JAGIELLONIAN UNIVERSITY
UJ Małopolskie Centrum Biotechnologii
Bioinformatics Research Group

UNIVERSITAT POLITÈCNICA DE VALÈNCIA
Escola Tècnica Superior D'Ingenyeria Agronòmica I del Medi Natural

Comparison of RNA-Seq differential expression analysis tools – concordance at the gene and functional level

BIOTECHNOLOGY BACHELOR'S THESIS
(TRABAJO DE FIN DE GRADO EN BIOTECNOLOGÍA)

Author:
Rubén FUERTES Gill de Albornoz

UJ tutor:
Dr. Paweł ŁABAJ

UPV tutor:
Javier FORMENT Millet

Course 2017-2018
Krakow, June 2018

Comparison of RNA-Seq differential expression analysis tools – concordance at the gene and functional level

Author: Rubén FUERTES Gill de Albornoz
UPV tutor: Javier FORMENT Millet
UJ tutor: Dr. Paweł ŁABAJ

Location and date: Krakow, June 2018

Abstract

Unveiling of a gene function remains a major bottleneck in improving our understanding of living systems, by understanding the processes and mechanisms that are going on in the cell. Beyond classic sequence analysis, an important source of information about the function of a gene is when, where, and how strongly it is being expressed. While the DNA analysis gives us information about all the genetic information that is in a cell, it is the RNA quantitative analysis which allows us to know which information is being processed and which not.

Thus, in the post-genomic era the genome-scale expression profiling has become a key tool of functional genomics. The introduction of expression profiling by Next Generation Sequencing has brought a new wave of findings in this matter, as it allows the expression profiling of the whole transcriptome in contrast to other, well-established methods which can focus only on handful of selected targets.

As expression profiling is a fast-moving field, both in terms of the technology and data analysis development, the understanding how the data is generated and how the data analysis process works is crucial in order to correctly interpret the results. The recent study by US FDA led SEQC-III/MAQC consortium has shown that there is no technological gold standard in expression profiling. Moreover, this study as well as follow up work has also shown that there is also noticeable discordance between results of RNA-Seq data analysis approaches.

This project aimed to look deeper into this challenge and focus on tools for differential gene expression calling. The benchmarking data set provided by MAQC-III/SEQC consortium was utilized to compare a set of approaches for differential expression gene (DEG) calling as well as for Gene Ontology enrichment analysis. The areas of discordance for both levels (DEG calling end GO terms enrichment) as well the propagation of discordance from one level to another were analyzed.

Keywords – RNA-seq; Differential expression; Gene Ontology enrichment analysis

Resumen

Desvelar la función de los genes sigue siendo clave para nuestro entendimieno de los seres vivos, entendiendo de esta manera los procesos y mecanismos que están teniendo lugar en la célula. Mas allá de un análisis de secuencia clásico, una fuente importante de información sobre la función de un gen es cuánto, dónde y cuándo éste está siendo expresado. Mientras que el análisis de ADN nos muestra toda la información genética que se encuentra en la célula, es el análisis cuantitativo de ARN lo que nos permite conocer que información está siendo procesada y cual no.

Por tanto, en la era post-genómica los perfiles de expresión a escala genómica se han convertido en una herramienta clave de la genómica funcional. La introducción de los perfiles de expresión por *Next Generation Sequencing* ha traido consigo una nueva ola de descubrimientos en esta materia, al permitir analizar el perfil de expresión de todo el transcriptoma, al contrario que otros métodos bien establecidos que podían centrarse tan solo en unos pocos objetivos.

Como el análisis de perfiles de expresión es un campo en constante avance, en términos de tecnología, así como en términos de análisis de datos, poder entender como se generan los datos y como funciona el análisis de datos es crucial para poder interpretar los resultados correctamente. Un estudio reciente de la US FDA conducido por el consorcio SEQC-III/MAQC ha mostrado que no hay una tecnología de referencia en el análisis de perfiles de expresión. Es más, este estudio, junto a trabajos posteriores, ha mostrado que hay una discordancia notable entre los deferentes enfoques hacia el análisis de datos de RNA-seq.

El objetivo de este TFG es indagar con más profundidad en este desafío y centrarse en herramientas de análisis de expresión diferencial. Aprovecharé los datos de referencia proporcionados por el consorcio MAQC-III/SEQC y compararé una serie de enfoques para el análisis de expresión diferencial, así como el análisis de enriquecimiento en términos *Gene Ontology*. Identificaré las áreas de discordancia en ambos niveles (expresión diferencial y análisis de términos GO). También investigaré la propagación de la discordancia de un nivel al otro. Un objetivo adicional será identificar las fuentes de discordancia para ayudar a la mejora de futuros análisis.

Palabras clave – RNA-seq; Expresión Diferencial; analisis enriquecimiento Gene Ontology

Acknowledgements

I would like to express my deep gratitude to my tutor Paweł, who has guided me into the exciting world of the bioninformatics. My grateful thanks are also extended to the members of the lab, Gosia and Agata, who gave a helping hand with my English and helped me with the cohesion of the text. Their willingness to give their time so generously has been very much appreciated.

I also wish to thank the support received from those friends which are always around to have a laugh with them. Especial mention to Fran and Adri, whose backing in the last sprint has been essential.

But, above all, a huge thanks to my family, which has been crucial in this long road. This is the result of the considerable effort that has been made.

Contents

List of Figures

List of Abbreviations

cDNA	**c**omplementary **DNA**
CPM	**C**ounts **P**er **M**illion
ddATP	**did**eoxy**A**denosine **T**ri**P**hosphate
ddCTP	**did**eoxy**C**ytidine **T**ri**P**hosphate
ddGTP	**did**eoxy**G**uanosine **T**ri**P**hosphate
ddNTP	**did**eoxy**N**ucleoside **T**ri**P**hosphate
ddTTP	**did**eoxy**T**hymidine **T**ri**P**hosphate
DE	**D**iferential/ly **E**xpression/ed
DNA	**D**esoxiribo**N**ucleic **A**cid
dNTP	**d**eoxy**N**ucleoside **T**ri**P**hosphate
EST	**E**xpressed **S**equence **T**ags
FDR	**F**alse **D**iscovery **R**ate
FPKM	**F**ragments **P**er **K**ilobase per **M**illion mapped reads
GO	**G**ene **O**ntology
indel	**in**sertion and **del**etion
LASSO	**L**east **A**bsolute **S**hrinkage and **S**election **O**perator
mRNA	**m**essenger **R**ibo**N**ucleic **A**cid
NGS	**N**ext **G**eneration **S**equencing
ONT	**O**xford **N**anopore **T**echnologies
PCR	**P**olymerase **C**hain **R**eaction
RLE	**R**elative **L**og **E**xpression
RNA	**R**ibo**N**ucleic **A**cid
RT-qPCR	**q**uantitative **R**everse **T**ranscription **PCR**
SAGE	**S**erial **A**nalysis of **G**ene **E**xpression
SBL	**S**equencing **B**y **L**igation
SMRT	**S**ingle **M**olecule **R**eal **T**ime sequencing
SNP	**S**ingle **N**ucleotide **P**olimorphism
ssDNA	**s**ingle **s**trand **D**esoxiribo**N**ucleic **A**cid
TMM	**T**rimmed **M**ean of **M** values
TPM	**T**ranscripts **P**er **M**illion
uppQ	**upp**er**Q**uartile

Introduction

1.1 Background

Since the discovery of nucleic acids as the molecules that carry genetic information, understanding how this information can be stored and used has been one of the main areas of study in the molecular biology field. Unveiling of a gene function remains a major bottleneck in improving our understanding of living systems, by understanding the processes and mechanisms that are occurring in the cell. Classic sequence analyses are centered in discovering what information is inside the cell, revealing all the information that can be potentially used by the cell machinery, focusing on DNA profiling. Beyond classic sequence analysis, an important source of information about the function of a gene is when, where, and how strongly it is expressed. Here is where the Ribonucleic Acid (RNA) gains importance, as it is known as an essential molecule present in every single living cell. Analyzing the identity and amount of each RNA molecule in a certain tissue or cell under specific conditions is the aim of the transcriptomics (Hrdlickova et al., 2016) (FIGURE 1.1). Although in genomics the DNA analysis gives us information about all the genetic information that is in a cell, it is the RNA quantitative analysis which allows us to know which information is being processed and which not.

