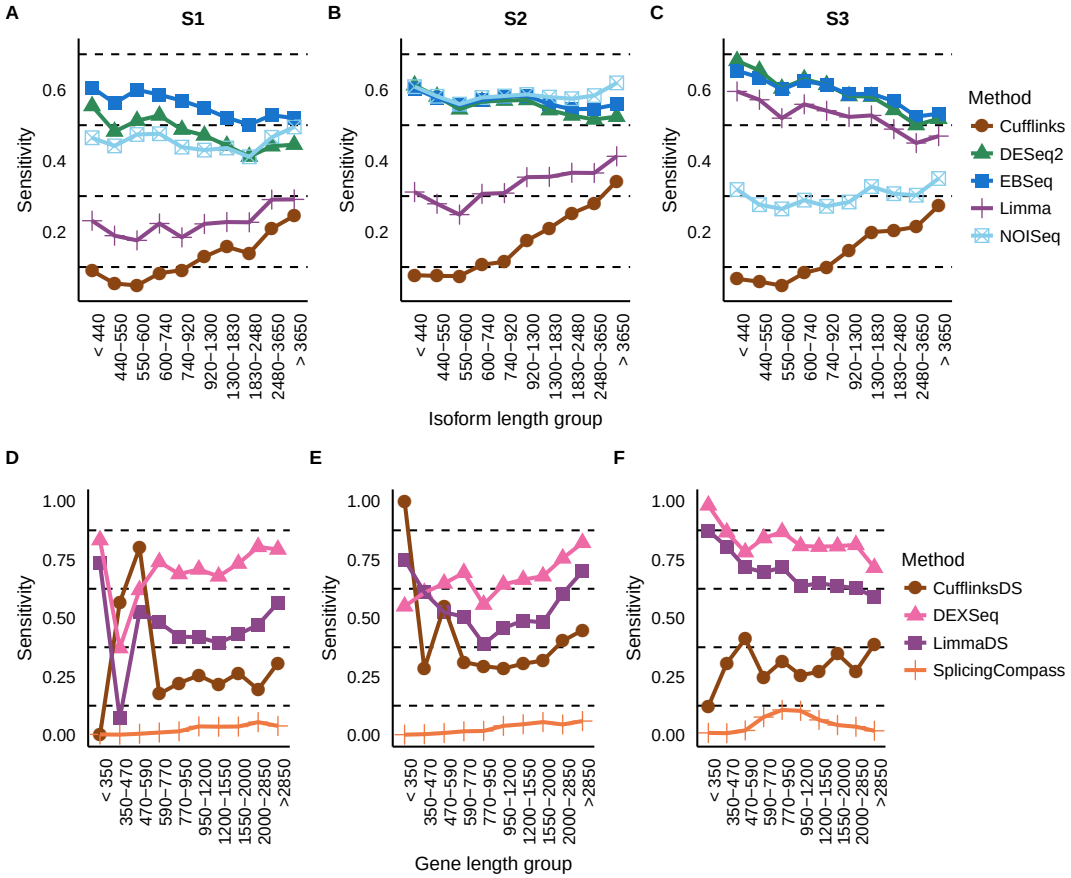

Figura 4.19: Sensibilidad promedio evaluada en cada uno de los grupos de isoformas/genes definidos según la longitud. Los paneles **A-C** representan los resultados de los flujos DIE, mientras que los paneles **D-F** ilustran los resultados de los *pipelines* DS.

eq2, EBSeq, Limma, DEXSeq y LimmaDS tendieron a **disminuir** su sensibilidad a medida que la longitud de las isoformas/genes **aumentó** (correlación negativa), mientras que los pipelines Cufflinks y CufflinksDS evidenciaron **correlación positiva** entre la sensibilidad y la longitud y NOISeq junto con SplicingCompass fueron los más robustos.

La correlación entre la precisión y la longitud de isoformas/genes (Figura 4.20) resultó más evidente en los primeros dos escenarios (S1 y S2) que en el tercero

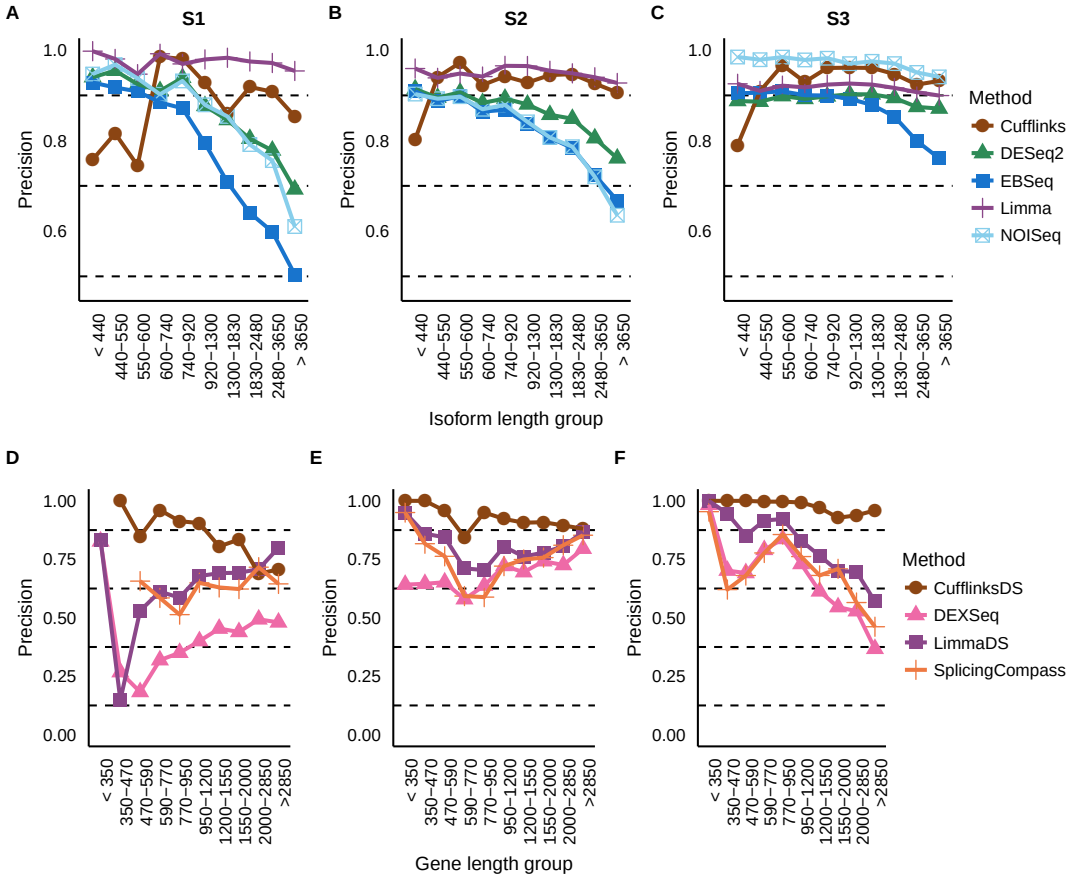

Figura 4.20: Precisión promedio evaluada en cada uno de los grupos de isoformas/genes definidos según la longitud. Los paneles **A-C** representan los resultados de los flujos DIE, mientras que los paneles **D-F** ilustran los resultados de los *pipelines* DS.

(S3). En particular, los flujos basados en las herramientas *Tuxedo*, Cufflinks y CufflinksDS, mostraron la mayor robustez, mostrando precisiones medias prácticamente invariables a lo largo de los distintos grupos evaluados. Cabe destacar que en el escenario S3 casi todos los flujos DIE lograron **estabilidad** de la pre-

cisión con la longitud de las isoformas, mientras que los pipelines DS exhibieron una clara **tendencia de correlación negativa** entre tales cantidades.

Al analizar los valores medios de F-score a lo largo de los grupos de longitud de isoformas/genes (Figura 4.21) se encontró que el comportamiento de los *pipelines*

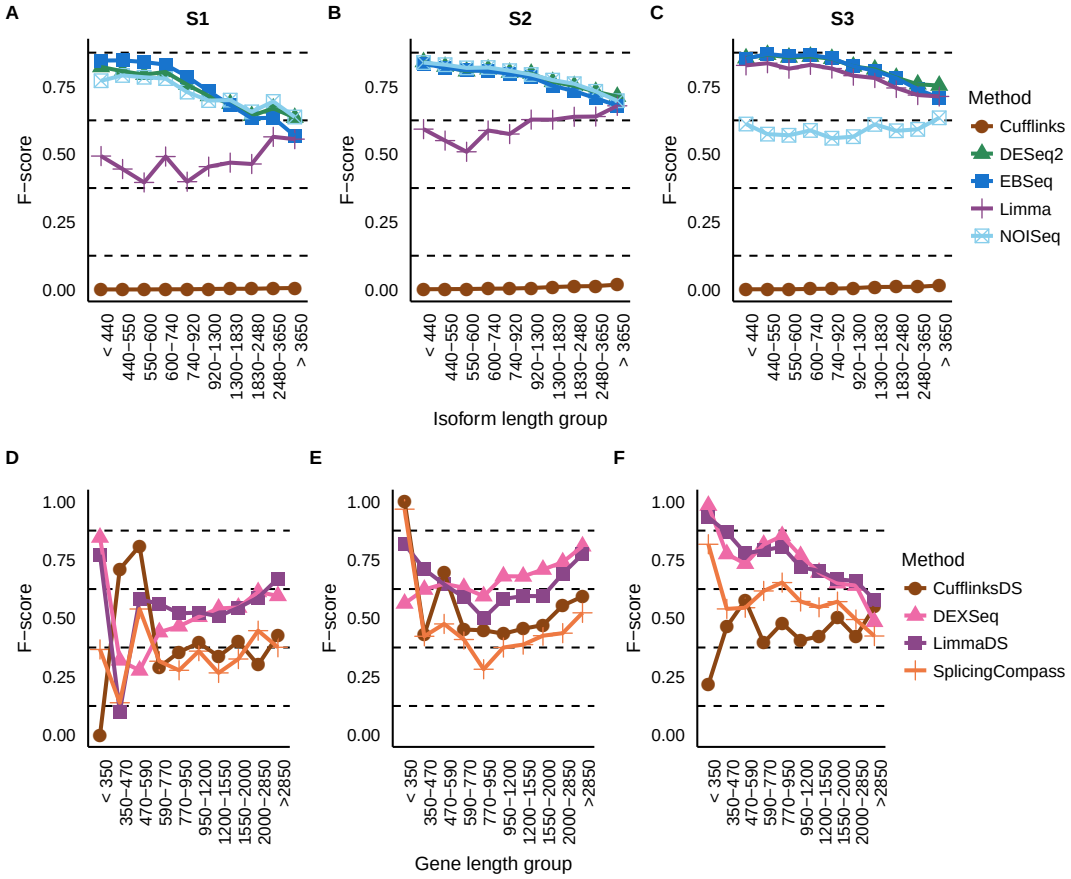

Figura 4.21: F-score promedio evaluada en cada uno de los grupos de isoformas/genes definidos según la longitud. Los paneles **A-C** representan los resultados de los flujos DIE, mientras que los paneles **D-F** ilustran los resultados de los *pipelines* DS.

DIE resultó más **consistente** que el de los flujos DS, probablemente por el hecho de que los primeros comparten, en su mayoría, la herramienta de **cuantificación** (`RSEM`). La estabilidad de los valores medios de F-score a lo largo de los intervalos de longitud resultó mucho más clara para los flujos DIE, más aún en el escenario S3.

En general, los resultados encontrados se pueden resumir en que se determinó que tanto el valor de expresión como la **longitud afectan** más el desempeño de los

pipelines DS que los flujos DIE. Más aún, estos últimos exhibieron valores estables de las medidas de desempeño a lo largo de los distintos grupos, principalmente en S2 y S3. En particular, en el escenario S3 se detectaron los mejores desempeños y la menor dependencia, de las medidas evaluadas, del valor de expresión y de la longitud, tanto en *workflows* DIE como DS. Para las tres medidas evaluadas se determinó que los valores más elevados fueron para aquellos genes/isoformas con **mayor expresión** y **menor longitud**, siendo más evidente este comportamiento en el primer caso.

En base a los resultados analizados en los parrafos anteriores, se determinó que los *pipelines* Cufflinks, EBSeq, CufflinksDS y SplicingCompass fueron los de menor desempeño, por lo que fueron excluidos de los análisis posteriores. De esta manera, el grupo de nueve *workflows* analizados se **redujo** a cinco: DESeq2, Limma y NOISeq, para el análisis DIE, DEXSeq y LimmaDS, para el análisis DS.

4.3.4. Efecto del número de isoformas por gen

La Figura 4.22 ilustra la **relación** encontrada entre el TPR y el *número de isoformas* por gen, para cada flujo evaluado y cada escenario simulado. Los paneles superiores representan los resultados de los *pipelines* DIE, mientras que los inferiores contienen los resultados de los flujos DS. En el primer caso (Figura 4.22**A-C**) se encontró que el TPR resultó mayor para las isoformas provenientes de genes con sólo **un transcrito** anotado (grupo "**1**") y menor para las isoformas provenientes de genes dentro del grupo "**> 9**", en todos los escenarios. Por ejemplo, en el escenario S2 (Figura 4.22**B**), todos los *pipelines* lograron porcentajes superiores a 75 % para las isoformas del grupo "**1**", mientras que el TPR en el grupo "**> 9**" fue menor a 50 %. Se sospecha que este comportamiento ha sido causado por los **bajos** valores de expresión de las isoformas menos expresadas de cada gen y la **complejidad** que representa el problema de reconstrucción de isoformas a partir de lecturas cortas de secuenciación, más aún cuando se incrementa el número de isoformas por gen. DESeq2 y NOISeq mostraron los valores más altos, similares entre sí, de TPRs en S1 (Figura 4.22**A**) y en S2, mientras que DESeq2 y Limma fueron los de mejor desempeño en S3 (Figura 4.22**C**). En términos generales, todos los *pipelines* mejoraron sus TPRs en S2 respecto de S1 y en S3 respecto de S2, a excepción de NOIseq que mostró los menores valores de TPR en el último escenario (S3).

En el caso de los *workflows* DS (Figura 4.22**D-F**), los TPRs observados en

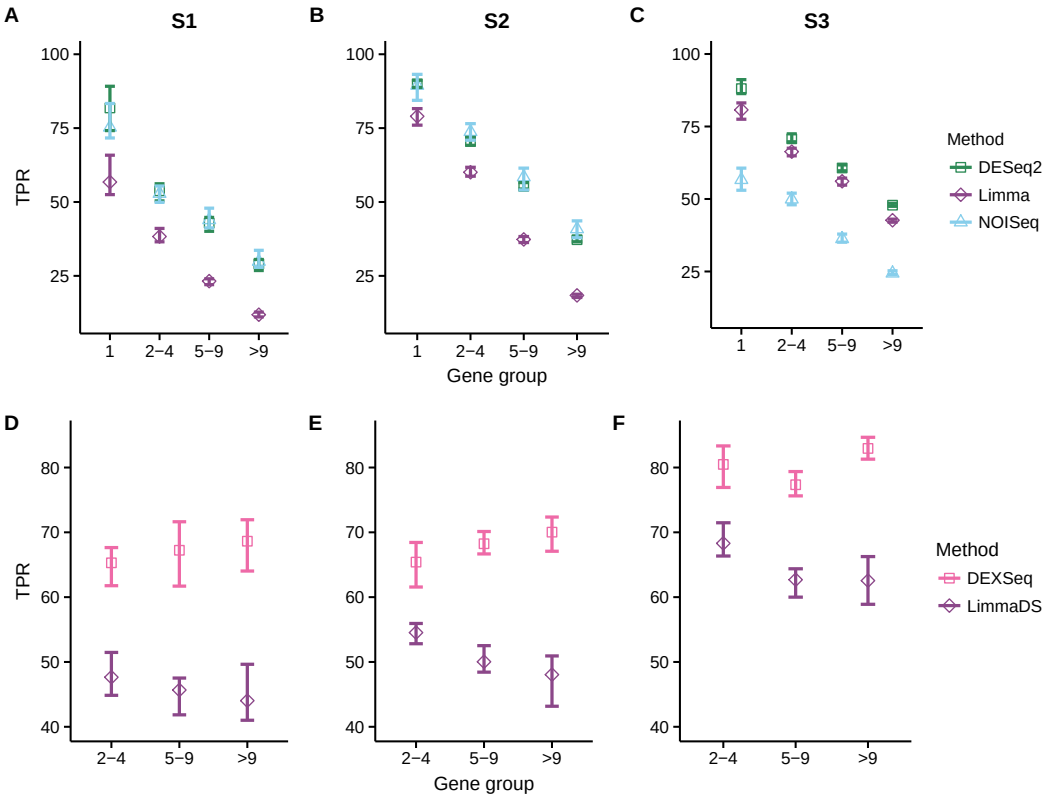

Figura 4.22: Tasa de verdaderos positivos (TPR) de los flujos DIE y DS como función del número de isoformas por gen. Los paneles **A**, **B** y **C** se corresponden con los escenarios S1, S2 y S3 de los flujos DIE, mientras que los paneles **D**, **E** y **F** representan los resultados de los esenarios S1, S2 y S3 para los *pipelines* DS. Figura extraída de Merino et al. (2017a).

todos los escenarios, para cada uno de ellos, resultaron **similares** entre sí y entre los grupos de genes. Los valores más elevados los logró DEXSeq (más de 60 %). LimmaDS exhibió valores de TPRs superiores a 40 %, evidenciando que su desempeño, en términos de esta medida, fue superior a la de su par Limma. En la comparación S1 y S2 no se detectaron cambios en los valores de TPRs, mientras que si determinó que tanto DEXSeq como LimmaDS incrementaron sus porcentajes en S3 respecto de S2.