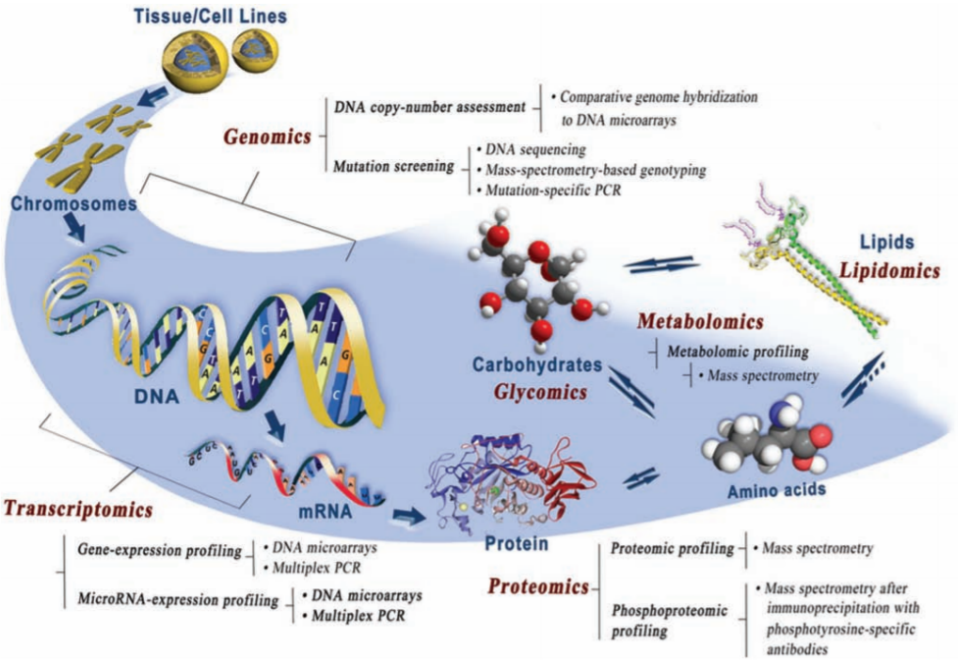

FIGURE 1.1: Schema of omics technologies. DNA (genomics) is first transcribed to mRNA (transcriptomics) and translated into protein (proteomics) which can catalyze reactions that act on and give rise to metabolites (metabolomics), glycoproteins and carbohydrates (glycomics), and lipids (lipidomics) (Wu et al., 2011).

1.2 RNA expression profiling

Many qualitative and quantitative methods for RNA expression profiling analysis have been developed over time. Many of them are no longer in use, however, it is important to know how they performed due to the availability of a vast amount of data coming from these technologies.

In the following subsections I will provide a brief overview of past and current technologies.

1.2.1 Method focused on selected targets

This methods do not take into account the whole transcriptome, they focus on some genes of interest whose expression is quantified very precisely, to the detriment of higher throughput.

The most widely used method is known as quantitative reverse transcription PCR (RT-qPCR) and it has been established as a refernce method for specific gene expression analysis during the past years. This method starts with the isolation of RNA from the cells; followed by cDNA reverse-transcription; then, the cDNA is used as the template for the qPCR reaction which utilizes sequence-specific primers; the production of amplification products can be monitored during each cycle of the PCR reaction, thanks to fluorescent reporter molecules; as a final step, the initial concentration of the selected transcript is calculated based on the exponential phase of the reaction (Wagner, 2013).

This technology has became a mainstream research tool for numerous reasons. There is no need of a post-PCR processing because it is a homogeneous assay. It allows direct comparison between transcripts that differ widely in their abundance due to its huge dynamic range ($>1\times10^7$). The assay takes advantage of the inherent quantitative potential of the PCR, making it a quantitative as well as a qualitative analysis (Ginzinger, 2002).

1.2.2 Gene expression profiling by microarrays

DNA microarrays can simultaneously measure the expression level of thousands of targets (genes and/or transcript) within a particular sample. The key physiochemical process involved in microarrays is DNA hybridization. In these arrays, specific DNA sequences, derived from transcripts are either deposited or synthesized in a 2-D array on a surface in such a way that the DNA is attached to the surface. Lately, fluorescently labeled mRNA sequences which appear in an examined sample hybridize with complementary probes. The amount of target is detected by measuring fluorescent signal intensity.

Most microarrays for RNA applications are centered in the measurement of the expression level of individual genes, however it is well known that the transcriptome of higher organisms is way more complex. *"Alternative splicing, the process by which individual exons of pre-mRNAs are spliced to produce different isoforms of mRNA transcripts from the same gene"* (Xu et al., 2011). This is the major source of protein isoform diversity, which is strongly related with their function and, subsequently, with the biological implication of such mRNA isoforms.

Recently, a new, more sophisticated type of array has been developed. *"This array uses a high-density tiling approach for the measurement of gene and exon expression and genome-wide identification of alternative splicing as well as analysis for coding SNP detection and noncoding transcripts"*, in addition, the exon–exon junction probes of this array were shown to improve the detection of alternative splicing events (Xu et al., 2011).

However, it has also some drawbacks. This technology enables interrogation of transcripts genome-wide, however, it requires a priori sequence information for designing the probes. This, limited the development of microarray technology and its application in some studies. In addition, cross-hybridization and background signals might lead to low specificity or low sensitivity for some genes (Hrdlickova et al., 2016).

1.2.3 RNA-seq: gene expression profiling by (Next Generation) Sequencing

The first method of interrogation of RNA sequences at large scale was based on the partial sequencing of complementary DNA (cDNA) it was also known as expressed sequence tag (EST) method, developed in the early 1990s (Adams et al., 1992). Later, also in the 1990s the Serial Analysis of Gene Expression (SAGE) cut down the cost of expression analysis, thanks to sequencing only a short tag region per cDNA (Velculescu et al., 1995).

Both of these methods are based on Sanger's "chain terminator" technique, also known as Sanger sequencing technology. It was developed by Frederik Sanger and it entailed *"the major breakthrough that forever altered the progress of DNA sequencing technology"* (Heather and Chain, 2016). This technique makes use of dideoxynucleotides (ddNTPs), which are structural analogues of the deoxynucleotides (dNTPs), the monomers of the DNA strands. The ddNTPs have an unique change in their structure, which is lacking of the 3'hydroxyl group that is required for extension of DNA strands, because they cannot longer form a bond with the 5'phosphate of the next dNTP (Chidgeavadze et al., 1984).

The technique requires four different extension reactions, all of them contain the four dNTPs (dATP, dGTP, dCTP and dTTP) and the polymerase. Single type of ddNTP (ddATP, ddGTP, ddCTP, or ddTTP) is added to each tube in a 100-fold lower concentration than the dNTPs. Afterwards, the results of the extension reactions are run on four lanes of a polyacrylamide gel, where DNA strands of each possible length can be observed, allowing to infer the DNA sequence.

"The accuracy, robustness and ease of use led to the dideoxy chain-termination method – or simply, Sanger sequencing – to become the most common technology used to sequence DNA for years"(Heather and Chain, 2016). Some improvements were made to the technique, including the replacement of previous radiolabelling with fluorometric based detection, allowing to perform the sequencing in one reaction instead of four. This technique is still the most precise one (99.999%) and perform considerably long reads 400-900 bp, however its low throughput has relegated it to very few applications.

It was with the massive parallel sequencing, around 2002, when the sequencing technology bursted out. New technologies were developed, known as Next Generation Sequencing (NGS). Some of this brand new techniques were: 454, solexa/Illumina,

SOLiD and IonTorrent. More recently, some other technologies appeared in the market, such as PacBio and OxfordNanopore (ONT), they are known as Third Generation Sequencing, due to their capability of single molecule real-time sequencing (SMRT).