La distribución de los FP de los *pipelines* evaluados en los grupos de genes definidos y en los tres escenarios evaluados se ilustra en la Figura 4.23, donde se muestran los valores medios (± desvío estándar) de los porcentajes de FP en cada uno de los grupos estudiados. En el caso de los flujos de análisis DIE, mostrados en la Figura 4.23**A-C**, se determinó que la distribución de los FP resultó diferente tanto entre escenarios como entre *pipelines*. DESeq2 y NOISeq

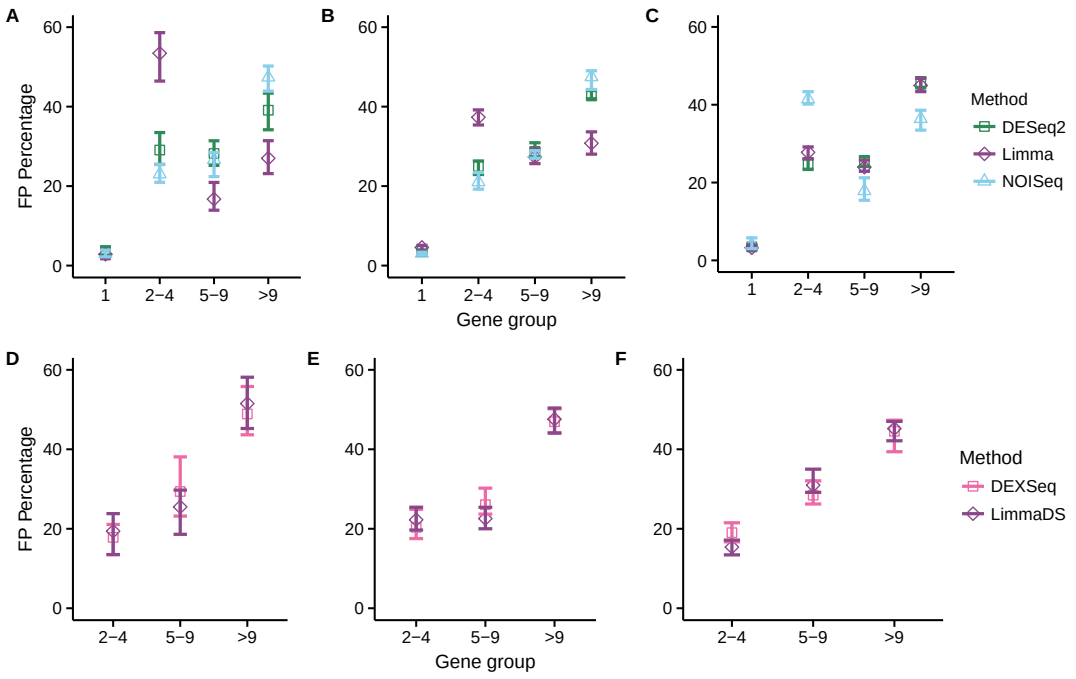

Figura 4.23: Distribución de los falsos positivos (FP) en los grupos de genes, definidos según la cantidad de isoformas anotadas que tiene cada gen. Los paneles **A**, **B** y **C** muestran los resultados para los flujos de análisis DIE y los paneles **D**, **E** y **F** para los *pipelines* DS, en los escenarios S1, S2 y S3, respectivamente.

exhibieron un comportamiento similar en los escenarios S1 y S2, con el porcentaje más alto de FP ($> 35\,\%$) en el grupo "**> 9**". En estos *workflows* se observó que

a medida que el número de isoformas por gen aumentó, también se incrementó el porcentaje de FPs, lo cual es esperable. Por otro lado, Limma evidenció el mayor porcentaje de FPs en el grupo "**2-4**" en S1 y en S2, mientras que, en S3, se comportó igual que DESeq2. Los valores medios de porcentajes FPs de los flujos DS (Figura 4.23**D-F**) resultaron muy similares entre grupos de genes y entre escenarios, en los dos *pipelines* evaluados. En general, los valores medios estuvieron cerca del 20 %, 25 % y 45 % para los grupos "**2-4**", "**5-9**" y "**> 9**", respectivamente. Tanto para DEXSeq como para LimmaDS se determinó que el porcentaje de FPs se **incrementó** con el aumento del número de isoformas por gen.

4.3.5. Efecto del nivel de cambio en la expresión

Finalmente, se evaluó el desempeño de cada uno de los flujos de análisis en términos del **tipo** y **magnitud** de cambio en la expresión, según los **subgrupos** definidos en la Tabla 4.1. Para ello se determinó el valor medio del TPR en cada uno de dichos subgrupos, en los tres escenarios simulados. Los resultados para los *pipelines* DIE se ilustran en la Figura 4.24**A-C**. Como es de esperar, estos flujos de análisis mostraron **incrementos** en el TPR a medida que la magnitud del cambio en la expresión absoluta de las isoformas fue **superior**, aunque se encontraron diferencias entre los grupos **DE**, **DIE** y **DIEDS**. Si bien todos los *pipelines* exhibieron los mayores TPRs en el subgrupo **DE-4**, en todos los escenarios, cuando el fold change simulado fue "2", se encontraron importantes diferencias. Específicamente, Limma mostró los valores de TPRs más bajos en S1 para los subgrupos **DE-2** y **DIE-2**, mientras que NOISeq fue el de peor desempeño en S3 en dichos subgrupos. Un comportamiento que se detectó en todos los flujos de análisis y que resultó sorprendente es que todos ellos evidenciaron **menores** valores de TPRs en los subgrupos **DIE** comparados con los **DE**, incluso cuando los fold changes simulados fueron los mismos. Por ejemplo, en el caso del subgrupo **DE-4** los TPRs medios resultaron casi perfectos (cercanos a 100 %), mientras que estos valores cayeron casi en un 50 % en el subgrupo **DIE-4**, en todos los escenarios. Este comportamiento puede ser causado por el hecho de que todas las isoformas de los genes, es decir las de baja y alta expresión, fueron consideradas en la simulación. Por ello la **dificultad** en detectar estas isoformas de baja expresión podría haber contribuido a un menor TPR en los grupos **DIE**. Cuando se analizó los subgrupos con cambios en expresión absoluta

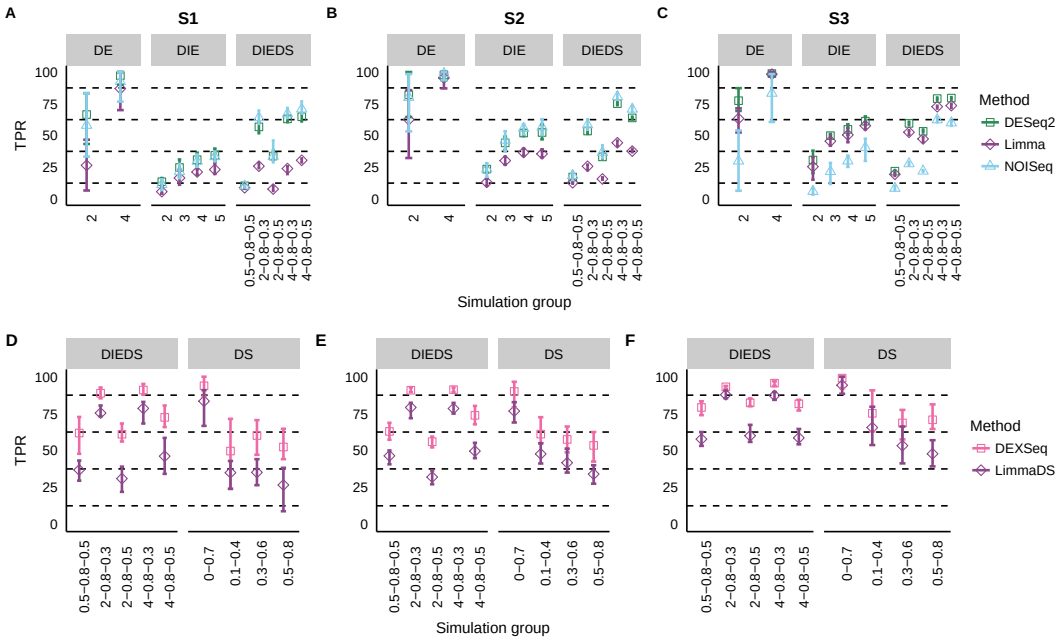

Figura 4.24: Valores medios (± desvío estándar) de la tasa de verdaderos positivos (TPR) obtenida con los *pipelines* evaluados en cada uno de los subgrupos de simulación definidos según la magnitud y el tipo de cambio en la expresión. Los paneles **A**, **B** y **C** corresponden a flujos DIE, mientras que los paneles **D**, **E**, y **F** representan *workflows* DS en los escenarios S1, S2 y S3, respectivamente. Figura extraída de Merino et al. (2017a).

y relativa de las isoformas, grupo **DIEDS**, se determinó que los TPRs medios superaron a los encontrados en los subgrupos **DIE**, ante el mismo fold change, fundamentalmente para Limma y DESeq2 en S1. Por ejemplo, el TPR de estos flujos en el grupo **DIE-2** de S1 resultó inferior a 25 %, mientras que los subgrupos **DIEDS-2-0,8-0,3** y **DIEDS-2-0,8-0,5** alcanzaron TPRs medios superiores a 50 % y 25 %, respectivamente. Este comportamiento se vió sólo para fold change de "4" en S2 y prácticamente no existió en S3. En consistencia con los resultados encontrados anteriormente, los peores desempeños de Limma se detectaron en S1 y S2, mientras que para NOISeq se encontraron en S3. Por su parte, DESeq2 fue el más estable a lo largo de los escenarios.

Los valores medios de TPRs en función de los subgrupos simulados para evaluar los *workflows* DS se muestran en la Figura 4.24**D-F**. En este caso, los resultados mostraron comportamientos más **esperables** y **consistentes** entre escenarios. Como regla general, se encontró que DEXSeq tuvo **mejor** desempeño que LimmaDS en todos los subgrupos de simulación y en todos los escenarios. Los valores de TPRs más elevados es encontraron para los grupos en los cuales la diferencia de proporciones de la isoforma mayor fue más amplia: **DS-0-0,7**, **DIEDS-2-0,8-0,3** y **DIEDS-4-0,8-0,3**. Los mejores desempeños en el grupo **DIEDS** fueron superiores a 75 % y estuvieron asociados a las diferencias de expresión relativa más que a las diferencias en el fold change, ya que los TPRs en los subgrupos **DIEDS-0,5-0,8-0,5**, **DIEDS-2-0,8-0,5** y **DIEDS-4-0,8-0,5** fueron muy similares entre sí e inferiores a los observados para los otros dos subgrupos **DIEDS**. En el caso de genes simulados sólo con cambio en la expresión relativa de sus isoformas, grupo **DS**, los mejores resultados se encontraron cuando el gen estuvo silenciado (sin expresión) en una condición y cuando la expresión relativa de la isoforma mayor fue baja: **DS-0-0,7** y **DS-0,1-0,4**, respectivamente. Adicionalmente, el desempeño de LimmaDS y DEXSeq aumentó de S2 a S1 y de S3 a S2, logrando incluso en el tercer escenario, valores de TPR medios superiores a 50 % en todos los casos.

4.3.6. Aplicación sobre datos reales

La base de datos real completa, 16 muestras con ocho réplicas por condición, se analizó con los nueve flujos aquí evaluados. El **número de DI** y **ASG reportados** por cada uno de estos flujos, así como también el **solapamiento** entre las detecciones, se resumen en la Tabla 4.5. Se encontraron grandes **diferencias**

en el número de isoformas/genes detectados como diferenciales por cada uno de los flujos evaluados, tal y como lo reportó anteriormente Liu et al., 2014. El mayor número de isoformas diferencialmete expresadas fue reportado para EBSeq (4.600), mientras que el *pipeline* que menos DI detectó fue NOISeq, reportando sólo 11, seguido de Cufflinks (204) y Limma (342). Por otro lado, de los *workflows* DS, DEXSeq fue el que detectó más genes con splicing diferencial (1.727), mientras que CufflinksDS no reportó ASG. Las tendencias observadas en los datos reales son consistentes con las descritas anteriormente con los datos simulados. El solapamiento entre las detecciones de los *workflows* develó que la mayoría de las DI detectadas por DESeq2, Limma y NOISeq fueron también encontradas por EBSeq, lo que sugiere que estos flujos capturan la misma estructura de datos pero con diferentes limitaciones. Contrariamente, Cufflinks mostró muy poco solapamiento de DI con el resto de flujos DIE. Por otro lado, los flujos que analizan DS resultaron más **discrepantes** entre sí. Por un lado, SplicingCompass evidenció solapamientos de entre el 8 % y 38 % con los otros *pipelines*, mientras que el 80 % de los ASG encontrados por LimmaDS también fueron reportados por DEXSeq. Esto refleja la importancia que requiere la selección del *pipeline* de análisis para DS.

Tabla 4.5: Cantidad de isoformas/genes reportados como diferencialmente expresados y solapamiento entre las detecciones de los nueve flujos de análisis evaluados. Tabla extraída de Merino et al. (2017a).

Tipo de detección	*Flujos de análisis*					
DIE		*Cufflinks*	*DESeq2*	*EBSeq*	*Limma*	*NOISeq*
	Cufflinks	204	92	90	19	1
	DESeq2		1.572	1.269	281	10
	EBSeq			4.600	324	11
	Limma				342	10
	NOISeq					11
DS		*CufflinksDS*	*DEXSeq*	*LimmaDS*	*SplicingCompass*	
	CufflinksDS	0	0	0	0	
	DEXSeq		1.727	82	35	
	LimmaDS			103	7	
	SplicingCompass				91	

4.4. Conclusiones y propuesta

El estudio presentado en este capítulo **evaluó** nueve flujos de procesamiento de datos de RNA-seq, cinco de los cuales se utilizan en el análisis de cambios de expresión absoluta de isoformas, mientras que los cuatro restantes se ocupan para determinar la ocurrencia de splicing diferencial de genes. La comparación presentada se basó en conjuntos de datos **sintéticos**, donde los perfiles de expresión de genes e isoformas se controlaron para simular los tipos de cambio anteriormente mencionados y así poder evaluar objetivamente el desempeño de los flujos de análisis. Se consideraron tres **escenarios experimentales** con diseño balanceado, involucrando pocas (cuatro) o muchas (ocho) réplicas por condición. Como referencia se utilizó un experimento real de RNA-seq con variabilidad del tipo paciente cáncer-sano, del cual se extrajeron tanto parámetros necesarios para la simulación de las lecturas de secuenciación como para la simulación de cambios reales de expresión.

En términos generales, el estudio permitió identificar que de los escenarios experimentales evaluados, el **indicado** fue aquél donde el porcentaje de genes con cambios en la expresión fue superior (10 % vs 5 %) y donde el número de réplicas por condición fue mayor (8 vs 4), es decir S3. Para esta configuración experimental se encontraron los números más altos de isoformas diferencialmente expresadas detectadas, genes detectados con splicing diferencial, TPs y concordancia en las diez repeticiones de dicho escenario. Los flujos de mejor **desempeño** fueron DESeq2, Limma y NOISeq, para análisis DIE, mientras que DEXSeq y LimmaDS se destacaron en el análisis DS.

Sensibilidad, precisión y F-score se utilizaron como indicadores de **desempeño**. Para experimentos con **bajo número de réplicas** los mejores flujos de análisis DIE fueron DESeq2 y Limma. Éstos además mostraron estabilidad de dichas medidas de desempeño cuando se analizaron distintos grupos de nivel de expresión y longitud de las isoformas. En base a los resultados obtenidos, se concluyó que, si se desea obtener elevada **sensibilidad**, entonces el *pipeline* adecuado es DESeq2, mientras que Limma debe utilizarse si se desea controlar la **precisión** en la detección. Asimismo, cuando el número de réplicas es superior, NOISeq es un flujo más restrictivo y preciso que Limma. En el caso de los *pipelines* DS, se encontró que DEXSeq fue el mejor en términos de **sensibilidad** y **F-score**, aunque, en términos de **precisión**, LimmaDS fue superior. Incluso, en el escenario donde el número de réplicas fue superior, LimmaDS alcanzó los valores de

F-score de DEXSeq. El desempeño de los flujos DS resultó más influenciados por el valor de expresión y la longitud de los genes. En particular, se encontraron mejores resultados para genes con más de 50 conteos y más de 470 nucléotidos de longitud efectiva. En base a los resultados obtenidos se **concluyó** que tanto DEXSeq como LimmaDS son indicados para analizar DS, el primero de éstos priorizando **sensibilidad** y el segundo, **precisión**. En adición, la evaluación del FPR determinó que Limma y LimmaDS fueron superiores a DESeq2 y DEXSeq, respectivamente.

La evaluación del efecto del **número de isoformas por gen** sobre el desempeño de los flujos de análisis reveló que los *pipelines* DIE resultaron más influenciados por este efecto que los flujos DS, probablemente por la presencia de isoformas de baja expresión. Específicamente, el TPR para los flujos DIE mostró una correlación negativa con el número de isoformas por gen. En particular, TPRs entre 30 % y 90 % cuando se utilizaron cuatro réplicas por condición para DESeq2 y NOISeq, mientras que este último sólo alcanzó valores entre 25 % y 60 % cuando se consideraron más réplicas por condición. Limma, por su parte, logró valores similares a DESeq2 en este último caso. En el caso de DEXSeq y LimmaDS se encontró TPRs cercanos a 40 % con un número bajo de réplicas, mientras que estos valores se incrementaron al 60 % cuando se utilizaron más réplicas. Adicionalmente, se encontró que la mayoría de los FPs provinieron de genes más de nueve isoformas anotadas en la referencia.

Por último, la exploración del efecto de la **magnitud** del cambio en la expresión diferencial sobre el TPR develó que dicha tasa fue superior cuando el cambio en la expresión absoluta fue mayor, con o sin la presencia de cambios en la expresión relativa de las isoformas. En el caso de DESEq2 y NOISeq se encontró que los porcentajes de detección de TPs se mejoró cuando el porcentaje de genes diferenciales simulados fue superior. Esto sugiere que dichos *pipelines* serían indicados cuando la expresión diferencial afecta un conjunto de transcritos cercanos al 10 % del tamaño total del transcriptoma. Mientras tanto, el mejor desempeño de Limma se asoció con mayor número de réplicas. En el caso de los flujos DS, los TPRs más elevados se encontraron para DEXSeq. Tanto éste como LimmaDS encontraron un alto porcentaje de los genes simulados con cambios en el splicing combinados o no con cambios de expresión absoluta. En este último caso, los porcentajes de detección fueron levemente superiores, más aún en la detección de los genes cuya diferencia de expresión relativa de su isoforma mayor

fue superior.

Finalmente, y en base a los resultados obtenidos, se **sugiere** que si el número de réplicas del experimento es bajo, los *pipelines* basados en el paquete `R Limma` son los indicados para analizar tanto expresión diferencial absoluta de isoformas como splicing diferencial, con elevada precisión. Si se espera o desea tener el mayor número de isoformas/genes diferenciales, es decir priorizar la sensibilidad, se recomienda entonces el uso de los flujos de analisis DESeq2 y DEXSeq. Si por el contrario, el número de réplicas por condición es elevado, entonces se recomienda utilizar NOISeq para el análisis DIE y DEXSeq o LimmaDS para el análisis DS, dependiendo de si se priorizará sensibilidad o precisión, respectivamente. En base a estas sugerencias se diseñó un **esquema** de *selección de flujo de análisis*, que se muestra en la Figura 4.25.

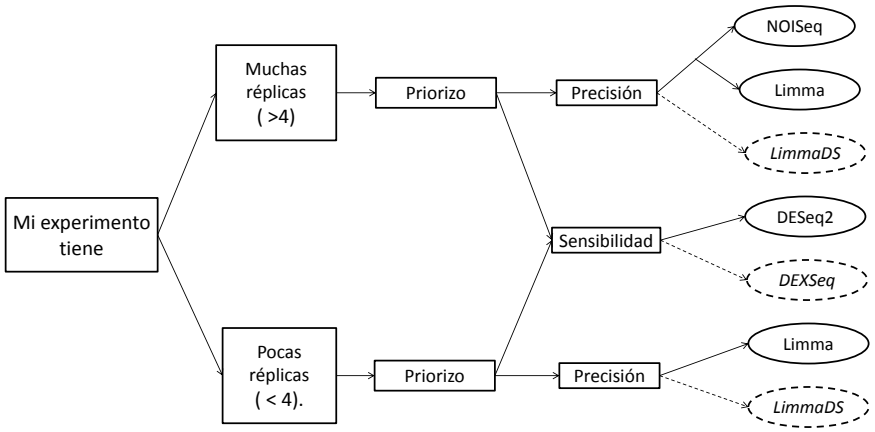

Figura 4.25: Esquema de selección de flujo de análisis. Cada círculo contiene los flujos recomendados según las necesidades del grupo investigador. Las líneas enteras denotan flujos de análisis de expresión diferencial absoluta de isoformas y las líneas entrecortadas flujos de análisis de splicing diferencial. Figura modificada de Merino et al. (2017a).

Capítulo 5

Análisis de Splicing Diferencial

5.1. Introducción

El estudio del SA y sus alteraciones se lleva a cabo mediante la aplicación de **métodos estadístico computacionales** que se nutren de información de expresión de genes, isoformas y exones (Alamancos et al., 2014). A la fecha, se han desarrollado un gran número de métodos para **evaluar el DS** (Hooper, 2014; Wang et al., 2015). Sin embargo, la *complejidad* del proceso de cuantificación de isoformas ha llevado a que la mayoría de ellos e incluso los más utilizados, realicen la inferencia de forma **indirecta** a través de la detección del *uso diferencial de exones* (Alamancos et al., 2014; Anders et al., 2012; Ritchie et al., 2015). En este sentido, si bien esta estrategia permite inferir los genes que han evidenciado DS, la **identidad** de las isoformas afectadas por dicha modificación permanece **oculta** ya que no se sabe que isoformas han cambiado su expresión relativa. Adicionalmente, los métodos típicos de análisis de DS **no** estiman las magnitudes del cambio en el SA. Estos dos aspectos, que en la actualidad se están dejando de lado, resultan fundamentales para comprender el **impacto biológico funcional** del DS. Si bien hasta hace unos años la cuantificación de isoformas resultaba muy **tediosa**, recientemente se han desarrollado herramientas que han evidenciado elevada **precisión** a la hora de llevar a cabo dicha tarea (Bray et al., 2016; Li and Dewey, 2011). Pese a esto, los métodos actuales de análisis de DS **no** han sido modificados para admitir este tipo de datos.

En este capítulo se presenta `NBSplice`, un algoritmo desarrollado para la **detección de DS** a partir de la **cuantificación de isoformas** en experimentos de RNA-seq. Esta herramienta toma como punto de partida los enfoques

comúnmente utilizados y basados en GLMs para inferir diferencias significativas en las **proporciones** de cada una de las isoformas expresadas de un gen, entre dos condiciones experimentales, por ejemplo, caso-control. `NBSplice` ha sido **evaluada** sobre un conjunto de datos simulados de RNA-seq, donde se ha controlado la magnitud y el tipo de cambio en el SA para simular situaciones de DS. La herramienta se ha **comparado** con los programas más utilizados de análisis de DS, evidenciando mejor desempeño al mismo tiempo que permite conocer tanto la identidad de las isoformas afectadas por el DS, como las proporciones de cada una de ellas en cada condición experimental.

El trabajo desarrollado ha sido presentado en el *XXI Congreso Argentino de Bioingeniería*, realizado los días 25, 26 y 27 de octubre del año 2017 en Córdoba, Argentina (Merino and Fernández, 2017).

5.2. Materiales y métodos

5.2.1. NBSplice

Fundamento

El método de detección de DS desarrollado, `NBSplice`, se basa en el uso de **GLMs** para modelar los conteos que representan la expresión de las isoformas. Este enfoque no es **novedoso**, sino que ha sido ampliamente utilizado en el análisis de datos de RNA-seq. Herramientas como `edgeR` y `DESeq2` para el análisis de DGE, para análisis DIE y `DEXSeq` para análisis de DS, se basan todas en el uso de GLMs con distribución NB. En general, estas consideran que y_{ik} representa los conteos del i-ésimo gen/exón en la k-ésima muestra secuenciada y asumen a y_{ik} como una **variable aleatoria** de distribución NB, según se indica en la Ecuación 5.1.

$$y_{ik} \sim NB(\mu_{ik} = p_{ik} s_k, \phi_i) \tag{5.1}$$

La distribución NB se caracteriza por dos **parámetros**: la *media* μ_{ik} y la *dispersión* ϕ_i. Específicamente, los paquetes `R` previamente mencionados asumen que la media se puede expresar como el producto de la proporción de fragmentos de ARNm (p_{ik}) del i-ésimo gen que originalmente había en la k-ésima muestra con un factor s_k, relacionado con el tamaño de la libería de la k-ésima muestra. El parámetro ϕ_i está directamente relacionado con la varianza de la variable

aleatoria y_{ik}, como lo evidencia la Ecuación 5.2.

$$V(y_{ik}) = \mu_{ik} + \phi_i \mu_{ik} \tag{5.2}$$

Posteriormente, los métodos utilizan GLMs con *enlace logarítmico* (log_2) del tipo $log_2(p_{ik}) = \sum_r x_{kr}\beta_{ir}$, con los elementos x_{kr} de la matriz de diseño para sus inferencias. En el caso más sencillo, un experimento control-tratamiento, $r = 1$ y la matriz de diseño consistirá de una columna donde se indica si la muestra k pertenece o no a la condición tratamiento.

`NBSplice`, toma como base la estrategia previamente descrita aunque propone un **enfoque alternativo** al comúnmente usado. El método desarrollado asume que los conteos, en escala CPM, de la i-ésima isoforma que proviene del j-ésimo gen en la k-ésima muestra, y_{ijk}, siguen una distribución NB con parámetros dados por la Ecuación 5.3,

$$y_{ijk} \sim NB(media = p_{ijk}\mu_{jk}, dispersion = \phi_j) \tag{5.3}$$

donde p_{ijk} representa la **proporción** de los fragmentos de mRNA del j-ésimo gen (μ_{jk}) que pertenecen a la i-ésima isoforma y ϕ_j es la dispersión del j-ésimo gen. En este contexto, para cada isoforma, es posible **predecir** su expresión media relativa en cada condición, es decir su proporción media, mediante el ajuste de un GLM a **nivel de gen** según la Ecuación 5.4,

$$ln(p_{ijk}) = x_{r.}\beta_{ij} \tag{5.4}$$

donde β_{ij} representa el cambio en p_{ijk} debido al efecto descrito por la r-ésima columna de la matriz del modelo $\boldsymbol{X}$ ($x_{r.}$). De esta manera, `NBSplice` propone, a diferencia de otros métodos como `DEXSeq` o `Limma`, el ajuste de un **modelo para cada gen** que considere a sus distintas isoformas como efectos. En el caso más sencillo, en el que se analiza un experimento que incluye sólo un factor *condición* con dos niveles $l = 1, 2$, la Ecuación (5.4) se resume a:

$$ln(p_{ijl}) = \mu_0 + iso_{ij} + cond_l + iso_{ij} : cond_l \tag{5.5}$$

Una vez estimados los coeficientes del modelo, $\hat{\beta}$, se determina si los cambios en la proporción p_{ijl} al comparar los valores de la l-ésima condición con la de

referencia son **significativos**, mediante un *test de hipótesis lineal* o de Wald (Ecuación 5.6) (Greene, 2017). Esta prueba se basa en el estadístico F definido según la Ecuación 5.7, donde H es la matriz de contraste, q es una matriz de un elemento que contiene el valor del lado derecho de la Ecuación 5.6 (es decir, 0), $\hat{cov}(\hat{\beta})$ es la matriz de varianzas y covarianzas de $\hat{\beta}$ y p son es la cantidad de filas de H, es decir 1. Específicamente, H es una matriz de rango completo, conformable con $\hat{\beta}$ de manera que al multiplicarse por éste determina la combinación lineal definida en el lado izquierdo de la ecuación 5.6. Luego, el estadístico F tendrá, bajo hipótesis nula, distribución F con 1 grado de libertad en el numerador y $n - K$ en el denominador, donde n es la cantidad de veces que se cuantificó una de las isoformas del gen y K es la cantidad de coeficientes del modelo. En el caso particular en el que sólo se tiene un factor con dos niveles (comparación caso-control) K será igual al producto de la cantidad de isoformas del gen j analizadas, I_j, por el número de niveles del factor, 2. La comparación del estadístico F con su distribución bajo hipótesis nula permitirá asignar un valor p a la prueba, correspondiente a la probabilidad de que dicho estadístico tenga esa distribución.

$$H_0 : cond_l + iso_{ij} : cond_l = 0 \tag{5.6}$$

$$F = \frac{(H\hat{\beta} - q)'(H\hat{cov}(\hat{\beta})H')^{-1}(H\hat{\beta} - q)}{p} \tag{5.7}$$

Dado que el DS afecta a todo el gen, una vez que se han hecho todos los test de hipótesis de las isoformas del j-ésimo gen, `NBSplice` utiliza el *Test de Simes* (Simes, 1986) para combinar todos los *valores p* de los test de hipótesis de cada isoforma y así obtener un sólo valor para caracterizar dicho gen. Dada una familia de hipótesis nulas $H_{01}, ..., H_{0I_j}$, para probar los cambios en la expresión relativa de las I_j isoformas del gen j, esta prueba pretende evaluar una hipótesis nula global, $H_0 = \bigcap_{m=1}^{I_j} H_{0i}$. Para ello, los valores p asociados a dichas hipótesis se ordenan $valor\ p_{(1)j} < \ldots < valor\ p_{(m)j} < \ldots < valor\ p_{(I)j}$ y posteriormente se define el estadístico de simes, T_j, según la Ecuación 5.8. Bajo hipótesis nula, este estadístico se distribuye $U(0, 1)$, por lo que es en sí un valor p y el Test de Simes rechaza H_0 si T_j es menor o igual al umbral de significancia α (Simes, 1986).

$$T_j = \text{mín}\{\frac{I_j valor\ p_{(m)j}}{m}\} \tag{5.8}$$

Finalmente, tanto los valores p a nivel de isoforma como los obtenidos a nivel de gen son corregidos con el método FDR (Benjamini and Hochberg, 1995). `NBSplice` se ha escrito en lenguaje `R`. Para su funcionamiento, éste se basa en paquetes y funciones de `R` ya existentes implementados a partir de los métodos clásicos de estimación para GLMs (McCullagh, 1984). En particular, la función `glm.nb` del paquete `MASS` (Venables and Ripley, 2002) es utilizada para ajustar los parámetros de los modelos. Se optó por esta implementación ya estima el parámetro ϕ y además reporta, mediante una variable lógica, si las estimaciones del modelo han o no convergido. Esto último es utilizado en `NBSplice` para decidir si se va o no a hacer inferencias sobre un gen, ya que si no hubo convergencia, esto no será posible. Adicionalmente, la función `lht` del paquete `car` (Fox and Weisberg, 2011) se ha utilizado para realizar los contrastes lineales de hipótesis y la función `simes.test` del paquete `mppa` (Rubin-Delanchy and Heard, 2014), para el Test de Simes.

Funcionalidades

`NBSplice` se encuentra disponible en forma de paquete `R` actualmente disponible en el repositorio GitHub `https://github.com/gamerino/NBSplice`. En forma resumida, el análisis con `NBSplice` sigue los pasos listados a continuación:

- *Construcción de la matriz de expresión*: A partir de los datos de cuantificación a nivel de isoforma de herramientas como `RSEM` o `kallisto` se construye en `R` la matriz de expresión en escala CPM, de dimensiones $N \times P$.

- *Cuantificación a nivel de genes*: La matriz de expresión a nivel de isoforma se utiliza para obtener la expresión a nivel de genes. Para facilitar esta tarea se ha diseñado la función `totalGeneCounts` la cual toma cada gen y, en cada muestra, suma la expresión de sus isoformas para poder estimar su expresión total y devolverla en forma de matriz del mismo tamaño que la matriz de isoformas ($N \times P$). Esta función recibe tres parámetros:

  - `iso_cm`: matriz de expresión a nivel de isoformas. Éstas deben estar organizadas en las filas y las muestras en columnas, como se ejemplifica

en la Ecuación 5.9.

$$ME = \begin{array}{c} \\ Isoforma_1 \\ Isoforma_2 \\ \vdots \\ Isoforma_N \end{array} \begin{array}{c} \begin{array}{cccc} Muestra_1 & Muestra_2 & ... & Muestra_P \end{array} \\ \begin{pmatrix} m_{11} & m_{12} & \dots & m_{1P} \\ m_{21} & m_{22} & \dots & m_{2P} \\ \vdots & \vdots & \vdots & \vdots \\ m_{N1} & m_{N2} & \dots & m_{NP} \end{pmatrix} \end{array} \tag{5.9}$$

- `geneIso`: matriz IG, generalmente representada mediante un `data.frame`, especificando a que gen pertenece cada isoforma de la matriz de expresión. Para ello debe contener dos columnas llamadas `isoform_id` y `gene_id` y debe seguir el mismo orden que la matriz de expresión, según lo indica la Ecuación 5.10.

$$IG = \begin{array}{c} \begin{array}{cc} isoform\_id & gene\_id \end{array} \\ \begin{pmatrix} Isoforma_1 & Gen_1 \\ Isoforma_2 & Gen_1 \\ Isoforma_3 & Gen_2 \\ \vdots & \vdots \\ Isoforma_N & Gen_M \end{pmatrix} \end{array} \tag{5.10}$$

- `BPPARAM`: parámetro que determina el número de procesadores a utilizar. Debe ser de la clase `bpparam` o equivalente, de la librería `BiocParallel` (Morgan et al.).