Next Generation Sequencing technologies – brief review

- 454 pyrosequencing (Life Sciences/Roche): it was firstly owned by Life Sciences since 2000 and was acquired by Roche in 2007 and, lastly, was shut down by Roche in 2013. Its development was the milestone that started the NGS era (Heather and Chain, 2016). In this method, template-bound beads are distributed into a PicoTiterPlate along with beads that contain an enzyme cocktail. When a dNTP is incorporated into a strand, an enzymatic cascade occurs, resulting in a bioluminescence signal. A charge-couple device camera detects each single burst of light and can assign it to the incorporation of one or more identical dNTPs at a particular bead. The relatively high average read length (up to 700 bp) and the accuracy 99.9%, made this technology very useful in some applications, however, the problems when reading polybases bigger than 6 repeated nucleotides, the high cost and the low throughput relegated this technology to a second plain (Goodwin et al., 2016). Despite the fact that this technology is not longer available, it is still important due to the considerable amount of data sets produced by it that can still be used.
- Solexa/Illumina: in 2005 Solexa completed sequencing of the Bacteriophage phiX-174 genome, the same that Sanger sequenced for the first time, however, this new technology achieved a considerably higher amount of data, reaching 3 million bases in a single run. In 2007 it was acquired by Illumina, and nowadays it is the most widely used sequencing technology. Briefly, the process starts with a preparation of DNA library by fragmentation of the DNA, followed by the adapter ligation. Adapter-ligated fragments are then amplified in the PCR reaction. The library is loaded into the flowcell, where the fragments are captured by oligos complementary to the library adapters. During each cycle, a mixture of all four individually labelled and 3'-blocked deoxynucleotides (dNTPs) are added. Then the strand grows by single nucleotide and then unbound dNTPs are removed and the surface is imaged to identify which dNTP was incorporated to each cluster. Then, the nucleotide is unblocked, the fluorophore is removed and a new cycle can begin (Heather and Chain, 2016). The maturity as a technology, a high level of cross-platform compatibility and its wide range of platforms has made Illumina the dominator of the short-read sequencing industry. *"The suite of instruments available ranges from the low-throughput MiniSeq to the ultra-high-throughput HiSeq X, which is capable of sequencing around 1,800 human genomes to 30x coverage per year"* (Goodwin et al., 2016). Further diversification is derived from the many options available for runtime, read structure and read length (up to 300 bp). Currently the availability of platforms offered by Illumina are (from smaller to bigger throughput) iSeq, MiniSeq, MiSeq, NextSeq, HiSeq and NovaSeq, with outputs varying form 1.2 Gb to 6000 Gb. Although it has an overall accuracy rate of >99.5%, the platform does display some under-representation in AT-rich and GC-rich regions, as well as a tendency towards substitution errors (Goodwin et al., 2016).

- SOLiD: the sequencing by oligonucleotide ligation and detection was firstly launched by Applied Biosystems (which became Life Technologies following a merger with Invitrogen). As the name suggests, it is different to the previous techniques regarding the main reaction that take part in the sequencing process, here is the ligation (sequencing by ligation), which uses a DNA ligase to elongate the new DNA strand adding dinucleotides, instead the polymerase used in the sequenced by synthesis methods (Heather and Chain, 2016). It utilizes two-base-encoded probes, in which each fluorometric signal represents a dinucleotide. Thus, the direct output is not associated with the incorporation of a known nucleotide. There are 16 different combinations of dinucleotides, and it is not possible to label them with this many various types of fluorophores, therefore four different fluorophores are used, each of them representing a subset of four different dinucleotides. Consequently, each ligation signal represents one of several possible dinucleotides, leading to the term colour-space (rather than base-space), which must be decoded during data analysis. The SOLiD sequencing procedure is composed of a series of probe–anchor binding, ligation, imaging and cleavage cycles to elongate the complementary strand. Over the course of the cycles, single-nucleotide offsets are introduced to ensure every base in the template strand is sequenced. The Sequencing By Ligation (SBL) technique used by SOLiD affords this technology a very high accuracy (99.99%), as each base is probed multiple times. Although accurate, this platform also shows evidence of a trade-off between sensitivity and specificity, such that true variants are missed while few false variants are called. Its maximum read length and its long runtimes (several days) have relegated this technology to a small niche within the industry (Goodwin et al., 2016).

- Ion Torrent: Launched by Torrent Systems Inc. in 2010 it is known as the first 'post-light-sequencing' sequencer, as it uses neither fluorescence nor luminescence (Rothberg et al., 2011). Similar to 454, beads containing specific colonies of DNA templates (generated by emulsion PCR) are distributed over a picowell plate (Heather and Chain, 2016). Rather than using an enzymatic cascade to generate a signal, the Ion Torrent platform detects the H+ ions that are released as each dNTP is incorporated. The resulting change in pH is detected by an integrated complementary metal-oxide-semiconductor and an ion-sensitive field-effect transistor. One of the drawbacks of this technology is that homopolymer regions are problematic, as in 454, the incorporation of multiple nucleotides at the same time produces a signal imperfectly proportional to the number of bases incorporated, leading to errors when measuring homopolymers larger than 6–8 bp (Goodwin et al., 2016).. Insertion and deletion (indel) errors dominate, although the overall error rate is on par with other NGS platforms in non-homopolymer regions (Loman et al., 2012).

- PacBio: it was started by Pacific Biosciences in 2010 and it is based on the single molecule real time sequencing (SMRT). This technology uses a specialized flow cell that contains several thousands of transparent individual picolitre wells. In contrast to the previous technologies the polymerase is fixed to the bottom of the well and it is the DNA molecule which progresses through the enzyme. The fact of having the polymerase fixed allows to visualize continuously the location of incorporation of the new nucleotide using a laser and camera system that records the color and duration of emitted light as the labeled nucleotide momentarily pauses during incorporation. As the dNTP is incorporated the

fluorophore is cleaved from the nucleotide and it diffuses away from the sensor. The SMRT platform utilizes circularized DNA templates, allowing to read the same template multiple times in the same sensor, improving considerably the quality of the sequencing by forming a consensus sequence. This is only possible for DNA templates shorter than 3kb (Eid et al., 2009). It can generate reads up to 50 kb long and 10-15 kb on average using a long-insert library. However, it has also some limitations. The single-pass error rate for long reads is as high as 15% with indel errors dominating (Carneiro et al., 2012). The use of circular DNA and deeper coverages can mitigate the error, increasing the accuracy up to 99.999% for insert sequences derived from at least 10 subreads. The limited throughput, the elevated costs of PacBio RS II (around $1000 per Gb) and the need for high coverage, make impossible for small laboratories to afford this technology (Goodwin et al., 2016). A new system known as Sequel, which has more modest features and, thus, a more affordable pricing, is becoming popular in some smaller laboratories.

- Oxford Nanopore: in 2014 the first nanopore sequencer — the MinION from Oxford Nanopore Technologies (ONT) — became available. All the previous sequencing platforms monitor one way or another the incorporation of nucleotides to a template DNA strand. In contrast, nanopore sequencers directly detect the DNA composition of a native ssDNA molecule, while it is guided through a nanopore protein membrane, where a voltage is continuously applied. The current undergoes different changes depending on the nucleotides that pass through it, and this information can be processed to determine the sequence of the DNA molecule that passes through the pore (Goodwin et al., 2016). Theoretically it has no limitation in the read length, although in practice there are some constraints with ultra-long fragments (Goodwin et al., 2015). *"As a consequence of the unique nature of the ONT technology, in which there are more than 1,000 distinct signals, ONT MinION has a large error rate – up to 30% – and is dominated by indel errors"* (Goodwin et al., 2016). Homopolymers are also a pending issue for ONT MinION. Modified bases have slightly different output when read, making the platform more prone to mistakes. Fortunately, recent improvements in the chemistry and the base calling algorithms (mostly thanks to *crowd sourcing science*) are improving accuracy.

Novel approaches brought new challenges, producing huge amounts of data and making development of new data analysis approaches necessary .

1.2.4 Comparison between RNA-microarrays and RNA-seq

Currently, both of the above described high-throughput approaches, high-density microarrays and RNA-seq, are widely used in order to perform gene expression profiling. It has been thoroughly discussed which one of them has better overall performance in terms of specificity, sensibility and reproducibility.

Many different statements have been said. First approaches concluded that *"RNA-seq was more sensitive than microarrays, with genes detected only by RNA-seq being in the lowest range of expression levels"*(Sultan et al., 2008) or *"RNA-seq provided more accurate estimation of absolute transcript levels"*(Fu et al., 2009), giving more credibility to RNA-seq, although, always maintaining that both platforms provided very similar data with high correlation.

However, more recent and, often, better designed studies have shown that both approaches have their own pros and cons. These studies concluded that *"RNA-seq was more sensitive in detecting genes with low expression levels"* (Su et al., 2011), however, it shows a higher technical variability, requires more material to be performed and the data analysis takes more time (Xu et al., 2011). While, *"a higher percentage of differentially expressed genes was identified by microarrays"* (Mooney et al., 2013), nonetheless, they have a lower coverage of the transcriptome due to their limited gene models, and are less prone to detect new transcript variants. Despite the differences, all of the research concluded that the expression profiles generated using both methods were highly correlated in terms of relative expression. Taking all this into account, it is suggested to combine both of the approaches to get a comprehensive base from which conclusions can be drawn (Su et al., 2014).

1.3 Approaches in analyzing RNA-Seq data

There are multiple ways to analyze the millions of reads produced by an RNA-seq experiment. It is a powerful technology with many applications, ranging from gene and splice variant discovery to differential expression analysis, detection of fusion genes, variants, and RNA editing. This makes impossible to have an unique workflow scheme that can cover all of them (Korpelainen, 2015). I will explain the major steps, specially focusing on the differential expression analysis approach.

With the purpose of addressing this analysis, multiple bioinformatic tools have been developed, many of them are available as a standalone software written in different languages (Java, Perl, C++, Python, etc.). Most of them work in a Linux environment and require some knowledge about the Unix terminal. There are also some graphical wrappers which allow the user to work in a much more intuitive and user-friendly interface (e.g., Galaxy or Chipster). However, this increase in comprehensibility, drives a detriment in control over the data analysis, flexibility to change parameters, use all available options, and import and export data to different tools that may be standard or not. It is important to note that this is a very fast developing field with tens or even hundreds of tools being developed every year (Korpelainen, 2015). The major steps can be observed in the FIGURE 1.2.

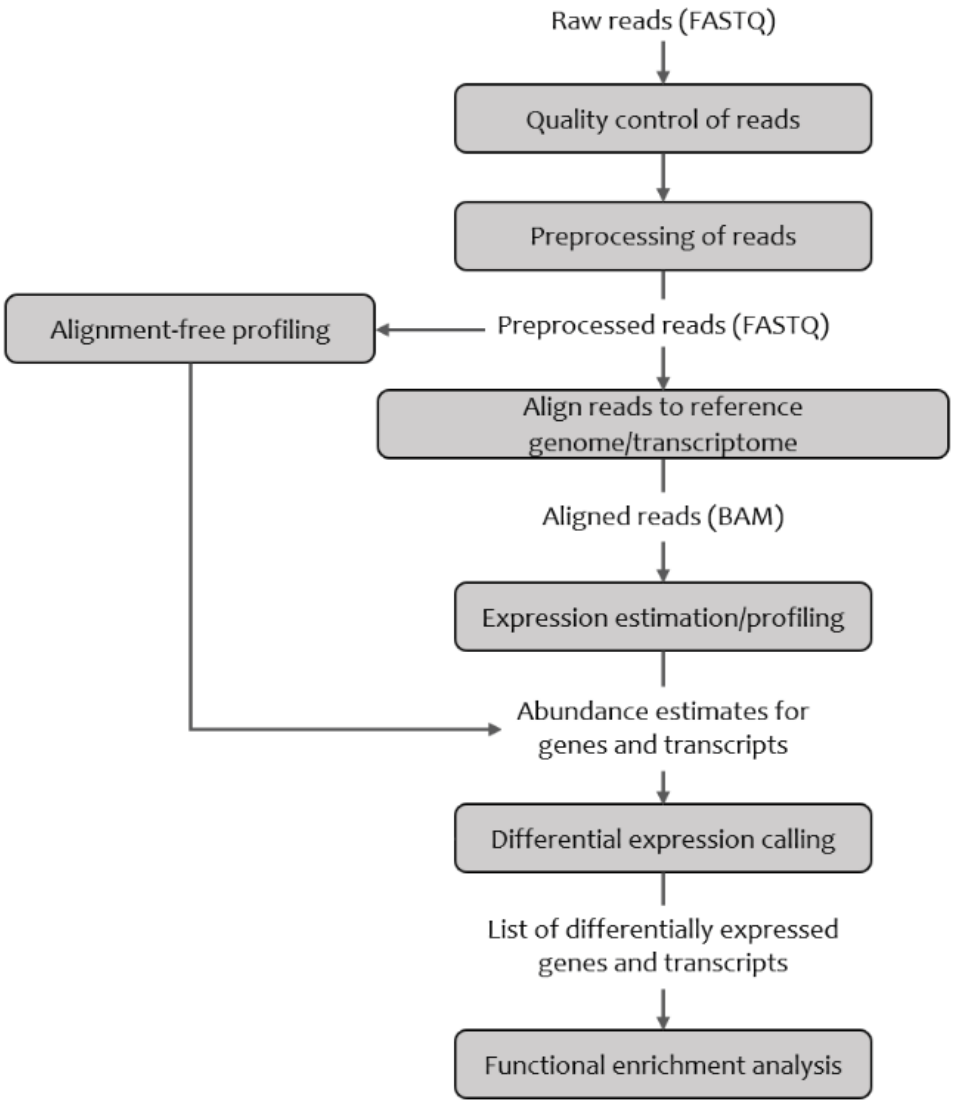

FIGURE 1.2: Differential expression analysis workflow consists of several, interrelated steps. The typical output file formats are indicated in parentheses.

1.3.1 R language overview

R is a programming language and environment specially focused on statistical computing and graphics. It is similar to the S language and environment (R can be considered as a different implementation of S), which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. It is free software, which allows it to be used widely, and, thus, promoting the development of new free source packages, enriching the amount of tools available (r-project.org, 2018).

Many of the bioinformatic tools used in RNA-seq analysis have R wrappers and are included in the Bioconductor project, which means that whole analysis pipeline can be done using R. Bioconductor is an open source, open development software project to provide tools for the analysis and comprehension of high-throughput genomic data (bioconductor.org, 2018a).

The widespread access to a broad range of powerful statistical and graphical methods for the analysis of genomic data and the great amount of high-quality documentation make R and Bioconductor a perfect choice to perform this work.

1.3.2 Quality Control and Preprocessing

Raw reads from the sequencer (FASTQ file format) often have low-quality bases and artifacts that need to be removed in order not to interfere in the downstream analysis. Quality problems include untrimmed adapters, library construction sequences, poly-A tails, sequence-specific bias, 3'/5' positional bias, polymerase chain reaction (PCR) artifacts and sequence contamination. These problems interfere and bias the posterior analyses such as mapping to reference, expression estimation and assembly. However, a proper preprocessing including filtering and/or trimming can correct them. There are also approaches to identify and partially remove the hidden confounding factors (Labaj and Kreil, 2016)(Su et al., 2014).

The confidence in the base call is indicated by the base quality. This quality is expressed in Phred scale, where *"log 10 is taken of the probability that the base is wrong, and multiplied by -10"* (Korpelainen, 2015). For instance, if there is 1 in 1000 chance the base to be wrong, the Phred value is $q = -10 \log_{10} 0.001 = 30$. These values normally range between 0 and 40. In order to save space, ASCII characters are used instead of numbers in the FASTQ files. The quality of the bases tend to decrease in the later cycles of sequencing (Korpelainen, 2015).

There are two non-exclusive approaches to cope with low-quality bases. They can be either filtered, removing the entire read, or trimmed, removing just the low-quality ends of reads. There is no an agreement on which should be the optimal quality threshold for trimming in the context of RNA-seq. Both *de novo* assembly and the alignment of reads to a reference can be ameliorated by trimming low-quality bases, however, it entails a reduction in the coverage because trimmed reads are shorter and there are less of them. Deciding what quality threshold to choose is an equilibrium between the two.

There are numerous tools that can be used for reading quality control and preprocessing. Tools for checking read quality include `FastQC` and `PRINSEQ`, which allow to examine various quality metrics and are able to provide reports with informative visualizations. The `PRINSEQ` package also offers filtering and trimming functionality. Other preprocessing tools include `Trimmomatic`, `Cutadapt`, and `FastX`, just to name a few (Korpelainen, 2015).

1.3.3 Mapping

The next step in the analysis pipeline is the alignment of the sequences against the reference genome or transcriptome to search out where the read originated from. This is known as 'mapping' and it gives us information about the genomic location, thus, allowing us to discover novel genes and transcripts. When there is not a reference genome or only known transcripts are to be quantified, reads are mapped to a transcriptome. This limits the discovery of new transcript variants. However, using the genome instead of the transcriptome bring several challenges: there are millions of short reads, genomes are huge and they contain nonunique sequences such as repeats and pseudogenes. Moreover, aligning algorithms need to deal with

mismatches and indels caused by sequencing errors and genomic variation. In addition, the introns present in the genes of many organisms make necessary to map RNA-seq reads to the genome in a noncontiguously way. Coping with spliced reads and determining exon–intron boundaries correctly is difficult, due to the limited sequence signals at splice sites and the length of the introns, that reach thousands of bases long (Korpelainen, 2015).