- *Identificación de isoformas de baja expresión*: Las isoformas de baja expresión son removidas para evitar la introducción de ruido a los modelos. La baja expresión, teniendo en cuenta el contexto de DS, se definió como absoluta y/o relativa. Para identificar dichas isoformas se construyó la función `lowExprIso` que genera un índice de isoformas que deben ser excluidas. Para su funcionamiento, la función recibe los siguientes parámetros:

  - `iso_cm`: matriz de expresión a nivel de isoformas. Éstas deben estar organizadas en las filas y las muestras en columnas, como se ejemplifica en la Ecuación 5.9.
  - `totalCounts`: matriz de expresión a nivel de genes, obtenida mediante la función `totalGeneCounts`. En forma genérica esta última matriz

será de la forma mostrada en la Ecuación 5.11, donde mg_{ij} representa la expresión total del gen del cual proviene la i-ésima isoforma, según la información provista por la matriz IG, en la j-ésima muestra.

$$MG = \begin{array}{c} \\ Isoforma_1 \\ Isoforma_2 \\ \vdots \\ Isoforma_N \end{array} \begin{array}{c} \begin{array}{cccc} Muestra_1 & Muestra_2 & \dots & Muestra_P \end{array} \\ \begin{pmatrix} mg_{11} & mg_{12} & \dots & mg_{1P} \\ mg_{21} & mg_{22} & \dots & mg_{2P} \\ \vdots & \vdots & \vdots & \vdots \\ mg_{N1} & mg_{N2} & \dots & mg_{NP} \end{pmatrix} \end{array} \tag{5.11}$$

- `designMatrix`: matriz, generalmente representada mediante un `data.frame`, especificando el diseño experimental. A modo de ejemplo, suponga que sólo se tiene un efecto, la matriz de diseño estará definida por la Ecuación 5.12.

$$MD = \begin{array}{c} \\ Muestra_1 \\ Muestra_2 \\ \vdots \\ Muestra_P \end{array} \begin{array}{c} \begin{array}{c} condition \end{array} \\ \begin{pmatrix} control \\ caso \\ \vdots \\ caso \end{pmatrix} \end{array} \tag{5.12}$$

- `contrast`: cadena de caracteres indicando el nombre de la columna de `designMatrix` que especifique las condiciones experimentales que se usarán luego para el análisis. De esta manera se podrán identificar isoformas de baja expresión por condición. Su valor por defecto es "`condition`", por lo que si se desea utilizarlo, `designMatrix` debe contener una columna con este nombre, como en la Ecuación 5.12.

- `CPM`: es un parámetro lógico que indica si la matriz `cm` está en escala de CPM. Si no está, la transforma antes de identificar las isoformas de baja expresión. Su valor por defecto es `TRUE`.

- `ratioThres`: valor numérico indicando el umbral a aplicar la expresión relativa de las isoformas (proporción). Cualquier isoforma que tenga al menos una proporción promedio por condición que no supere este valor será considerada como de baja expresión.

- `countThres`: valor numérico indicando el umbral a aplicar la expresión absoluta de las isoformas. Cualquier isoforma que tenga al menos una expresión promedio por condición que no supere este valor será

considerada como de baja expresión.

- `BPPARAM`: parámetro que determina el número de procesadores a utilizar. Debe ser de la clase `bpparam` o equivalente, de la librería `BiocParallel` (Morgan et al.).

- *Ajuste de los modelos y test de hipótesis*: Es realizado con la función `NBTest` de `NBSplice`. Ésta se encarga de iterar el ajuste del GLM y los test de hipótesis, a lo largo de los genes. Una vez que esto ha finalizado, realiza las correcciones de valores p a nivel de isoformas y de genes. La función retorna una matriz, en forma de `data.frame`, conteniendo las proporciones estimadas para cada isoforma, en cada condición, el valor de dispersión estimado, el estadístico del test de hipótesis, y los valores p crudos y ajustados. Internamente, `NBTest` llama a otra función, `fitModel`, que realiza el ajuste del modelo a nivel de gen y los test de hipótesis asociados a las isoformas. Los parámetros que deben especificarse para llamar a `NBTest` son:

  - `iso_cm`: matriz de expresión a nivel de isoformas. La misma que se utilizó en la llamada a `totalGeneCounts`.
  - `gene_cm`: matriz de expresión a nivel de genes obtenida mediante la función `totalGeneCounts`.
  - `idxLowExpr`: índice obtenido mediante la función `lowExprIso` indicando cuáles isoformas deben ser ignoradas en el análisis por considerarse de baja expresión.
  - `geneIso`: matriz, generalmente representada mediante un `data.frame`, la misma que se utilizó en la llamada a `totalGeneCounts`.
  - `designMatrix`: matriz, generalmente representada mediante un `data.frame`, especificando el diseño experimental.
  - `test`: es una cadena de caracteres que determina la distribución que se considerará para el estadístico de cada test de hipótesis lineal. Por defecto, toma el valor "`F`". Si el número de muestras es lo suficientemente grande puede cambiarse a "`chisq`" para asumir distribución χ^2.
  - `BPPARAM`: parámetro que determina el número de procesadores a utilizar. Debe ser de la clase `bpparam` o equivalente, de la librería `BiocParallel` (Morgan et al.).

Finalmente, cabe destacar que si bien las isoformas de baja expresión no son analizadas en términos de DS, sí son consideradas a la hora de determinar la expresión de los genes y, consecuentemente, las proporciones de las isoformas que sí se han analizado. De esta manera se pretende evitar tanto la ocurrencia de **falsos positivos**, característicos de los valores de baja expresión, como **falsas estimaciones** en torno a la proporción real que cada isoforma representó originalmente en la muestra biológica. El archivo `useNBSplice.R` de la Sección A.3.2 del Anexo Digital contiene las instrucciones necesarias para analizar el DS en un conjunto de datos de expresión.

5.2.2. Experimento sintético con control de DS

Con el fin de **caracterizar** y **evaluar** el desempeño de `NBSplice`, se simuló una base de datos de RNA-seq, donde se controló el DS. Para ello se siguió el mismo enfoque y utilizó la misma base de datos reales que se presentó en el Capítulo 4. En particular, la base de datos sintética creada para evaluar `NBSplice` consistió de diez replicaciones de un experimento en base a las mismas **ocho muestras** que se utilizaron para simular los escenarios S1 y S2. Cada simulación contó aproximadamente con 110.000 isoformas provenientes de unos 16.100 genes, de los cuales un **10 %** se simuló con DS.

Subgrupos simulados

Dado que la herramienta desarrollada está destinada a evaluar DS, sólo los grupos **DS** y **DIEDS** previamente definidos se simularon en esta nueva base de datos. Específicamente, la Tabla 5.1 resume los distintos **subgrupos** de cambio que se consideraron en estos dos grupos. Al igual que en la simulación anterior, los subgrupos se diseñaron para controlar el DS principalmente en la isoforma mayor M en las condiciones control (C) y tratamiento (T).

5.2.3. Preprocesamiento

Cada una de las diez replicaciones del experimento sintético fue procesada individualmente. En una primera instancia, se obtuvieron los **perfiles de expresión** para cada muestra, a nivel de transcriptos mediante el uso de la herramienta `kallisto` (Bray et al., 2016). Este programa cuantifica abundancias de isoformas

Tabla 5.1: Casos Simulados de Splicing Diferencial

Grupo	Cambio absoluto en la expresión*	Cambio relativo en la expresión**	Subgrupo
DS	-	0,3-0,5	DS-0,3-0,5
	-	0,3-0,7	DS-0,3-0,7
	-	0,5-0,7	DS-0,5-0,7
	-	0,5-0,9	DS-0,5-0,9
DIEDS	0,5	0,5-0,7	DIEDS-0,5-0,5-0,7
	0,5	0,5-0,9	DIEDS-0,5-0,5-0,9
	2	0,3-0,5	DIEDS-2-0,3-0,5
	2	0,5-0,3	DIEDS-2-0,5-0,3
	2	0,3-0,7	DIEDS-2-0,3-0,7
	2	0,7-0,3	DIEDS-2-0,7-0,3
	2	0,5-0,7	DIEDS-2-0,5-0,7
	2	0,5-0,9	DIEDS-2-0,5-0,9
	4	0,5-0,7	DIEDS-4-0,5-0,7
	4	0,7-0,5	DIEDS-4-0,7-0,5
	4	0,5-0,9	DIEDS-4-0,5-0,9
	4	0,9-0,5	DIEDS-4-0,9-0,5

* Cambio en condición tratamiento respecto a condición control;
** Expresado en término de la proporción de la isoforma de mayor expresión en la condición control-tratamiento.

mediante un **pseudo-alineamiento** de las lecturas de secuenciación al transcriptoma de referencia. De esta manera realiza en simultáneo las dos primeras etapas de cualquier análisis de expresión: alineamiento y cuantificación. Se optó por esta herramienta porque ha demostrado tener la misma o incluso mejor exactitud que RSEM, logrando reducir los tiempos de procesamiento al prescindir de un alineamiento previo (Zhang et al., 2017). Una vez obtenidos los perfiles de expresión, se obtuvo en `R` la **matriz de expresión** para cada una de las repeticiones del experimento. Cabe destacar que estas matrices están en escala de CPMs, de manera que el rango de todas las muestras es el mismo. Posteriormente, cada matriz fue **procesada** según los pasos descritos en la Sección 5.2.1. En particular, los valores de umbral que se utilizaron fueron `ratioThres`= 0,01, `countThres`= 1 CPM y un nivel de significancia α igual a 0,05 tanto para la identificación de genes con DS como para las pruebas estadísticas.

5.2.4. Estrategia de evaluación

La herramienta desarrollada se ha evaluado sobre el conjunto de datos simulados de RNA-seq, donde el DS ha sido controlado. Con el fin de evaluar la **robustez** de la predicción realizada por `NBSplice`, se determinó la **correlación** de Spearman existente entre los valores de proporción simulados con los estimados con dicha herramienta (función `cor.test`). Las **distribuciones** de las proporciones simuladas y las estimadas también se compararon mediante la prueba de Wilcoxon (función `wilcox.test`).

Cada conjunto de datos simulado, se ha procesado individualmente con `NBSplice`. Luego, teniendo en cuenta el estado definido para cada gen durante la simulación y el resultado de `NBSplice`, se clasificaron los genes como:

- *TP*: genes simulados con DS detectados por `NBSplice` como DS
- *FN*: genes simulados con DS detectados por `NBSplice` como NO DS
- *TN*: genes simulados sin DS detectados por `NBSplice` como NO DS
- *FP*: genes simulados sin DS detectados por `NBSplice` como DS

El **desempeño** del método se caracterizó en función del número de genes detectados en todas las simulaciones (concordancia), el promedio de genes identificado como DS y el número de genes incorrectamente identificados como DS. También se calculó, para cada simulación, la exactitud, la sensibilidad, la precisión y el F-score, definidos anteriormente en la Tabla 4.2. Con el fin de evaluar el efecto de la magnitud del DS, se determinó la tasa de TP (TVP o del inglés *true positive rate*, TPR) para cada uno de los grupos simulados descritos en la Tabla 5.1.

5.2.5. Comparación con métodos existentes

El desempeño de `NBSplice` fue comparado con el de **herramientas existentes** destinadas a la detección de DS. En particular, se optó por los paquetes `R`: `DEXSeq` (Anders et al., 2012), `edgeR` (Robinson et al., 2010) y `Limma` (Ritchie et al., 2015). Cabe destacar que estos paquetes son ampliamente usados para inferir el DS a partir de la detección del uso diferencial de exones. Por lo tanto, las estimaciones que éstos realizan no son en base a las proporciones de cada transcrito, como en `NBSplice`, por lo que **no** informan que isoforma cambió su

proporción. El alineamiento y la cuantificación se realizaron con `STAR` y el *script* provisto por `DEXSeq`, tal y como se especifica en la Figura 4.1. Todos estos métodos se utilizaron siguiendo las sugerencias de sus manuales y correspondientes publicaciones.

5.3. Resultados

5.3.1. Evaluación del modelo

Una ventaja que presenta `NBSplice` frente al resto de herramientas de análisis de DS es que permite **predecir** qué valor de **proporción** representó cada isoforma respecto del total de la expresión del gen. Previo a cualquier análisis sobre las estimaciones del método propuesto se determinó la **correlación** entre los valores medios de expresión impuestos por la simulación y los obtenidos luego de ella en las diez repeticiones con `kallisto`. En la condición C se encontró un valor de 0,892, mientras que la correlación para la condición T fue 0,887. Si bien estos valores son muy cercanos entre sí, se encontraron diferencias significativas entre ellos (valor p de la prueba de Wilcoxon = 0,002). Teniendo estos valores como referencia, se evaluó la correspondencia entre los valores de proporción simulados y los estimados por `NBSplice`. Se determinó que el coeficiente de **correlación** de Spearman entre tales valores fue superior a $0,94$ en todas las simulaciones (valor p $< 0,001$). En particular, los valores medios obtenidos fueron 0,9437 para la condición C y 0,942 para la condición T. Al comparar estos valores es notable que son superiores a los obtenidos cuando se correlacionaron los valores de expresión absoluta de las isoformas impuestos por la simulación con los obtenidos por `kallisto`. Esto posiblemente se debe a que en el primer caso se consideraron todas aquellas isoformas que se simularon, mientras que en el segundo caso sólo se tuvo en cuenta las isoformas analizadas por `NBSplice`, es decir que superaron la **etapa de filtrado** (ver Sección 5.2.3). Adicionalmente, se encontró que las distribuciones de las proporciones simuladas y las estimadas en los grupos de genes simulados con DS **no** mostraron diferencias significativas (valor p $> 0,95$) cuando se compararon mediante la prueba de Wilcoxon. En la Figura 5.1 se presentan tales proporciones para las isoformas de mayor expresión de los genes correctamente detectados. La exploración de los FPs develó que los valores de proporción simulados y los estimados **no** mostraron diferencias significativas, en

ninguna de las replicaciones (valores p de la prueba de Wilcoxon $> 0{,}7$), con una correlación media superior a 0,91 para ambas condiciones. Específicamente, en las isoformas que exhibieron valores p ajustados menores a 0,05 se encontraron proporciones promedio cercanas a 0,2, para ambas condiciones, correlacionadas con los valores simulados por un coeficiente $\rho > 0{,}78$. Las diferencias entre estas proporciones fueron en promedio, de 0,15. Por lo tanto, es muy probable que los FPs estén relacionados con isoformas de expresión media, con variabilidad suficiente para detectarlas como DS.

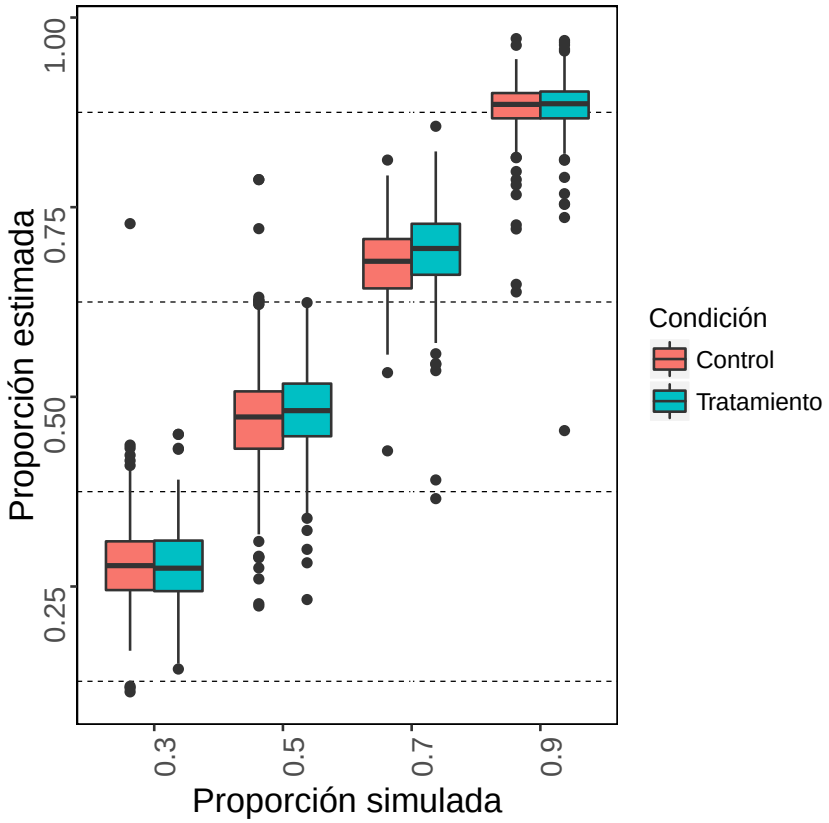

Figura 5.1: Ejemplo de la estimación de la proporción de la isoforma de mayor expresión en los genes simulados con splicing diferencial, correctamente detectados por `NBSplice`.

La Tabla 5.2 resume los resultados obtenidos al analizar las matrices de expresión con `NBSplice`. En promedio, se analizaron 12.028 genes. En total, 14.948 genes se encontraron en al menos una repetición del experimento, de los cuales aproximadamente el 64,4 % (11.014) se estudiaron en todas las simulaciones,

mientras que un total de 85.422 isoformas se encontraron en al menos una muestra en todas las repeticiones. En promedio, `NBSplice` identificó 1.091 (±24) genes con DS. La herramienta identificó un total de 2.122 genes con DS en al menos una replicación, evidenciando un porcentaje de **solapamiento** (concordancia) entre las simulaciones del 77,5 %. En todas las replicaciones se encontraron 1.540 TPs, con un promedio de 1.015 genes. Aproximadamente un 30 % de los genes TPs se identificaron en todas las replicaciones. Al analizar el grupo del cual provienen estos TPs concordantes se encontró que éstos comprendían aproximadamente un 26 % de los genes simulados en el grupo **DS** y un 74 % de los genes del grupo **DIEDS**, porcentajes que se corresponden con los simulados. En término de FPs, se encontraron 582 genes identificados como tales, de los cules sólo el 0,2 %, se identificó en todas las oportunidades. Este resultado es muy importante ya que reveló la **eficacia** de la simulación para representar el DS.

Tabla 5.2: Resultados obtenidos por `NBSplice` en las diez bases de datos de RNA-seq sintéticas

Genes	**Total**	**Promedio** ($\pm DE$)	**Concordancia** (%)
Analizados	14.948	12.028 (±36)	64,6
Detectados como DS	2.122	1.091(±24)	77, 5
TPs	1.540	1.016 (±18)	32, 4
FPs	582	75 (±11)	0, 2

5.3.2. Desempeño de NBSplice

En término de las medidas de desempeño, `NBSplice` evidenció elevados valores de **exactitud**, entre 0,937 y 0,942. La capacidad de identificar TPs fue medida a través de la **sensibilidad**, la cual mostró un valor medio de 0,608 en un rango entre 0,589 y 0,633. `NBSplice` también resultó ser muy **preciso** a la hora de identificar los genes DS (precisión media=0,931). El balance entre estas dos últimas medidas, en términos del **F-score** tomó un valor promedio de 0,736.

El efecto del subgrupo de simulación y la magnitud del DS sobre la TPR en cada uno de los grupos descritos en la Tabla 5.1 se ilustra en la Figura 5.2. En el caso de los subgrupos **DS**, se aprecia que la TPR fue mayor para los grupos en los cuales el cambio en el DS fue **mayor**, con una diferencia de $0, 4$ en la proporción de la isoforma de mayor expresión entre las condiciones C y T. En el

caso del grupo **DS-0,5-0,9**, el valor medio de TPR fue 77,6 %, mientras que el valor medio del grupo **DS-0,3-0,7** (85,5 %) fue significativamente mayor (valor p de la prueba de Wilcoxon $< 0,05$). En los grupos en los cuales la diferencia de proporción de la isoforma M fue $0,2$, se determinó una TPR promedio menor a 40 %, siendo mayor en el grupo **DS-0,3-0,5** que en el caso **DS-0,5-0,7** (valor p de la prueba de Wilcoxon $< 0,05$) .

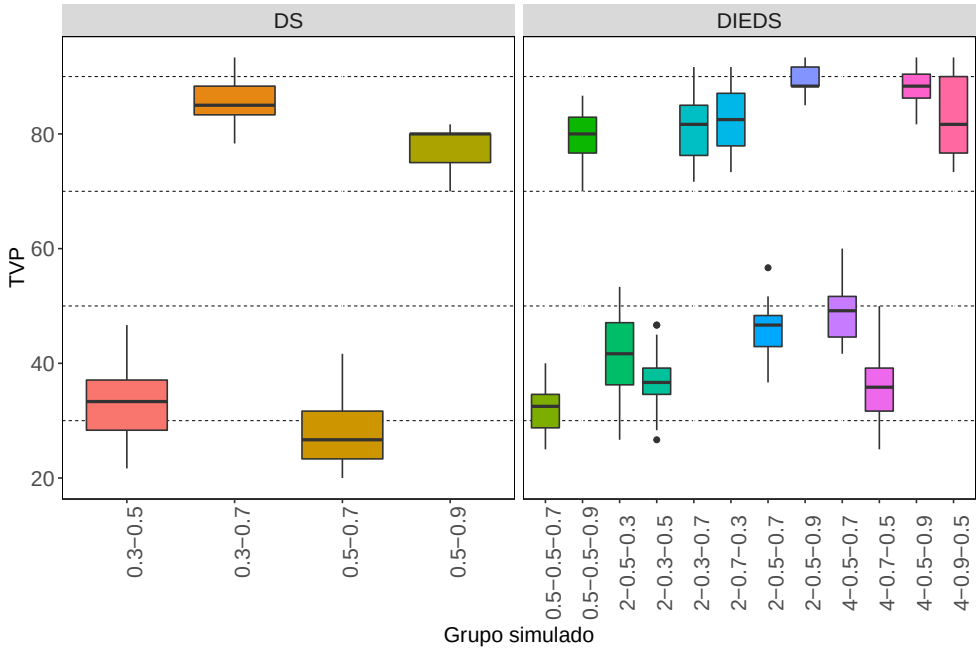

Figura 5.2: Tasa de verdaderos positivos detectados por `NBSplice` en cada uno de los grupo simulados.

En el caso del **Grupo DIEDS**, se encontró una **separación** de los grupos, siendo los que involucraron cambios de proporción de $0,4$ los que alcanzaron mayores valores de TPR respecto de los grupos que involucran cambios en la proporción de 0,2. En particular, los valores **más altos** de TPR se observaron para los grupos donde el cambio de proporción de la isoforma M fue de $0,5$ (condición C) a $0,9$ (condición T), con cambios absolutos en la expresión **favoreciendo** a la condición T. Estos grupos, **DIEDS-2-0,5-0,9** y **DIEDS-4-0,5-0,9**, no mostraron diferencias significativas en la TPR (valor p de la prueba de Wilcoxon $= 0,173$). Sin embargo, para el caso en el cual la expresión absoluta fue en la dirección opuesta al de la proporción, es decir en favor de la condición C, los valores de TPR fueron significativamente menores (valor p de la prueba de Wilcoxon $< 0,05$).

También se evaluó el desempeño de `NBSplice` al **mezclar** las réplicas de la condición T con las de la condición C con el fin de evaluar el manejo de FPs. Para ello, en cada repetición del experimento, se cambiaron aleatoriamente dos de las etiquetas de las cuatro réplicas de cada condición por las de la condición opuesta. De esta manera, las condiciones experimentales quedaron **ocultas**, ya que según la nueva asignación cada una de ellas recibió dos réplicas de cada condición original. Al analizar estos datos con `NBSplice`, sólo en tres de las diez replicaciones se identificaron genes DS. Específicamente, en estas tres oportunidades se encontró, en cada una de ellas, **sólo** un gen con diferencias significativas, evidenciando en estas muestras una FPR menor a 0,004 %. Notablemente, al promediar estos resultados en las diez replicaciones este valor se redujo al 0,001 %, lo cual refleja la **robustez** del método desarrollado.

También se evaluó el efecto de la expresión media, la longitud y la proporción promedio, sobre el desempeño de `NBSplice`. La estrategia utilizada fue similar a la empleada en la Sección 4.3.3. Cada isoforma se categorizó segun grupos definidos en cada una de estas variables de interés y, posteriormente, se calcularon las medidas de desempeño en cada grupo. La Figura 5.3 ilustra los diagramas de cajas obtenidos para los grupos definidos en base a la expresión promedio simulada de las isoformas analizadas por `NBSplice`. En las cuatro medidas consideradas se encontraron diferencias significativas entre los grupos (valores $p < 0{,}05$) cuando se analizaron con la prueba de Kruskal-Wallis (Kruskal and Wallis, 1952). La exactitud (Figura 5.3**A**), la sensibilidad (Figura 5.3**B**) y el F-score (Figura 5.3**D**) evidenciaron un aumento a medida que el valor medio de expresión simulado se incrementó. Mientras, la precisión (Figura 5.3**C**) exhibió un comportamiento diferente, siendo más estable a lo largo de los grupos de expresión. En particular, el grupo de mayor expresión evidenció los mayores valores de precisión, superiores a 0,95.

La Figura 5.4 muestra los diagramas de cajas obtenidos para los grupos definidos sobre la expresión relativa promedio simulada de las isoformas analizadas por `NBSplice`. Al igual que en el caso anterior, se encontraron diferencias significativas entre los grupos de isoformas en las cuatro medidas analizadas (valores p de la prueba de Kruskal-Wallis $< 0{,}05$). En términos de exactitud (Figura 5.4**A**), se encontró que ésta no tuvo un comportamiento claro entre los tres grupos analizados aunque los valores más altos se encontraron para el grupo de mayor expresión relativa promedio. Por otro lado, tanto para la sensibilidad (Fi-

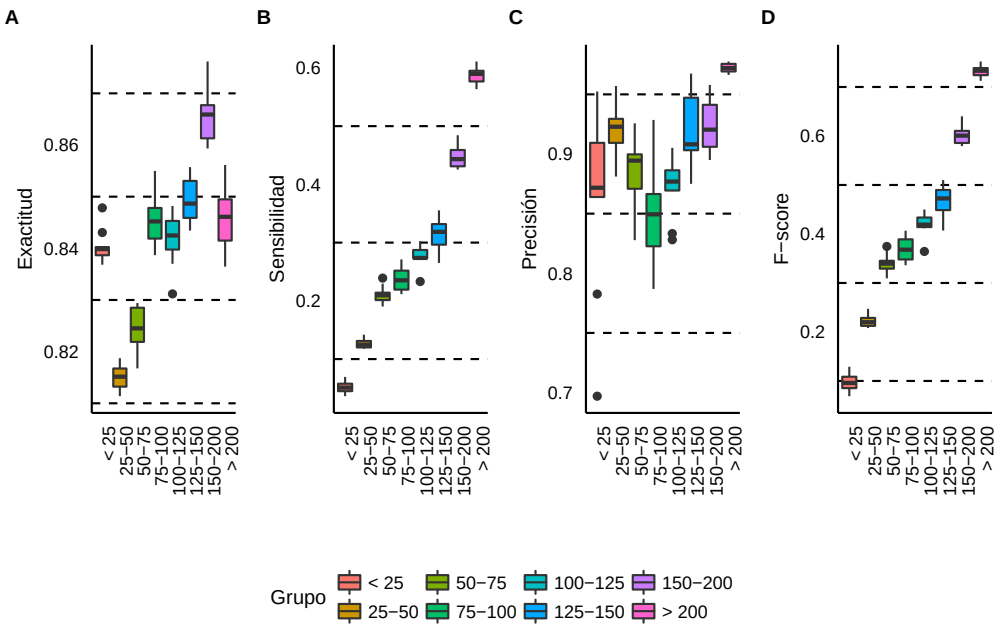

Figura 5.3: Medidas de desempeño evaluadas en grupos de isoformas divididas según su nivel de expresión absoluta promedio simulado **A)** Exactitud, **B)** Sensibilidad, **C)** Precisión y **D)** F-score.

gura 5.4**B**) como para la precisión (Figura 5.4**C**) y consecuentemente, para el F-score (Figura 5.4**D**), se encontró un incremento a medida que el valor medio de expresión relativa aumentó.

Finalmente, los diagramas de cajas de las medidas de desempeño sobre los grupos definidos en función de la longitud de las isoformas se ilustran en la Figura 5.5. El análisis de cada uno de los paneles de datos evidenció la existencia de diferencias significativas entre las medidas de cada grupo (valores p de la prueba de Kruskal-Wallis $<$ 0,05). Análogamente a lo encontrado cuando se exploró el efecto de la expresión absoluta de isoformas, se determinó que la exactitud (Figura 5.5**A**), la sensibilidad (Figura 5.5**B**) y el F-score (Figura 5.5**D**) evidenciaron aumentos con el incremento de la longitud de las isoformas. Este comportamiento se cree que es una consecuencia de que, mientras más larga y/o más expresión tuvo una isoforma, más ADNc fue obtenido a partir de ella (ver Sección 2.2.3) y consecuentemente, más sencilla y precisa ha resultado su cuantificación, por lo que tales isoformas tienen mayor probabilidad de ser correctamente identificadas por el método de análisis de DS. Opuestamente, la precisión (Figura 5.5**C**) mostró disminución a medida que aumentó la longitud de las isoformas, probablemente

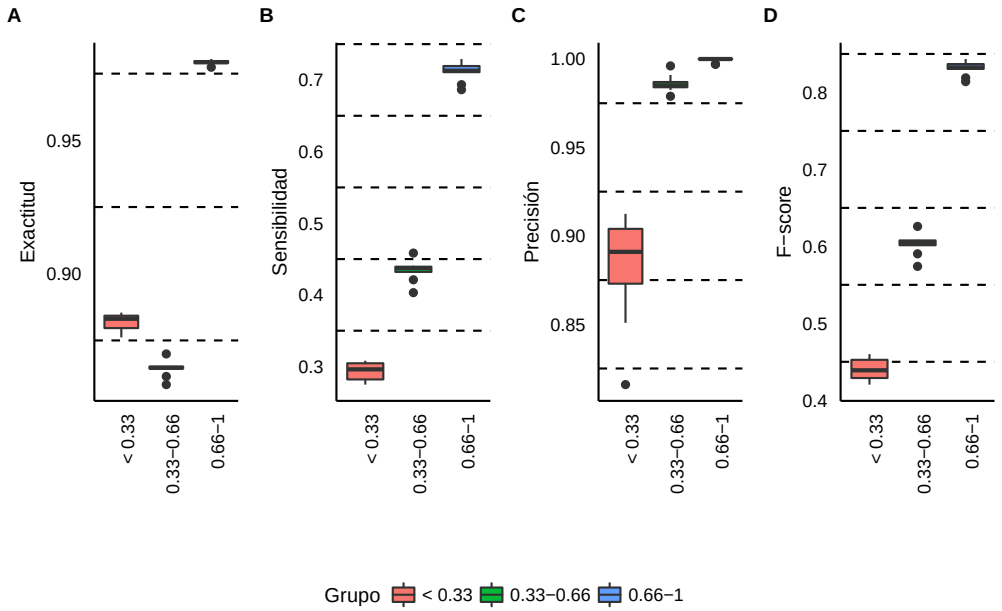

Figura 5.4: Medidas de desempeño evaluadas en grupos de isoformas divididas según su nivel de expresión relativa simulado promedio. **A)** Exactitud, **B)** Sensibilidad, **C)** Precisión y **D)** F-score.

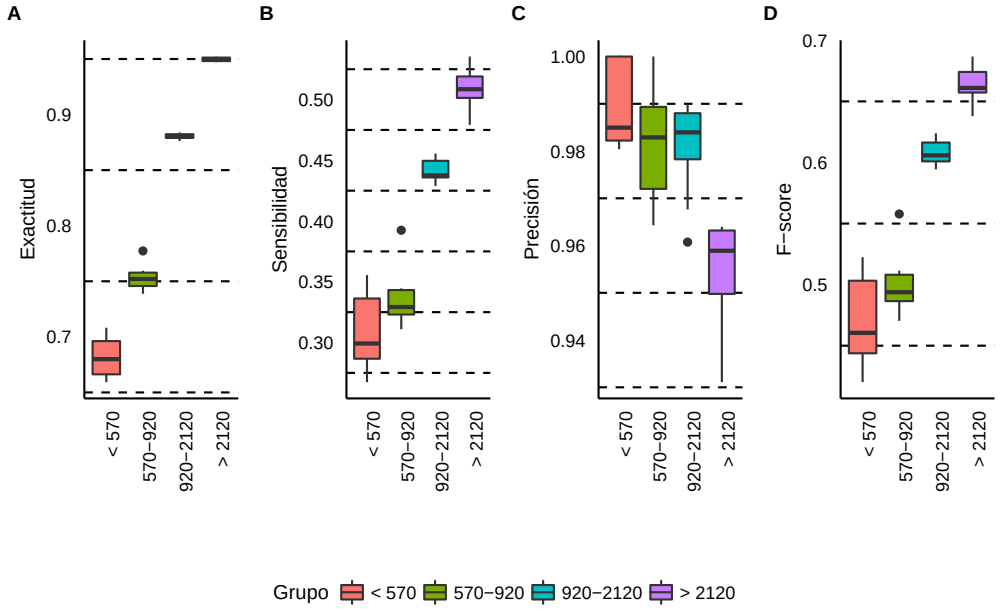

Figura 5.5: Medidas de desempeño evaluadas en grupos de isoformas divididas según su longitud. **A)** Exactitud, **B)** Sensibilidad, **C)** Precisión y **D)** F-score.

ocasionada por el aumento del solapamiento entre las que provienen del mismo gen. Cabe destacar que los patrones observados tanto para el nivel de expresión como para la longitud son similares a los encontrados en la Sección 4.3.3, cuando se analizó el efecto del nivel de expresión sobre el desempeño de los flujos DS.