Number of alignment tools are available, offering different approaches to handle these challenges. Aligners use different heuristic approaches and indexing schemes to accelerate the process. Some of them take into account base quality when a mismatch occur, as well as, expected distance and relative orientation of paire-end reads. Confidence in the mapping location is expressed as mapping quality ($Q = 10log_{10}P$, where P is the probability that the read originated elsewhere) and it depends mostly on the uniqueness of the mapping. Multi-mapped reads can be distributed proportionally to the coverage between the equally matching positions by some aligners. When dealing with spliced reads, aligners perform an initial alignment to discover exon junctions and, then they complete the alignment. Aligners can use genomic annotation, when available, for placing spliced reads (Korpelainen, 2015).

When choosing an aligner for RNA-seq the more important consideration is whether spliced alignment is needed or not. The absence of introns or if microRNAs were sequenced, allows to use aligners originally developed for DNA, such as `Bowtie` or `BWA`. They can also be used when mapping to a transcriptome instead of a genome. Nevertheless, if introns are present in the genome, a spliced aligner such as `TopHat2`, `STAR`, `Subread` or `HISAT` is necessary. Different aligners are used for different purposes. DNA aligners are used for purposes such as whole genome sequencing or whole exome sequencing. RNA aligners are used when spliced reads need to be mapped against the genome. However, DNA aligners can be used for RNA-Seq when aligning to the transcriptome.

`TopHat2` (Kim et al., 2013) stands out for its speed and memory efficiency. It uses `Bowtie2` as its alignment engine and it is optimized for reads equal or longer than 75 bp. The `TopHat2` approach includes a multi-step alignment process which first aligns reads to the transcriptome and, if genomic annotation is available it aligns the non-aligned reads to the genome. This improves alignment accuracy, avoids absorbing reads to pseudogenes, and speeds up the overall alignment process. If the read ends do not align `TopHat2` does not trim them. Therefore, it leads to a low tolerance for mismatches, so reads with low-quality bases might not align well. Finally, `TopHat2` can be used to detect genomic translocations, as it can align reads across fusion breakpoints. Although still widely used and considered as a state-of-the-art approach, it has been discountinued in early 2016 and replaced by `HISAT2`.

`STAR` (Dobin et al., 2013) (Spliced Transcripts Alignment to a Reference) is another spliced alignment program which runs very fast. The drawback is that it needs considerably more memory than TopHat. Speed is not the only one of its advantages. It is able to perform an unbiased search for splice junctions because it does not need any prior information on their locations, sequence signals, or intron length. `STAR` can also align a read that contains various splice junctions, indels and mismatches, and low-quality ends can be managed. Finally, it can map long reads and even full-length mRNA, which is required as read lengths are increasing. It also provides so called '2-pass mode' with increased sensitivity for novel exon-exon junctions discovery.

`Subread` (Liao et al., 2013) strategy chooses the mapped genomic location for the read directly from the seeds. It utilizes a big amount of short reads (subreads) coming from each read and allows all the seeds to vote on the optimal location. Then, more conventional algorithms are used to align the complete read that is more likely to be the optimal. This is a rapid strategy because the overall genomic location has been chosen before the detailed alignment is done. It is sensitive because individual subreads are not constrained to map close by other subreads and no individual subread is required to map exactly. The final location must be supported by many different subreads, which increases the accuracy. The strategy extends easily to find exon junctions, by locating reads that contain sets of subreads mapping to different exons of the same gene. It scales up efficiently for longer reads.

`HISAT/HISAT2` (Kim et al., 2015) (hierarchical indexing for spliced alignment of transcripts) is a newer and high efficient system. *"`HISAT` uses an indexing scheme based on the Burrows-Wheeler transform and the Ferragina-Manzini (FM) index, employing two types of indexes for alignment: a whole-genome FM index to anchor each alignment and numerous local FM indexes for very rapid extensions of these alignments"* (Kim et al., 2015). `HISAT` was designed as a successor to `TopHat2` and has compatible outputs, but runs approximately 1-2 orders of magnitude faster (Pertea et al., 2016).

1.3.4 Expression estimation/profiling

Once the reads are mapped to the genome, genomic annotation can be inferred from their location. This makes possible to quantify the gene expression by counting reads per genes, transcripts and exons. Intuitively, the calculation of the mapped reads gives us a straight way to estimate transcript abundance, however, in practice, several factors need to be taken into account. Due to alternative splicing, Eucaryotic genes may produce diverse transcript isoforms. Because of common or overlapping exons in those transcripts longer reads are better for their quantitation and assembly. Moreover, the coverage along the transcripts is not uniform as a result of data generation and processing biases. In order to address these challenges a simplified approach is used, where expression is often estimated only at the gene or exon level. Still, gene level counts are not the optimal option for differential expression analysis. For the complex genes with multiple alternative transcripts the simple reads covering the gene locus approach is heavily biased.

As long as an annotated reference genome is available, mapped reads can be counted based on their genomic features depending on the location information. *Ab initio* assemblers can produce annotation files that allow to quantify new genes and transcripts. Instead, and particularly when there is no reference genome available, the transcriptome can be used to map and count the reads. If there is no reference transcriptome either, you can assemble one using a *de novo* assembler.

Many different tools can be used for counting, such as, `HTSeq` , `BEDTools` , and `Qualimap`. Also some Bioconductor packages such as `Rsubread` (R wrapper for `Subread`) and GenomicRanges offer counting functionality. The input for these tools is genomic read alignments in SAM/BAM format and genome annotation in GFF/GTF or BED format. What makes them different is the way each one of them deal with multimapping reads (reads which map to several genomic locations due to homology or sequence repeats): `HTSeq` ignores all these multireads, `Qualimap` divides the counts equally between the different locations, and `Cufflinks` has an

option to divide each multimapping read probabilistically based on the abundance of the genes it maps to.

A more sophisticated and robust method for transcript quantification is also available. It uses a general linear model for transcript quantification that *"leverages reads spanning multiple splice junctions to ameliorate identifiability"* (Huang et al., 2013). Knowing that, RNA-seq reads sampled from the transcriptome have unknown position-specific and sequence-specific biases, this method simultaneously learns bias parameters during transcript quantification in order to ameliorate accuracy. Also it takes into account that a candidate set of isoforms is provided for transcript quantification, while not in all types of tissue, or condition all of them are prone to be expressed. *"Resolving the linear system with LASSO (least absolute shrinkage and selection operator), this approach can infer an accurate set of dominantly expressed transcripts while existing methods tend to assign positive expression to every candidate isoform"* (Huang et al., 2013).

1.3.5 Alignment-free expression profiling

There is also an alternative approach that allows to bypass the mapping step. There are some tools able to estimate the gene counts directly from the reads. They are known as ultra fast RNA-seq quantitation methods. Such strategies are way faster, they can perform the same analysis in significantly less time.

`kallisto`(Bray et al., 2016) is a program for quantifying abundances of transcripts from RNA-Seq data, or more generally of target sequences using high-throughput sequencing reads. It needs a reference transcriptome to use it as its target. It uses *pseudoalignment* for rapidly determining the compatibility of reads with these targets, without the need for alignment. Pseudoalignment of reads preserves the key information needed for quantification, this makes their developers to state that *"`kallisto` is therefore not only fast, but also as accurate as existing quantification tools"*.

`Salmon`(Patro et al., 2017) is a tool for ultra fast transcript quantification from RNA-seq data. As all of these tools, it requires a set of target transcripts (either from a reference or *de-novo* assembly) to quantify. The *quasi*-mapping-based mode of `Salmon` runs in two phases; indexing and quantification. The indexing step is independent of the reads, and only need to be run once for a particular set of reference transcripts. The quantification step is specific to the set of RNA-seq reads and is thus run more frequently. `Salmon` combines a new dual-phase parallel inference algorithm and feature-rich bias models with an ultra fast read mapping procedure making it one of the preferred options in terms of alignment-free expression profiling.

1.3.6 Differential expression calling

Differential expression (DE) analysis consist of the comparison of different groups of samples in order to identify the genes, transcripts or exons that are expressed in significantly different quantities. These groups of samples have different conditions, such as biological variations (drug-treated vs. controls), diseased vs. healthy individuals, different tissues, different stages of development, etc.

RNA-seq data are in the form of discrete counts generated from a sampling process, oppositely, microarray measurements are continuous (amount of fluorescent signal).