5.3.3. Comparación con herramientas existentes

La Figura 5.6 ilustra los resultados de la comparación de `NBSplice` con tres herramientas actuales de detección de DS. La prueba de Kruskal-Wallis y la prueba de Dunn (Dunnett, 1955) se utilizaron para determinar diferencias significativas entre los métodos analizados. Esto se realizó mediante las funciones `kruskal.test` y `dunnTest` del paquete `FSA` (Ogle, 2017). Los resultados revelaron la existencia de diferencias significativas entre todos los métodos, para las cuatro medidas de desempeño, soportando los resultados visuales de dicha figura. Tal y como se puede apreciar, `NBSplice` resultó significativamente **superior** en términos de **exactitud** (Figura 5.6**A**), **precisión** (Figura 5.6**C**) y **F-score** (Figura 5.6**D**). En particular, la exactitud promedio observada fue, para `NBSplice`, superior a 0,939, mientras que el resto de las herramientas lograron valores inferiores a dicha cantidad. El segundo valor más elevado fue para `DEXSeq` el cual logró una exactitud promedio de 0,925, mientras que el valor más bajo se registró para `edgeR` (0,871). Si bien se encontraron diferencias significativas entre todos estos grupos de datos, el valor más bajo de éstos es casi un 90 % del valor más alto, lo que revela que en realidad todos los métodos tuvieron desempeño similar según esta medida.

Por otro lado, `DEXSeq` logró los valores más altos de **sensibilidad** (Figura 5.6**B**). Este método logró recuperar, en promedio, el 73 % de los genes simulados con DS. El segundo valor más alto de sensibilidad, 17 % menor que el anterior, fue el de `NBSplice` (0,608). Cabe destacar que en ambos casos los valores fueron superiores al 50 %, mientras que las otras dos herramientas comparadas **no** superaron dicho valor. Más aún, tanto `DEXSeq` como `NBSplice` lograron detectar el doble o más de los genes simulados con DS que las otras dos herramientas evaluadas. Pese a su elevada sensibilidad, `DEXSeq` resultó ser la menos precisa a la hora de determinar los TP, lo cual evidencia la baja confiabilidad en sus detecciones. Contrariamente, `NBSplice` acompañó su buena sensibilidad con una **precisión** media de 0,93, la más alta entre todos los métodos, mientras que `DEXSeq` apenas logró el 68 % de dicho valor. `Limma` y `edgeR` superaron a `DEXSeq`, logrando valores prome-

dio de 0,847 y 0,882, respectivamente. Por último, el **F-score** reveló que `NBSplice` logró el mayor balance entre sensibilidad y precisión, evidenciando un valor medio de 0,736. Los valores de este indicador encontrados para `DEXSEq`, `Limma` y `edgeR` representaron el 69 %, 59 % y 41 % del F-score logrado por `NBSplice`.

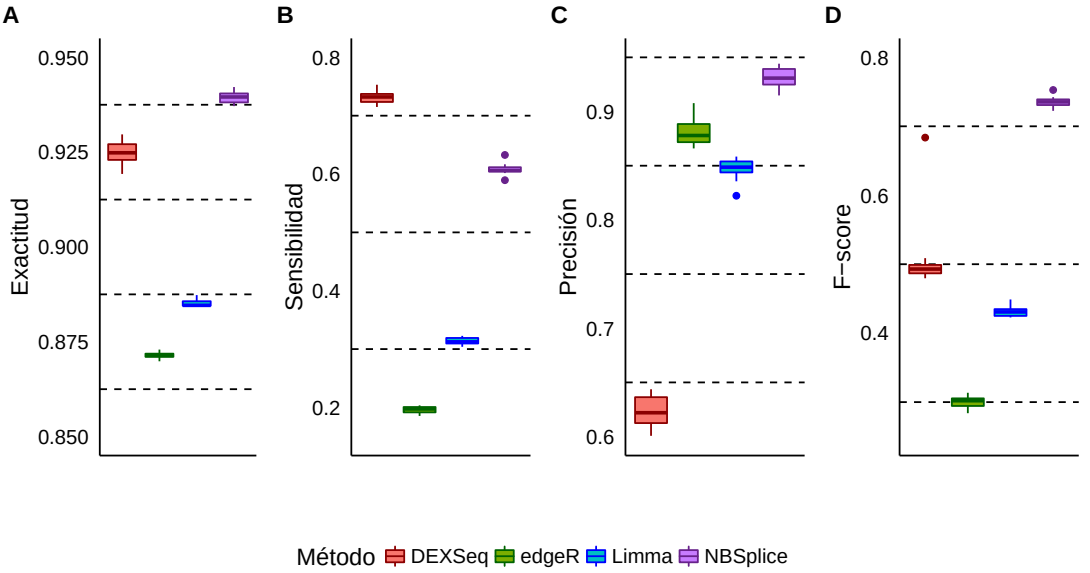

Figura 5.6: Comparación del desempeño de `NBSplice` con tres paquetes `R` para la detección de splicing diferencial, evaluados sobre la base de datos simulados de RNA-seq. Los paneles resumen los resultados de las medidas de **A)** Exactitud, **B)** Sensibilidad, **C)** Precisión y **D)** F-score.

5.4. Conclusiones

Aquí se presentó `NBSplice`, un método novedoso para la **detección** y **cuantificación** del DS a partir de datos de RNA-seq. La herramienta basa su desarrollo en los GLMs con distribución NB, ampliamente utilizados en el ámbito del análisis de expresión diferencial en datos de RNA-seq. Lo novedoso de su enfoque radica en la **estimación** de cambios a partir de la **expresión relativa** de las isoformas de un gen, en lugar del uso diferencial de exones. `NBSplice` es una herramienta sencilla, escrita en código `R`, que emplea para su funcionamiento diversos paquetes ampliamente utilizados en este lenguaje.

`NBSplice` se encuentra en una etapa temprana de desarrollo, lo que fundamenta el hecho de que aún no se haya confeccionado como un paquete `R` disponible

mediante los repositorios más utilizados. Sin embargo, los resultados aquí encontrados evidencian su **potencialidad** a la hora de detectar genes que presentan DS entre dos condiciones experimentales. El análisis desarrollado demostró que las estimaciones de los valores de expresión relativa de las isoformas se **correlacionaron positivamente** con los valores de expresión simulados, con coeficientes superiores a 0,94 en todas las replicaciones. El método propuesto resultó ser **más eficiente**, en término de genes simulados con DS correctamente identificados, cuando los cambios en el DS fueron mayores. Tanto la sensibilidad como la precisión resultaron superiores para valores de expresión absoluta y relativa altos. **Comparado** con métodos existentes, `NBSplice` evidenció mayor **precisión** y **exactitud** en la detección del DS, demostrando su potencialidad como herramienta para el estudio las modificaciones en el SA.

Capítulo 6

Conclusiones y trabajo futuro

En esta tesis se presentó la aplicación de la **minería de datos al estudio de modificaciones post-transcripcionales del ARN**, en el contexto de experimentos **transcriptómicos**. Tal aplicación surge ante la necesidad de contar con un marco ordenado de trabajo que permita obtener información relevante a partir de conjuntos de datos voluminosos y de estructura compleja, como los generados por experimentos transcriptómicos. Particularmente, en el estudio de las modificaciones del ARN como el *splicing alternativo* y sus alteraciones se encuentran diversas limitaciones. Si bien existen herramientas para tal fin, no existe un claro consenso acerca del flujo de trabajo a seguir, los niveles de información que deben analizarse y cuáles de ellas utilizar a la hora de extraer la mayor cantidad de información relevante. En este contexto, en el Capítulo 1 se describieron el concepto de **minería de datos**, como una etapa de un proceso mucho más general conocido como **descubrimiento de información en bases de datos** (KDD, por sus siglas del inglés). El KDD proporciona un marco de referencia *ordenado* de trabajo, *aportando* herramientas y *dirigiendo* el trabajo hacia la *búsqueda* de información relevante. Ésto se logra mediante la aplicación de algoritmos computacionales que buscan conocimiento en grandes volúmenes de datos. El KDD consta de distintas etapas, comenzando por la descripción y contextualización del problema (Sección 1.3.1), la obtención del conjunto de datos, su control de calidad y pre-procesamiento (Sección 1.3.2), el análisis de datos con herramientas apropiadas (Sección 1.3.3), la evaluación de los resultados obtenidos (Sección 1.3.4), y la generación de informes y reportes con visualizaciones apropiadas (1.3.5). Cada una de estas etapas han sido llevadas a cabo en el contexto de la problemática abordada por esta tesis y su resultado se presenta en

cada uno de los capítulos posteriores.

El Capítulo 2 presentó el *análisis transcriptómico* como estrategia para estudiar las modificaciones post-transcripcionales del ARN. En primer lugar se introdujo a la *transcriptómica*, como una herramienta para la exploración de las modificaciones del ARN, y todos los conceptos necesarios para el *entendimiento del problema*. Específicamente se estudió el *splicing alternativo*, que es la modificación más frecuente, y sus alteraciones. Las técnicas existentes para la exploración transcriptómica fueron presentadas, de las cuáles las tecnologías de segunda generación (conocidas como NGS) fueron las seleccionadas como fuente generadora de los datos a utilizar en esta tesis. Luego en la Sección 2.2 se abordó la creación del conjunto de datos, como parte del proceso de *entendimiento de los datos*. Se destacó la importancia de la etapa de **diseño del experimento** y las consideraciones necesarias en un contexto de datos transcriptómicos. Posteriormente, se describió detalladamente las distintas etapas que componen la creación del conjunto de datos desde la muestra biológica hasta la obtención de la **matriz de expresión**. En particular, se mostró que ésta puede ser construida considerando distintos niveles de información: exones, isoformas o genes. Finalmente, en la Sección 2.3 se presentó el **análisis de expresión diferencial** sobre matrices de expresión obtenidas en experimentos transcriptómicos que analizan más de una condición experimental. En este contexto, se identificaron tres niveles de análisis, la *expresión diferencial de genes*, la *expresión diferencial de isoformas* y el *splicing diferencial*. Para cada uno de ellos, se presentaron las estrategias de procesamiento comúnmente utilizadas así como también las limitaciones que éstas poseen. Fundamentalmente, se hizo hincapié en la falta de control de calidad del proceso, consenso e integración de los resultados de los tres niveles de análisis, y en la ausencia de una herramienta de análisis de splicing basada en la cuantificación de isoformas.

En el Capítulo 3 se describió brevemente el **flujo de trabajo ordenado** propuesto desde el enfoque de la minería de datos al análisis transcriptómico a nivel de genes. A partir de un esquema de análisis comúnmente utilizado, se diseñó un protocolo de operación estándar innovador dirigido a controlar la calidad de los datos en *todas* las etapas del mismo (Sección 3.2.1). En este contexto se presentó `TarSeqQC`, una herramienta desarrollada en esta tesis, dirigida al control de calidad y exploración de regiones genómicas específicas. La *aplicación* de los productos de este capítulo se presentaron en la Sección 3.3. En primer lugar se

aplicó el protocolo de operación estándar en el análisis de un conjunto de datos reales, donde se encontraron dos muestras atípicas. Se evaluó el impacto de la inclusión de dichas muestras, por falta del control de calidad global, sobre los resultados del análisis de expresión diferencial, revelando un incremento de más del 300 % en los genes detectados cuando las muestras atípicas fueron identificadas. En la segunda aplicación, `TarSeqQC` se utilizó para analizar un conjunto de datos de *targeted sequencing* dirigido a la identificación de variantes genómicas en un pequeño grupo de genes. La herramienta desarrollada permitió identificar grupos de regiones (*features*) que no fueron secuenciadas correctamente. En forma global se detectó la existencia de problemas durante la preparación de librerías de dos de los pooles de PCR utilizados, lo cual determinó que las regiones que ellos abarcaban se descarten para análisis posteriores, por ser de baja calidad. En forma puntual, se identificaron 91 *features* de baja calidad en todas las muestras analizadas, dos de las cuales no fueron capturadas en ninguna de ellas. Finalmente, se indagó un gen de interés, en el cual se identificaron regiones comúnmente afectadas por variantes, que no pudieron ser indagadas por el experimento.

En el Capítulo 4 se realizó una **evaluación** *sistemática* y *objetiva* de nueve flujos de procesamiento, comúnmente utilizados para el **modelado** y análisis de cambios en la expresión absoluta y relativa de las isoformas. El producto de esta comparación es un conjunto de guías prácticas para asistir la selección de el o los flujos de análisis más apropiados. Estos flujos fueron presentados y categorizados en dos grupos, DIE y DS, según si estudian expresión diferencial de isoformas o splicing diferencial, respectivamente (Sección 4.2.1). Los *pipelines* bajo estudio se **evaluaron** mediante bases de datos sintéticas generadas mediante simulaciones donde se controló el cambio en la expresión. El proceso de simulación diseñado para generar diversos escenarios experimentales, así como también los cambios de expresión considerados, se presentaron en la Sección 4.2.2. La estrategia de evaluación del desempeño se basó en medidas comúnmente utilizadas, computadas sobre diez repeticiones de cada uno de los escenarios experimentales simulados (Sección 4.2.4). Se determinó el desempeño global de cada uno de los flujos, así como también el efecto que causa sobre éstos el nivel de expresión, la longitud, el número de isoformas por gen y la magnitud del cambio en la expresión. Los resultados obtenidos (Sección 4.3) indicaron, por un lado, que ante diferentes escenarios experimentales resulta adecuado aquél que tiene mayor número de réplicas por condición y donde se espera encontrar mayor variabilidad en término de ex-

presión de genes entre condiciones. En cuanto a los flujos evaluados, cinco de los nueve propuestos mostraron buen desempeño global, en término de número de detecciones, concordancia a lo largo de las replicaciones, sensibilidad y precisión. Se encontró un mayor efecto de la longitud y la expresión sobre los flujos DS que los DIE, aunque estos últimos resultaron más influenciados por el número de isoformas por gen. Adicionalmente, se determinó que los mayores cambios en la expresión ejercieron un efecto superior en el desempeño de las herramientas. La **aplicación** sobre datos reales, los mismos usados para construir la simulación, evidenció elevado solapamiento entre los resultados obtenidos por los flujos DIE evaluados, mientras que los flujos DS mostraron resultados más discrepantes. En base a los resultados obtenidos se construyó una guía para asistir la selección del flujo de procesamiento según el experimento y los aspectos a priorizar (resumida en la Figura 4.25).

Finalmente, en el Capítulo 5 se presentó el aporte a la etapa de **modelado**: `NBSplice` que es un **algoritmo estadístico** desarrollado para detectar *splicing diferencial* a partir de la cuantificación de *isoformas* en experimentos transcriptómicos del tipo caso-control. `NBSplice` (Sección 5.2.1) sigue el enfoque de las herramientas más utilizadas en análisis transcriptómico, que emplean modelos lineales generalizados (GLMs) con distribución binomial negativa para inferir cambios en la expresión. Sin embargo, es una herramienta innovadora ya que realiza inferencias en la expresión relativa de las isoformas, en vez de en su expresión absoluta o en el uso de exones, como lo hacen las metodologías actuales. El método propuesto asume que las isoformas de un gen comparten información, por lo que las analiza en un mismo modelo. De esta manera, en cada modelo se asume que la proporción de cada una de ellas es una variable aleatoria. Luego, utiliza GLMs para hacer inferencias sobre los cambios en dichas proporciones como consecuencia de las condiciones experimentales, obteniendo valores de significancia tanto a nivel de genes como de isoformas. De esta manera, el algoritmo desarrollado permite inferir tanto los valores de expresión relativa como el cambio en dicha expresión, debido a las condiciones experimentales, algo que hasta ahora no era posible. `NBSplice` está implementado en lenguaje `R` y utiliza para su funcionamiento una serie de métodos disponibles en paquetes ampliamente utilizados. La **evaluación** de esta herramienta se realizó utilizando bases de datos sintéticas (Sección 5.2.2) sobre las cuales fue aplicada para posteriormente determinar el desempeño en función de medidas adecuadas. Los resultados obtenidos (Sec-

ción 5.3) evidenciaron la potencialidad de `NBSplice` para detectar y cuantificar el splicing diferencial. Particularmente, se encontró que las estimaciones de los valores de expresión simulados y los estimados por el método tuvieron correlación casi ideal. La herramienta desarrollada mostró un muy buen desempeño en términos de exactitud, sensibilidad y precisión. Particularmente, fue más exacta y precisa que herramientas como `DEXSeq` y `Limma`, comúnmente utilizadas para analizar splicing diferencial. Se encontró además que la potencia en la detección de `NBSplice` se favoreció con mayores diferencias entre condiciones y con ocurrencia simultánea de cambios en expresión absoluta y relativa de las isoformas.