One aspect of this is that, *"because RNA-seq is a sampling procedure, there is a certain amount of "real estate" (the total number of all reads from the sequencing instrument) that the actual transcripts in the sequencing library have to share"* (Korpelainen, 2015). So, highly expressed and long transcripts constitute a great amount of the sequencing library, thus, in a shallow sequencing experiment genes with low abundance may not appear in the final data even though they were in the sample (Labaj and Kreil, 2016). An attractive feature of RNA-seq, though, is the possibility to re-sequence the same library to potentially recover more low expressed transcripts. Another approach is sort of combining power of microarrays and NGS in targeted RNA-Seq, where selected targets are enriched in the library prior sequencing.

Currently there is not a consensus about which method for RNA-seq DE analysis is the best. It is a field in constant development. There are active discussions about the best ways to normalize, and the approach to address the analysis, either at the gene or at the isoform level. Even the measurement unit to use for reporting gene expression levels is a theme of debate (Labaj and Kreil, 2016).

Normalization methods and removing confounding factors

Sequencing depth and transcript length are the most intuitive factors that affect the number of reads generated per transcript. Nevertheless, some less obvious factors such as transcriptome composition, Guanine-Cytosine (GC) bias, and sequence-specific bias (caused by random hexamers) also influence the amount of reads. When the aim is to compare read counts between different samples, all the factors need to be taken into consideration. To do so, different normalization methods can be applied and the choice depends on the posterior application.

Quantization tools normally outputs abundances in either raw counts or in FPKM (Fragments Per Kilobase per Million mapped reads). Another normalization is transcripts per million (TPM). TPM measures the proportion of transcripts in the pool of RNA, it takes into account the length, calculating first the counts per base and posteriorly dividing the number by the sum of all rates. The final step is scale by one million because the proportion is often too small. *"FPKM takes the same rate we discussed in the TPM and instead of dividing it by the sum of rates, divides it by the total number of reads sequenced and multiplies by a big number (10^9)"* (haroldpimentel.wordpress.com, 2018). FPKMs and TPM are mainly used for abundance reporting purposes, while raw counts and other approaches are used in differential expression analysis.

In this particular work, the main goal is to compare different DE calling tools, thus raw gene counts are going to be used and FPKM normalization is not used directly. In cases like this other normalization methods need to be applied, some of which can use internally some of this scaling factors, such as FPKM, TPM or counts per million (CPM). The most popular tools provide different normalization options and in this particular work four of them are going to be compared:

- Upper Quartile (uppQ), which firstly removes all the counts that are equal to 0, afterwards it calculates the scaling factors from the 75Th percentile of the counts for each library.
- Trimmed Mean of M-values (TMM), is based on the hypothesis that most genes are not differentially expressed, it trims the upper and lower percentage of the data and does not use them to calculate the scaling factors.

- Relative Log Expression (RLE), median library is calculated from the geometric mean of all columns and the median ratio of each sample to the median library is taken as the scale factor.
- Scaling factors can be set to 1 if 'None' method is applied.

Unlike TPM and FPKM, methods like TMM and RLE are batch normalization methods, that is, they are not designed for use on a single sample, but on a group of samples. While TPM and FPKM are local to the sample and not affected by other samples, the correction factors from TMM and RLE normalization should be recalculated each time the a sample is included to or removed from the data. Another difference is that RLE and TMM does not take into account the length of the transcript. When a DE analysis is performed this does not matter, due to the fact that the comparisons are being made between the same transcripts across conditions and different transcript abundances are not compared to each other, so he transcript length is always the same.

DE calling R-Bioconductor packages

Many distinct tools can be used to perform DE analysis. If the goal is to focus on exons `DEXSeq` is the most used tool. When isoforms are compared `BitSeq`, `Cuffdiff` or `ebSeq` are normally used. While if there is a gene-focused approach `DESeq2`(Love et al., 2014), `edgeR`(Robinson et al., 2010) and `limma`(Smyth, 2004) are the three main tools currently and all of them can be found in Bioconductor (Korpelainen, 2015). The PDF reference manuals and tutorials or "vignettes" for `DESeq2`, `limma`, and `edgeR` contain a lot of useful information. These three packages are the ones that are going to be compared in this work.

1.3.7 Functional enrichment analysis

Once the DE analysis is completed, a list of the differentially expressed genes is obtained. The list *per se* does not provide much information. Additional information, in form of annotation, about these genes is needed. This annotation refers to the process of identifying and locating the genes and other functional elements of an organism's genome and attaching some notes of their functions to them. Normally these annotations that are linked to a gene are: the genomic location (chromosome, cytoband, base pairs), the exonic and intronic structure, the transcripts, and some functional information. This functional concepts are usually in form of controlled vocabulary, such as Gene Ontology (GO), or the metabolic pathways that the translated proteins function in (e.g., KEGG, the Kyoto Encyclopedia of Genes and Genomes; Reactome, a database of reactions, pathways, and biological processes) (Korpelainen, 2015).

Having this annotation information of our particular list of genes, the functional enrichment analysis can be performed. It basically consists in observe if any of the notions present in our DE genes is statistically significantly enriched in respect to the others. Many different algorithms and tests can be applied to determine it. Some of them will be explained later.

Gene Ontology

An ontology formally represents knowledge as a set of concepts within a domain, and the relationships among those concepts. It can be used to reason about the entities within that domain and may be used to describe the domain. They allow organization of data within a database, linking between databases and searches across databases. The Gene Ontology (GO) is a way to capture biological knowledge for individual gene products in a written and computable form which consist of a set of concepts and their relationships to each other arranged as a hierarchy. GO is actually made up of three different ontologies:

- Cellular Component: it covers the part of a cell or its extracellular environment in which a gene product is located. A gene product may be located in one or more parts of a cell.
- Molecular Function: this ontology describes the actions of a gene product at the molecular level, such as catalysis or binding. A given gene product may exhibit one or more molecular functions.
- Biological Process: it involves those processes specifically pertinent to the functioning of integrated living units: cells, tissues, organs, and organisms. A process is a collection of molecular events with a defined beginning and end.

GO enrichment analysis

Different tools are available to perform the GO enrichment analysis. Some examples are `AmiGO`, `Babelomics`, `TopGO` and `RuleGO`. The one that is used in this work is `TopGO` and some of the different algorithms that can be chosen are compared. These are:

- *classic*: each GO term is tested independently, not taking the GO hierarchy into account.
- *elim*: this method processes the GO terms by traversing the GO hierarchy from bottom to top, ie. it first assesses the most specific (bottom-most) GO terms, and proceeds later to more general (higher) GO terms. When it assesses a higher (more general) GO term, it discards any genes that are annotated with significantly enriched descendant GO terms (considered significant using a pre-defined P-value threshold). This method does tend to miss some true positives at higher (more general) levels of the GO hierarchy.
- *parent-child*: when assessing a GO term, it takes into account the annotation of terms to the current term's parents, and so reduces false positives due to the inheritance problem.

1.4 Reproducibility in expression profiling analysis – Sequencing Quality Control (SEQC/MAQC-III) consortium

The variety of options available in order to perform transcriptomic analysis is huge nowadays, both in terms of technology and data processing new options are emerging each year. This overwhelming assortment claims for some benchmarking in order to make choosing the correct technology and analysis pipeline more adequate to

the application they are applied for.

With this objective in mind, the US Food and Drug Administration (FDA) has coordinated the Sequencing Quality Control project (SEQC/MAQC-III), a large-scale community effort to assess the performance of RNA-seq across laboratories and to test different sequencing platforms and data analysis pipelines (Su et al., 2014). Specifically, three RNA-seq platforms (Illumina HiSeq, Life Technologies SOLiD, and Roche 454) were tested at multiple sites for reproducibility, accuracy, and information content. The project also extensively compared RNA-seq to microarray technology and evaluated the transferability of predictive models and signature genes between microarray and RNA-Seq data. The impact of various bioinformatics approaches on the downstream biological interpretations of RNA-seq results was also comprehensively examined and the utility of RNA-seq in clinical application and safety evaluation was assessed. The project was completed by the end of 2014.

The results of the study showed that there is **no technological gold standard** in expression profiling. Moreover, this study as well as follow up work has also shown that there is also noticeable discordance between results of RNA-Seq data analysis approaches.

One of the result of consortium work was a very rich benchmarking data set (DataSet of accesion number GSE47774). This DataSet is used to perform the analysis necessary for this work.