Como propuesta de **trabajo a futuro**, se cree que es necesario continuar con el desarrollo de `NBSplice`, con el fin de lograr un paquete `R` que pueda estar en repositorios públicos. En este contexto es fundamental ampliar las capacidades de la herramienta, para así poder analizar experimentos más complejos que los del tipo caso-control. Por otra parte, es posible mejorar la precisión de `NBSplice` mediante la utilización de estrategias alternativas para la estimación de los parámetros de los modelos ajustados. La creación de funciones destinadas a la visualización y exploración de los resultados es otro de los objetivos a considerar. El paquete desarrollado deberá además contemplar la posibilidad de utilizar otros paquetes disponibles que realicen análisis a nivel de genes de manera de poder integrar los resultados de éstos con los de nuestra herramienta. Así, será posible generar reportes globales que unifiquen los resultados obtenidos a distintos niveles de información, enriqueciendo el proceso de extracción de conocimiento. Todos los nuevos desarrollos requerirán una nueva etapa de evaluación, para la cual se deberán considerar los casos ya estudiados, para así determinar si se han obtenido, o no, mejoras en los resultados. En adición, se deberán incluir otros escenarios experimentales de evaluación, variando el número de réplicas por condición, tamaño de librerías, considerando diseños no balanceados, entre otros posibles, y otras bases de datos reales a partir de las cuales se generan los datos simulados. Estas pueden ser obtenidas de repositorios públicos como TCGA o SRA. Dado que en el ámbito del análisis transcriptómico constantemente se generan nuevas herramientas, será necesario además incorporarlas a los nuevos análisis. Finalmente, una vez obtenido el desarrollo final, será indispensable contar con un experimento real de RNA-seq que se pueda analizar con la herramienta para validar los resultados que ésta encontró, mediante técnicas biológicas como *q-RT-PCR*. A la fecha no se dispone de experimentos de este tipo, lo que resulta fundamental para poder

tener estimaciones reales del desempeño de `NBSplice`.

Apéndice A

Anexo Digital

En este anexo se describen los archivos que se encuentran incorporados en el CD que acompaña al documento impreso de tesis y que no se encuentran publicados como material suplementario de alguna de las publicaciones surgidas a partir del trabajo desarrollado.

A.1. Exploración y control de calidad de los datos

En esta sección se describen los archivos conteniendo el código fuente (del inglés, *script*) que listan las funciones necesarias para el control de calidad de datos de secuenciación de dicha herramienta así como también las ordenes necesarias para su ejecución.

A.2. Flujo de análisis

En la Sección 3.2.1 se presentó un *protocolo de operación estándar* diseñado para optimizar la calidad de los resultados obtenidos mediante el análisis de datos de RNA-seq. El protocolo diseñado tiene como fin establecer un procedimiento de análisis basado en herramientas y metodologías existentes sentando una base para el desarrollo de trabajos futuros. En este apartado se lista el código fuente que contiene las funciones que conforman el protocolo de control de calidad y las ordenes necesarias para procesar la base de datos

A.2.1. sourcePipeline.R

Consiste de un *script* escrito en código `R` conteniendo la definición del conjunto de funciones que componen el flujo de operación estándar. Específicamente, las funciones allí definidas son:

- `gene2Analysis`: Función para realizar la sub-etapa *filtrado de genes*. A partir de una matriz de expresión, la función selecciona los genes que serán conservados en análisis posteriores, teniendo en cuenta la anotación actual, mínima longitud de gen y mínimo valor de expresión.
- `inDavid`: Es una función que internamente es utilizada por `gene2Analysis` para comprobar si un gen está o no anotado en la plataforma funcional DAVID `https://david.ncifcrf.gov/`.
- `lengthFilter`: Este método es internamente llamado por `gene2Analysis` para seleccionar los genes que superan un umbral de longitud mínima definida por el usuario.
- `countFilter`: Esta función es internamente llamada por `gene2Analysis` para identificar los genes que superan un umbral de expresión mínima definida por el usuario. El criterio de expresión sobre el que se considerará el umbral puede definirse de cuatro formas diferentes, mediante el parámetro `type`: "mean", expresión media en todas las muestras; "meanCond", expresión media por condición; "allCond", expresión en todas las réplicas de al menos una condición; "all", expresión en todas las muestras.
- `correctGCGL`: Método para corregir el contenido en GC o el efecto de longitud de gen mediante el paquete `cqn`.
- `plotPCA`: Función para graficar el diagrama de dispersión de las dos primeras componentes del PCA sobre la matriz de expresión.
- `countsDist`: Método gráfico para explorar los diagramas de caja de cada muestra, columna de la matriz de expresión en escala $\log_2$.
- `checkBias`: Función para realizar gráficas de control con el fin de explorar los posibles efectos del contenido GC o la longitud de gen sobre la expresión absoluta de genes o el fold change. El gráfico generado consiste de dos paneles: el izquierdo representa el diagrama de dispersión de la fuente de

sesgo (GC o longitud) y la medida de expresión elegida (absoluta o fold change), y el derecho presenta los diagramas de cajas de la medida de expresión en grupos de genes definidos mediante una categorización de la fuente de sesgo en cuatro grupos, según los cuartiles de la distribución de dicha cantidad.

- `decide`: Esta función toma un objeto `DESeqDataSet` y luego de haber ajustado los GLMs y realizado las correspondientes pruebas de Wald determina si un gen fue encontrado como sobre expresado, no diferencialmente expresado o sub-expresado.

- `log2FCBoxplot`: Método gráfico para obtener los diagramas de cajas de los valores de *fold changes* estimados, pudiendo estudiar todos los genes o sólo los encontrados como diferencialmente expresados.

- `checkResBias`: Método gráfico para realizar diagramas útiles para explorar los posibles efectos del contenido GC o la longitud de gen sobre las estimaciones de expresión media de genes o el fold change. El gráfico generado consiste de dos paneles: el izquierdo representa el diagrama de dispersión de la fuente de sesgo (GC o longitud) y la medida de expresión elegida (absoluta o fold change), y el derecho presenta los diagramas de cajas de la medida de expresión en grupos de genes definidos mediante una categorización de la fuente de sesgo en cuatro grupos, según los cuartiles de la distribución de dicha cantidad.

- `scatterPlot`: Función para construir diagramas de dispersión entre las columnas de la matriz obtenida como resultado del análisis de expresión diferencial con `DESeq2`. Las columnas de interés son "baseMean", "log2FoldChange", "stat" y "padj", que contienen la expresión media estimada, el fold change (en escala $\log_2$), el estadístico de Wald y el valor p ajustado asociado a la prueba de Wald, respectivamente. En particular, si el gráfico representa en el eje x a la expresión media y en el eje y al fold change recibe el nombre de *MA plot*; mientras que si el eje x se cambia al valor p ajustado, el gráfico se conoce como *vulcano plot*.

- `heatmapDE`: Método para obtener el mapa de calor (del ingles, heatmap) de los valores de expresión de los genes detectados como diferencialmen-

te expresados para determinar si se agrupan o no según las condiciones experimentales.

A.2.2. pipeline.R

Este script guía el procesamiento de los datos de RNA-seq. Como punto de partida, se requieren los perfiles de expresión obtenidos mediante la herramienta `htseq` u otra equivalente. Como primer paso se construye la matriz de expresión, la cual es posteriormente procesada para filtrar los genes que no serán considerados en análisis posteriores. Luego, se realiza la exploración gráfica para verificar la calidad y consistencia de los datos. Posteriormente, se realiza el análisis de expresión diferencial con `DESeq2`. Finalmente, se exploran y controla la calidad de los resultados obtenidos.

A.3. NBSplice

En la Sección 5.2.1 del Capítulo 5, se presentó una herramienta desarrollada en el marco de esta tesis, llamada `NBSplice`. En este apartado se lista el código fuente que contiene las funciones de dicha herramienta así como también las ordenes necesarias para su ejecución.

A.3.1. NBSplice.R

Consiste de un archivo de código fuente llamado "`NBSplice.R`", el cual se encuentra escrito en lenguaje `R`. El conjunto de funciones que componen la herramienta desarrollada y se encuentran definidas en dicho archivo se lista a continuación:

- `totalGeneCounts`: Permite obtener el total de conteos para cada gen en cada muestra, sumando los valores de expresión de todas sus isoformas. Para evitar errores en las estimaciones de las proporciones de las isoformas, esta función debe ser ejecutada antes de filtrar isoformas de baja expresión.

- `lowExprIso`: Función para identificar aquellas isoformas de baja expresión. En este caso, la baja expresión se determina combinando valores mínimos de expresión relativa y absoluta para las isoformas.

- `buildData`: Es una función interna, que es llamada por `NBTest`, método que se encarga de ajustar los GLMs y realizar las pruebas de hipótesis. Esta función construye la matriz de datos necesaria para el ajuste de un GLM para un gen determinado.
- `fitModel`: Método para ajustar un GLM a nivel de gen y evaluar los cambios en las proporciones de las isoformas expresas de dicho gen. No requiere ser ejecutado por el usuario sino que es internamente llamado por la función `NBTest`.
- `NBTest`: Es la función principal del paquete, se encarga de proveer la interfaz entre el usuario y los métodos que preparan los datos y ajustan los modelos.

A.3.2. usingNBSplice.R

Este script guía el procesamiento de datos de cuantificación de isoformas con `NBSplice`. Como primer paso se construye la matriz de expresión a partir de los archivos de cada una de las muestras. Como ejemplo se ha tomado como partida los archivos de cuantificación generados por `kallisto`. Una vez obtenida la matriz de expresión a nivel de isoformas, se construye la matriz de expresión a nivel de genes. Luego, se determina las isoformas que presentaron bajo nivel de expresión. Finalmente se ajustan los modelos y realizan las pruebas de hipótesis pertinentes y se identifican las isoformas y los genes diferencialmente expresados.

Bibliografía

Afgan, E., Chapman, B., Jadan, M., Franke, V., and Taylor, J. (2012). Using cloud computing infrastructure with CloudBioLinux, CloudMan, and Galaxy. *Current protocols in bioinformatics*, pages 11–9.

Alamancos, G. P., Agirre, E., and Eyras, E. (2014). Methods to study splicing from high-throughput RNA sequencing data. *Spliceosomal Pre-mRNA Splicing: Methods and Protocols*, pages 357–397.

Anders, S. and Huber, W. (2010). Differential expression analysis for sequence count data. *Genome biology*, 11(10):R106.

Anders, S., Pyl, P. T., and Huber, W. (2015). HTSeq—a Python framework to work with high-throughput sequencing data. *Bioinformatics*, 31(2):166–169.

Anders, S., Reyes, A., and Huber, W. (2012). Detecting differential usage of exons from RNA-seq data. *Genome research*, 22(10):2008–2017.

Andrews, S. (2011). FastQC: a quality control tool for high throughput sequence data. `http://www.bioinformatics.babraham.ac.uk/projects/fastqc/`.

Arrieta, M.-C., Stiemsma, L. T., Dimitriu, P. A., Thorson, L., Russell, S., Yurist-Doutsch, S., Kuzeljevic, B., Gold, M. J., Britton, H. M., Lefebvre, D. L., et al. (2015). Early infancy microbial and metabolic alterations affect risk of childhood asthma. *Science translational medicine*, 7(307):307ra152–307ra152.

Aschoff, M., Hotz-Wagenblatt, A., Glatting, K.-H., Fischer, M., Eils, R., and König, R. (2013). SplicingCompass: differential splicing detection using RNA-seq data. *Bioinformatics*, 29(9):1141–1148.

Bailey, P., Chang, D. K., Nones, K., Johns, A. L., Patch, A.-M., Gingras, M.-C., Miller, D. K., Christ, A. N., Bruxner, T. J., Quinn, M. C., et al. (2016).

Genomic analyses identify molecular subtypes of pancreatic cancer. *Nature*, 531(7592):47.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the royal statistical society. Series B (Methodological)*, pages 289–300.

Black, D. L. (2003). Mechanisms of alternative pre-messenger RNA splicing. *Annual review of biochemistry*, 72(1):291–336.

Blencowe, B. and Graveley, B. (2008). *Alternative Splicing in the Postgenomic Era*. Advances in Experimental Medicine and Biology. Springer New York.

Bray, N., Pimentel, H., Melsted, P., and Pachter, L. (2016). Near-optimal probabilistic RNA-seq quantification. *Nature biotechnology*, 34(5):525.

Bro, R. and Smilde, A. K. (2014). Principal component analysis. *Analytical Methods*, 6(9):2812–2831.

Brooker, R. (2017). *Concepts of Genetics: Biology, Genetics*. Cram101.

Cai, M., Gao, F., Zhang, P., An, W., Shi, J., Wang, K., and Lu, W. (2015). Analysis of a transgenic Oct4 enhancer reveals high fidelity long-range chromosomal interactions. *Scientific reports*, 5.

Chu, Y. and Corey, D. R. (2012). RNA sequencing: platform selection, experimental design, and data interpretation. *Nucleic acid therapeutics*, 22(4):271–274.

Ciriello, G., Gatza, M. L., Beck, A. H., Wilkerson, M. D., Rhie, S. K., Pastore, A., Zhang, H., McLellan, M., Yau, C., Kandoth, C., et al. (2015). Comprehensive molecular portraits of invasive lobular breast cancer. *Cell*, 163(2):506–519.

Clark, D. (2005). *Molecular Biology*. Elsevier Science.

Conesa, A., Madrigal, P., Tarazona, S., Gomez-Cabrero, D., Cervera, A., McPherson, A., Szcześniak, M. W., Gaffney, D. J., Elo, L. L., Zhang, X., et al. (2016). A survey of best practices for RNA-seq data analysis. *Genome biology*, 17(1):13.

Cordeiro, G. M. and Demétrio, C. G. (2008). Modelos lineares generalizados e extensões. *Sao Paulo*.

Dennis Jr, G., Sherman, B., Hosack, D., Yang, J., Gao, W., Lane, H., and Lempicki, R. (2003). DAVID: database for annotation, visualization, and integrated discovery. *Genome Biology*, 4(5):P3.

Dobin, A., Davis, C. A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M., and Gingeras, T. R. (2013). STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*, 29(1):15–21.

Dunnett, C. W. (1955). A multiple comparison procedure for comparing several treatments with a control. *Journal of the American Statistical Association*, 50(272):1096–1121.

Engström, P. G., Steijger, T., Sipos, B., Grant, G. R., Kahles, A., Rätsch, G., Goldman, N., Hubbard, T. J., Harrow, J., Guigó, R., et al. (2013). Systematic evaluation of spliced alignment programs for RNA-seq data. *Nature methods*, 10(12):1185–1191.

Fang, Z. and Cui, X. (2011). Design and validation issues in RNA-seq experiments. *Briefings in bioinformatics*, 12(3):280–287.

Fang, Z., Martin, J., and Wang, Z. (2012). Statistical methods for identifying differentially expressed genes in RNA-Seq experiments. *Cell & bioscience*, 2(1):26.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996a). The kdd process for extracting useful knowledge from volumes of data. *Communications of the ACM*, 39(11):27–34.

Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., and Uthurusamy, R. (1996b). *Advances in knowledge discovery and data mining*, volume 21. AAAI press Menlo Park.

Feng, H., Qin, Z., and Zhang, X. (2013). Opportunities and methods for studying alternative splicing in cancer with RNA-Seq. *Cancer letters*, 340(2):179–191.

Fernandez, E., Valtuille, R., Presedo, J., and Willshaw, P. (2005). Comparison of different methods for hemodialysis evaluation by means of ROC curves: from artificial intelligence to current methods. *Clinical nephrology*, 64(3).

Fox, J. and Weisberg, S. (2011). *An R Companion to Applied Regression*. Sage, Thousand Oaks CA, second edition.

Fresno, C. and Fernández, E. A. (2013). RDAVIDWebService: a versatile R interface to DAVID. *Bioinformatics*, 29(21):2810–2811.

Gallego-Paez, L., Bordone, M., Leote, A., Saraiva-Agostinho, N., Ascensão-Ferreira, M., and Barbosa-Morais, N. (2017). Alternative splicing: the pledge, the turn, and the prestige. *Human Genetics*, pages 1–28.

Garber, M., Grabherr, M. G., Guttman, M., and Trapnell, C. (2011). Computational methods for transcriptome annotation and quantification using RNA-seq. *Nature methods*, 8(6):469–477.

Gilbert, W. (1978). Why genes in pieces? *Nature*, 271(5645):501–501.

Goecks, J., Nekrutenko, A., and Taylor, J. (2010). Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome biology*, 11(8):R86.

Goodwin, S., McPherson, J. D., and McCombie, W. R. (2016). Coming of age: ten years of next-generation sequencing technologies. *Nature Reviews Genetics*, 17(6):333–351.

Greene, W. (2017). *Econometric Analysis*. Pearson Education.

Gresham, D., Dunham, M. J., and Botstein, D. (2008). Comparing whole genomes using DNA microarrays. *Nature reviews. Genetics*, 9(4):291.

Han, J., Pei, J., and Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.

Hatem, A., Bozdağ, D., Toland, A. E., and Çatalyürek, Ü. V. (2013). Benchmarking short sequence mapping tools. *BMC bioinformatics*, 14(1):184.

Heller, M. J. (2002). DNA microarray technology: devices, systems, and applications. *Annual review of biomedical engineering*, 4(1):129–153.

Hooper, J. E. (2014). A survey of software for genome-wide discovery of differential splicing in RNA-Seq data. *Human genomics*, 8(1):3.

Huang, D. W., Sherman, B. T., Tan, Q., Kir, J., Liu, D., Bryant, D., Guo, Y., Stephens, R., Baseler, M. W., Lane, H. C., and Lempicki, R. A. (2007). DAVID bioinformatics resources: expanded annotation database and novel algorithms

to better extract biology from large gene lists. *Nucleic Acids Research*, 35(Web Server issue):W169–W175.

Hummel, M., Bonnin, S., Lowy, E., and Roma, G. (2011). TEQC: an R package for quality control in target capture experiments. *Bioinformatics*, 27(9):1316–1317.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323.

Juan Cruz Rodríguez, Cristóbal Fresno, A. L. and Fernández, E. *MIGSA: Massive and Integrative Gene Set Analysis*. R package version 1.2.0.

Kalari, K. R., Rossell, D., Necela, B. M., Asmann, Y. W., Nair, A., Baheti, S., Kachergus, J. M., Younkin, C. S., Baker, T., Carr, J. M., et al. (2012). Deep sequence analysis of non-small cell lung cancer: integrated analysis of gene expression, alternative splicing, and single nucleotide variations in lung adenocarcinomas with and without oncogenic KRAS mutations. *Frontiers in oncology*, 2.

Kanitz, A., Gypas, F., Gruber, A. J., Gruber, A. R., Martin, G., and Zavolan, M. (2015). Comparative assessment of methods for the computational inference of transcript isoform abundance from RNA-seq data. *Genome biology*, 16(1):150.

Kannan, K., Wang, L., Wang, J., Ittmann, M. M., Li, W., and Yen, L. (2011). Recurrent chimeric RNAs enriched in human prostate cancer identified by deep sequencing. *Proceedings of the National Academy of Sciences*, 108(22):9172–9177.

Kim, D., Pertea, G., Trapnell, C., Pimentel, H., Kelley, R., and Salzberg, S. L. (2013). TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome biology*, 14(4):R36.

Kruskal, W. H. and Wallis, W. A. (1952). Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association*, 47(260):583–621.

Langmead, B., Trapnell, C., Pop, M., and Salzberg, S. L. (2009). Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome biology*, 10(3):R25.

Leinonen, R., Sugawara, H., Shumway, M., and Collaboration, I. N. S. D. (2010). The sequence read archive. *Nucleic acids research*, 39(suppl_1):D19–D21.

Leng, N., Dawson, J. A., Thomson, J. A., Ruotti, V., Rissman, A. I., Smits, B. M., Haag, J. D., Gould, M. N., Stewart, R. M., and Kendziorski, C. (2013). EBSeq: an empirical Bayes hierarchical model for inference in RNA-seq experiments. *Bioinformatics*, 29(8):1035–1043.

Li, B. and Dewey, C. N. (2011). RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. *BMC bioinformatics*, 12(1):323.

Li, H. and Durbin, R. (2009). Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–1760.

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., and Durbin, R. (2009). The sequence alignment/map format and SAMtools. *Bioinformatics*, 25(16):2078–2079.

Liu, R., Loraine, A. E., and Dickerson, J. A. (2014). Comparisons of computational methods for differential alternative splicing detection using RNA-seq in plant systems. *BMC bioinformatics*, 15(1):364.

Liu, Y., Zhou, J., and White, K. P. (2013). RNA-seq differential expression studies: more sequence or more replication? *Bioinformatics*, 30(3):301–304.

Love, M. I., Huber, W., and Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome biology*, 15(12):550.

Marioni, J. C., Mason, C. E., Mane, S. M., Stephens, M., and Gilad, Y. (2008). RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome research*, 18(9):1509–1517.

Maza, E., Frasse, P., Senin, P., Bouzayen, M., and Zouine, M. (2013). Comparison of normalization methods for differential gene expression analysis in RNA-Seq experiments: A matter of relative size of studied transcriptomes. *Communicative & integrative biology*, 6(6):e25849.

McCullagh, P. (1984). Generalized linear models. *European Journal of Operational Research*, 16(3):285–292.

Meldrum, C., Doyle, M. A., and Tothill, R. W. (2011). Next-generation sequencing for cancer diagnostics: a practical perspective. *The Clinical Biochemist Reviews*, 32(4):177.

Melé, M., Ferreira, P. G., Reverter, F., DeLuca, D. S., Monlong, J., Sammeth, M., Young, T. R., Goldmann, J. M., Pervouchine, D. D., Sullivan, T. J., et al. (2015). The human transcriptome across tissues and individuals. *Science*, 348(6235):660–665.

Merino, G. A., Conesa, A., and Fernández, E. A. (2017a). A benchmarking of workflows for detecting differential splicing and differential expression at isoform level in human RNA-seq studies. *Briefings in Bioinformatics*.

Merino, G. A. and Fernández, E. A. (2017). NBSplice: Método de evaluación de splicing diferencial en experimentos de RNA-seq. In *Libro de Resúmenes. XXI Congreso Argentino de Bioingeniería - X Jornadas de Ingeniería Clínica, SABI 2017*, volume 1 of *SABI*. Sociedad Argentina de Bioingeniería.

Merino, G. A., Fresno, C., La Greca, A., Soronellas, D., Beato, M., Saragüeta, P., and Fernández, E. A. (2013). A step forward to standard operating protocols for RNA-seq data analysis.

Merino, G. A., Fresno, C., Netto, F., Netto, E. D., Pratto, L., and Fernández, E. A. (2016). The impact of quality control in RNA-seq experiments. In *Journal of Physics: Conference Series*, volume 705, page 012003. IOP Publishing.

Merino, G. A., Murua, Y. A., Fresno, C., Sendoya, J. M., Golubicki, M., Iseas, S., Coraglio, M., Podhajcer, O. L., Llera, A. S., and Fernández, E. A. (2017b). TarSeqQC: Quality control on targeted sequencing experiments in R. *Human mutation*, 38(5):494–502.

Metzker, M. L. (2010). Sequencing technologies–the next generation. *Nature reviews. Genetics*, 11(1):31.

Morgan, M., Obenchain, V., Lang, M., and Thompson, R. *BiocParallel: Bioconductor facilities for parallel evaluation*. R package version 1.4.3.

Morozova, O. and Marra, M. A. (2008). Applications of next-generation sequencing technologies in functional genomics. *Genomics*, 92(5):255–264.

Mortazavi, A., Williams, B. A., McCue, K., Schaeffer, L., and Wold, B. (2008). Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature methods*, 5(7):621–628.

Ng, S. B., Turner, E. H., Robertson, P. D., Flygare, S. D., Bigham, A. W., Lee, C., Shaffer, T., Wong, M., Bhattacharjee, A., Eichler, E. E., et al. (2009). Targeted capture and massively parallel sequencing of twelve human exomes. *Nature*, 461(7261):272.

Nicol, J. W., Helt, G. A., Blanchard Jr, S. G., Raja, A., and Loraine, A. E. (2009). The Integrated Genome Browser: free software for distribution and exploration of genome-scale datasets. *Bioinformatics*, 25(20):2730–2731.

Nieuwenhuis, M. and Vasen, H. (2007). Correlations between mutation site in APC and phenotype of familial adenomatous polyposis (FAP): a review of the literature. *Critical reviews in oncology/hematology*, 61(2):153–161.

Nikolayeva, O. and Robinson, M. D. (2014). edgeR for differential RNA-seq and ChIP-seq analysis: an application to stem cell biology. *Stem Cell Transcriptional Networks: Methods and Protocols*, pages 45–79.

Ogle, D. H. (2017). *FSA: Fisheries Stock Analysis*. R package version 0.8.17.

Oshlack, A., Robinson, M. D., and Young, M. D. (2010). From RNA-seq reads to differential expression results. *Genome biology*, 11(12):220.

Pan, Q., Shai, O., Lee, L. J., Frey, B. J., and Blencowe, B. J. (2008). Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing. *Nature genetics*, 40(12):1413–1415.

Parker, J. and Honeysett, I. (2008). *Revise As/A2 Biology*. Letts A Level Success. HarperCollins Publishers Limited.

Pelish, H. E., Liau, B. B., Nitulescu, I. I., Tangpeerachaikul, A., Poss, Z. C., Da Silva, D. H., Caruso, B. T., Arefolov, A., Fadeyi, O., Christie, A. L., et al. (2015). Mediator kinase inhibition further activates super-enhancer associated genes in AML. *Nature*, 526(7572):273.

Quinlan, A. R. and Hall, I. M. (2010). BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, 26(6):841–842.

R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Ramaswamy, S. and Golub, T. R. (2002). DNA microarrays in clinical oncology. *Journal of Clinical Oncology*, 20(7):1932–1941.

Ramírez, F., Dündar, F., Diehl, S., Grüning, B. A., and Manke, T. (2014). deepTools: a flexible platform for exploring deep-sequencing data. *Nucleic acids research*, 42(W1):W187–W191.

Rhodes, D. R. and Chinnaiyan, A. M. (2005). Integrative analysis of the cancer transcriptome. *Nature genetics*, 37(6s):S31.

Ritchie, M. E., Phipson, B., Wu, D., Hu, Y., Law, C. W., Shi, W., and Smyth, G. K. (2015). limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic acids research*, 43(7):e47–e47.

Robinson, J. T., Thorvaldsdóttir, H., Winckler, W., Guttman, M., Lander, E. S., Getz, G., and Mesirov, J. P. (2011). Integrative genomics viewer. *Nature biotechnology*, 29(1):24–26.

Robinson, M. D., McCarthy, D. J., and Smyth, G. K. (2010). edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140.

Robles, J. A., Qureshi, S. E., Stephen, S. J., Wilson, S. R., Burden, C. J., and Taylor, J. M. (2012). Efficient experimental design and analysis strategies for the detection of differential expression using RNA-Sequencing. *BMC genomics*, 13(1):484.

Rubin-Delanchy, P. and Heard, N. A. (2014). *mppa: Statistics for analysing multiple simultaneous point processes on the real line*. R package version 1.0.

Sánchez-Pla, A., Reverter, F., de Villa, M. C. R., and Comabella, M. (2012). Transcriptomics: mRNA and alternative splicing. *Journal of neuroimmunology*, 248(1):23–31.

Sastry, D. (2010). *Cell And Developmental Biology*. Rastogi Publications.

Segditsas, S. and Tomlinson, I. (2006). Colorectal cancer and genetic alterations in the Wnt pathway. *Oncogene*, 25(57):7531.

Shendure, J. and Ji, H. (2008). Next-generation dna sequencing. *Nature biotechnology*, 26(10):1135–1145.

Shi, Y., Chinnaiyan, A. M., and Jiang, H. (2015). rSeqNP: a non-parametric approach for detecting differential expression and splicing from RNA-Seq data. *Bioinformatics*, 31(13):2222–2224.

Simes, R. J. (1986). An improved Bonferroni procedure for multiple tests of significance. *Biometrika*, 73(3):751–754.

Sims, D., Sudbery, I., Ilott, N. E., Heger, A., and Ponting, C. P. (2014). Sequencing depth and coverage: key considerations in genomic analyses. *Nature Reviews Genetics*, 15(2):121–132.

Soneson, C. and Delorenzi, M. (2013). A comparison of methods for differential expression analysis of RNA-seq data. *BMC bioinformatics*, 14(1):91.

Soneson, C., Matthes, K. L., Nowicka, M., Law, C. W., and Robinson, M. D. (2016). Isoform prefiltering improves performance of count-based methods for analysis of differential transcript usage. *Genome biology*, 17(1):12.

Spearman, C. (1904). The proof and measurement of association between two things. *The American journal of psychology*, 15(1):72–101.

Stamm, S., Smith, C., and Lührmann, R. (2012). *Alternative Pre-mRNA Splicing: Theory and Protocols*. EBL-Schweitzer. Wiley.

Stoughton, R. B. (2005). Applications of DNA microarrays in biology. *Annu. Rev. Biochem.*, 74:53–82.

Sumathi, S. and Sivanandam, S. (2006). *Introduction to Data Mining and Its Applications*, volume 29. Springer.

Tarazona, S., Furió-Tarí, P., Turrà, D., Pietro, A. D., Nueda, M. J., Ferrer, A., and Conesa, A. (2015). Data quality aware analysis of differential expression in RNA-seq with NOISeq R/Bioc package. *Nucleic acids research*, 43(21):e140–e140.

Teng, M., Love, M. I., Davis, C. A., Djebali, S., Dobin, A., Graveley, B. R., Li, S., Mason, C. E., Olson, S., Pervouchine, D., et al. (2016). A benchmark for RNA-seq quantification pipelines. *Genome biology*, 17(1):74.

Thota, S., Viny, A. D., Makishima, H., Spitzer, B., Radivoyevitch, T., Przychodzen, B., Sekeres, M. A., Levine, R. L., and Maciejewski, J. P. (2014). Genetic alterations of the cohesin complex genes in myeloid malignancies. *Blood*, 124(11):1790–1798.

Torralbo, L. and Alfonso, J. (2010). *Marco de Descubrimiento de Conocimiento para Datos Estructuralmente Complejos con Énfasis en el Análisis de Eventos en Series Temporales*. PhD thesis, Informatica.

Trapnell, C., Roberts, A., Goff, L., Pertea, G., Kim, D., Kelley, D. R., Pimentel, H., Salzberg, S. L., Rinn, J. L., and Pachter, L. (2012). Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nature protocols*, 7(3):562.

Trapnell, C., Williams, B. A., Pertea, G., Mortazavi, A., Kwan, G., Van Baren, M. J., Salzberg, S. L., Wold, B. J., and Pachter, L. (2010). Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature biotechnology*, 28(5):511–515.

Van't Veer, L. J., Dai, H., Van De Vijver, M. J., He, Y. D., Hart, A. A., Mao, M., Peterse, H. L., Van Der Kooy, K., Marton, M. J., Witteveen, A. T., et al. (2002). Gene expression profiling predicts clinical outcome of breast cancer. *nature*, 415(6871):530–536.

Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*. Springer, New York, fourth edition. ISBN 0-387-95457-0.

Walpole, R. E., Myers, R. H., Myers, S. L., and Ye, K. (2012). *Probabilidad y estadística para ingeniería y ciencias*. Pearson Education.

Wang, J., Ye, Z., Huang, T. H.-M., Shi, H., and Jin, V. (2015). A survey of computational methods in transcriptome-wide alternative splicing analysis. *Biomolecular concepts*, 6(1):59–66.

Wang, Z., Gerstein, M., and Snyder, M. (2009). RNA-Seq: a revolutionary tool for transcriptomics. *Nature reviews genetics*, 10(1):57–63.

Weinstein, J. N., Collisson, E. A., Mills, G. B., Shaw, K. R. M., Ozenberger, B. A., Ellrott, K., Shmulevich, I., Sander, C., Stuart, J. M., Network, C. G. A. R.,

et al. (2013). The cancer genome atlas pan-cancer analysis project. *Nature genetics*, 45(10):1113–1120.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics bulletin*, 1(6):80–83.

Williams, A. G., Thomas, S., Wyman, S. K., and Holloway, A. K. (2014). RNA-seq data: Challenges in and recommendations for experimental design and analysis. *Current protocols in human genetics*, pages 11–13.

Wolf, J. B. (2013). Principles of transcriptome analysis and gene expression quantification: an RNA-seq tutorial. *Molecular ecology resources*, 13(4):559–572.

Xu, Q., Modrek, B., and Lee, C. (2002). Genome-wide detection of tissue-specific alternative splicing in the human transcriptome. *Nucleic acids research*, 30(17):3754–3766.

Young, M. D., Wakefield, M. J., Smyth, G. K., and Oshlack, A. (2010). Gene ontology analysis for RNA-seq: accounting for selection bias. *Genome biology*, 11(2):R14.

Zhang, C., Zhang, B., Lin, L.-L., and Zhao, S. (2017). Evaluation and comparison of computational tools for RNA-seq isoform quantification. *BMC genomics*, 18(1):583.

Zhao, S., Xi, L., and Zhang, B. (2015). Union exon based approach for RNA-seq gene quantification: To be or not to be? *PloS one*, 10(11):e0141910.