Objective: Deeper look into DE calling

The SEQC-III/MAQC consortium concluded that there is no technological gold standard in RNA-seq nor in expression profiling in general and that the different approaches chosen to analyze the data impact the final result (Su et al., 2014). The goal of this Bachelor Thesis is to look deeper into this second statement and focus on three tools for differential gene expression calling. These are `DESeq2`, `edgeR` and `limma`. Two different parallel objectives are raised:

- Address the comparison among the different normalization methods available in the `edgeR` and `limma` packages.
- Perform the comparative analysis of the three packages running with their default parameters.

Both of the approaches can be divided into three major milestones:

- Compare the genes detected as differentially expressed.
- Identify the areas of discordance at the functional enrichment level, comparing the Gene Ontology terms enriched in each case.
- Investigate the propagation of discordance from one level to another.

Materials and Methods

3.1 Design

3.1.1 Comparison of normalization methods

As explained in the introduction, `limma` and `edgeR` allow the user to choose between four different normalization methods, which are Upper Quartile (uppQ), Trimmed Mean of M-values (TMM), Relative Log Expression (RLE) and 'None'.

The four different methods were applied to calculate the scaling factors. Afterwards, the analysis of the differential expression was performed following the same procedure in all the four cases. This assures that the differences obtained are certainly due to the difference introduced by the normalization method that has been used in each case.

Posteriorly, a comparative at the GO terms enrichment level was performed, allowing to infer if the results are more similar at the annotation level. In this step three different algorithms (*elim*, *parent-child* and *classic*) were applied to compare the normalization methods among them. Then, the Biological Process ontology was utilized to perform the functional enrichment analysis.

3.1.2 Comparison of DE calling packages

The approach is very similar to the normalization methods. In this case, instead of comparing normalization methods, the DE calling packages `limma`, `edgeR` and `DEseq2` were compared.

All of them were ran with their default settings. They were also compared at the gene and functional enrichment level in a parallel way to the above.

3.2 Data source (DataSet of accesion number GSE47774)

This is a DataSet that was published on Aug 08, 2014, and it was obtained by the SEQC project described above. The aim was to exterminate Illumina HiSeq (HiSeq 2000), Life Technologies SOLiD (AB 5500 Genetic Analyzer) and Roche 454 (GS FLX Titanium) platforms at multiple laboratory sites using reference RNA samples with built-in controls, assessing RNA sequencing (RNA-seq) performance for sequence discovery and differential expression profiling and compare it to microarray and quantitative PCR (qPCR) data using complementary metrics (NCBI, 2018).

There were four different reference RNA samples. Samples A (pooled human cell lines of different tissues - Universal Human RNA Reference - UHRR) and B (pooled

human cell lines of different brain structures - Human Brain RNA Reference - HBRR) from the MAQC consortium, adding spike-ins of synthetic RNA (ERCC). Samples C and D were then constructed by combining A and B in known mixing ratios, 3:1 and 1:3, respectively. They were distributed to several independent sites for library preparation and sequencing (NCBI, 2018).

This produced a large DataSet containing loads of counts of different genes, performed by different platforms. Read alignment and summarization were performed using programs included in `Subread` package. In this work, I only focused in the ones obtained using Illumina HiSeq 2000, concretely the counts of the samples B and D.

3.3 Bioinformatic analysis

For each of the cases already explained in the subsection 3.1.1, comprising all the normalization methods and packages, a similar analysis pipeline were performed. In the FIGURE 3.1 the overall workflow can be observed.

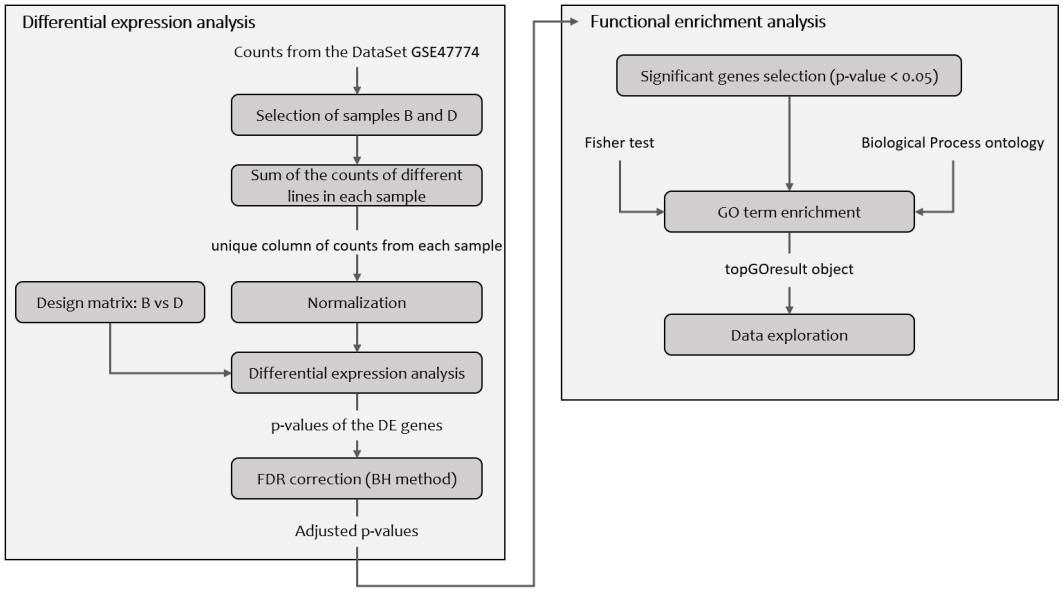

FIGURE 3.1: The bioinformatic workflow included several steps. It was divided in two different analyses: the DE analysis and the functional enrichment analysis.

All the analyses started by filtering the data coming from the DataSet that was mentioned in the section 3.2. The samples were divided in four different replicates, each of which was at the same time divided in seven different lines (making seven different columns for each replicate). The samples B and D were selected and the counts of all the lines each replicate were summed up, to get an unique column of counts from each sample.

The next step was to create a design matrix in which the comparison, that was going to be made, was established. In this case it was a very simple design, in which sample B was compared with the sample D.

Then, the normalization method was applied in order to calculate the scaling factors. The scaling factors are numbers, calculated by the normalization method, by which

each of the samples have to be multiplied in order to normalize the counts among samples. This normalization corrects diverse biases as explained in the subsection 1.3.6.

Once the replicates have been normalized and the design matrix was prepared, the proper analysis starts. Each package has its own methods to determine whether a gene is DE between the two conditions. The package calculates the probability of each gene to be DE and it calculates a statistic known as p-value, which is, formally, *"the probability for a given statistical model that, when the null hypothesis is true, the statistical summary (such as the sample mean difference between two compared groups) would be the same as or of greater magnitude than the actual observed results"* (Hubbard and Armstrong, 2006). In this case, this means that the p-value is the probability of a gene of being called as DE by chance, so, the lower is the p-value, the more confidence we can have in this gene being truly DE.

Next, the adjusted p-value was calculated. A bias appears when the p-value is calculated for multiple test, as it is the case. Multiple testing leads to a type I error, which means the occurrence of false positives. To correct this, the false discovery rate (FDR) method is applied. There are different ways to implement the FDR, in this case the Benjamini–Hochberg (BH) method is applied in all the cases and this gives an adjusted p-value, which in further sections of this work will be referred to, simply, as p-value.

The final output of the tools is a table, where each gene is in a row and several parameters related to the gene are in the correspondent columns. Among them, the adjusted value (from now on p-value); the average counts, mean of counts among samples; the fold-change, the ratio between the expression of B and D; and the logarithm of the fold-change, which helps to visualize the change between conditions. Only the genes with a p-value less than 0.05 are considered to be statistically significant DE.

Subsequent steps comprise the functional enrichment analysis. It starts defining a function that decide which genes are going to be taken into account as DE during the analysis. The condition was them to have a p-value less than 0.05. Then the `TopGO` package was used to perform the analysis. It was set to use the Biological Process ontology (see subsection 1.3.7), and was run three times, one per algorithm, as explained in the section 3.1. The objective of the analysis is to calculate if a GO term is overrepresented in the set of genes that are DE respect to the total of genes, to do so the fisher test was selected. Fisher test, formally, *"is one of a class of exact tests, so called because the significance of the deviation from a null hypothesis (e.g., P-value) can be calculated exactly, rather than relying on an approximation that becomes exact in the limit as the sample size grows to infinity, as with many statistical tests"* (Fisher, 1922).

The output is a topGOresult object from which we can access all the information relative to the analysis. We can extract the significant GO terms, the ones which have a p-value less than 0.05, from the output. Another features, such as the number of genes annotated in a particular term or the depth in the hierarchy of the GO ontology can also be obtained.

The bioinformatic analysis *per se* concluded here, but further exploratory analysis of the results needed to be performed in order to drawn some conclusions from them.

3.3.1 Exploratory analysis

Since the main objective of this work is to compare the different cases, most of the exploratory analysis is centered in using different graphic representations that contrast them.

One of the most useful tools, that sums up the results is the Venn diagram. Its interpretation is quite straightforward, each case is represented by an oval and in the intersections between the ovals we can see the number of genes/GO terms that are found as significant in each of the cases.

The other most important representation is the scatter plot, which uses Cartesian coordinates to display values for two variables. In this case, the two variables that are represented are the p-values calculated in two different conditions. To these scatter plots some features can be added, such as a dot density area, where the background of the plot appears the more colored when the more amount of dots are in the area. Another addition can be the change of the color an shape of the dots depending on other features of the data apart from the p-value.

3.4 Platform

3.4.1 Hardware

All the analyses were performed on an ASUS R510VX-DM010D with a 2.6 GHz Intel® Core™ i7-6700HQ CPU, 8GB 2133 MHz DDR4 memory and a NVIDIA GeForce GTX 950M/PCIe/SSE2 graphics card.

3.4.2 Software

Regarding the software, all the work was performed in an Ubuntu 16.04 LTS OS. The version of R utilized was 3.4.3 (2017-11-30) and all the scripts and graphs were generated using Rstudio, version 1.1.383.

Results and discussion

4.1 Comparison of normalization methods

The four different normalization methods: Upper Quartile (uppQ), Trimmed Mean of M-values (TMM), Relative Log Expression (RLE) and 'None' were applied to calculate the scaling factors. Then, the DE analysis was performed using `limma` and `edgeR`. Afterwards, the three algorithms: elim, parent-child and classic were used to calculate the significance of the GO terms associated with the DE genes.

4.1.1 Comparison of the DE genes

Depending on the normalization method utilized, different genes were found as DE. The level of agreement is shown in the Venn diagram shown in the FIGURE 4.1.

Note that for this data set, the extremely high number of DE genes are expected, due to the fact that samples are very different. In typical biological experiment the significantly lower number of DE genes is expected.

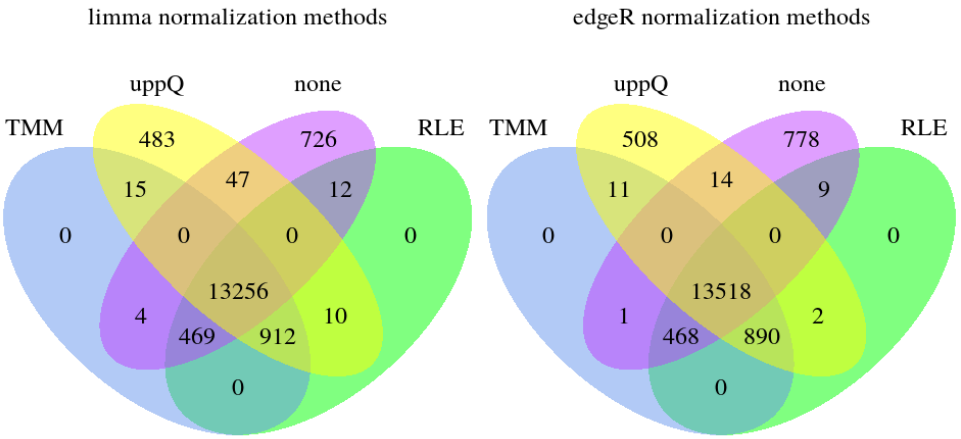

FIGURE 4.1: Venn diagram representing the different DE genes found by `limma` or `edgeR` depending on the normalization method used. The overlapping areas represent the number of genes that are found as DE when utilizing the normalization methods that overlay that area.

The first thing that drives the attention, is that both of the Venn diagrams are very similar. This was expected due to the fact that both packages, `limma` and `edgeR`, are performing the DE analysis using the same data. This may suggest that, probably, similar conclusions can be drawn regardless of which package is used to analyze the data. Later, a whole section will be dedicated to the comparison among tools.

We can observe in the FIGURE 4.1 that all the four methods agree in calling more than 13000 genes as DE, making a high percentage of agreement. The rest of categories, such as genes only found as DE by one method or the genes found as DE pair-wisely are way lower than this.

It is also remarkable that when RLE and TMM normalization methods are compared, their agreement is huge. More than 14500 genes are found as DE in both of the cases (`limma` and `edgeR` approaches) and less than 50 genes are found uniquely by one method or the other. This similarity was expected, both of the methods make the assumption that most of the genes do not change between conditions. It could seem at first sight that this assumption have been violated, since more than 14500 out of a total of 22427 genes have been found posteriorly as DE. However, the agreement level between these two methods suggest that they still work even with this high percentage of DE genes. Both methods make the same assumption but, they work differently, if it had been violated the results would have been much more disparate.

Both the upperquartile and 'None' methods are the most dissimilar compared to the others. This makes perfect sense, since, upperquartile makes different assumptions to calculate the scaling factors and 'None' directly does not apply any normalization to the samples.

4.1.2 Comparison of GO terms significance

When performing a DE analysis, the objective is to extract information about what is really different between conditions, biologically speaking. In order to do so, it is necessary to know which biological process are the DE genes related with. Here is where the GO terms are used, relating each gene with its biological implication.

The Venn diagram shows how the normalization method affect the results of the different algorithms. Also, in this subsection, the p-values that were estimated in each approach, are compared utilizing different plots. These p-values show how probably a term is significantly overrepresented in the pull of DE genes.

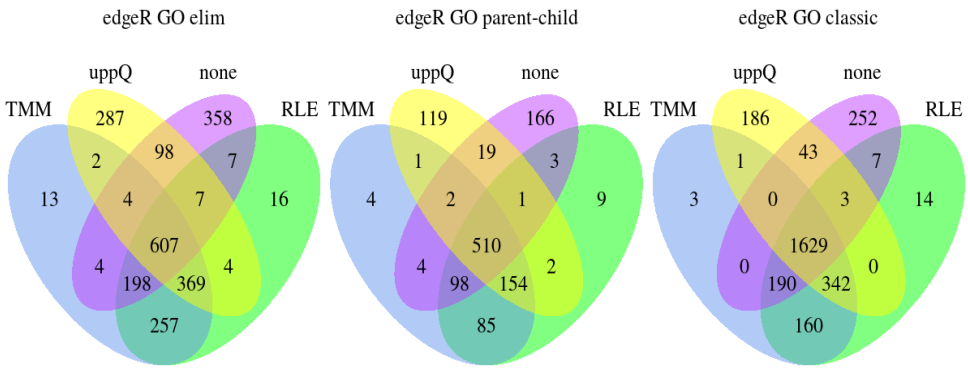

FIGURE 4.2: Venn diagram representing the number of significant GO terms found when using each one of the four normalization methods. Three different diagrams, corresponding with different GO term analysis algorithms, are represented. The package utilized to calculate the DE genes in the previous step was `edgeR`.

The same diagrams were made using the data from `limma` (they can be found in the supplementary data). Both `limma` and `edgeR` approaches showed very similar results, so the same conclusions can be obtained for the two cases.

The first thing that has to be remarked in the FIGURE 4.2 is that we observe a considerably bigger number of GO terms found as significant when using the classic algorithm. This makes sense, since classic looks globally while elim and parent-child takes into account also local relations between terms in order to focus on more specific ones.

Regarding the level of agreement among normalization methods, we observed, that in the case of classic, the percentage of GO terms found as significant, is higher than in the other two methods. It is also remarkable that the elim algorithm makes more influence than the normalization method used. This is again what was expected, the classic method works in a more general way, classifying term as significant easier. In the case of elim, the more specific terms have a bigger influence than the general ones, making small differences in the gene list more significant in the final results.

Comparing the different normalization methods within each case, a similar distribution can be observed. In all of the three cases the RLE and TMM methods are the most similar between them, due to the fact that both make similar assumptions. Upperquartile and 'None' methods show bigger differences for the same reason that was explained at the end of the subsection 4.1.1.

Dependence on the normalization method

All these observations can be supported by a scatter plot with the p-values of two different methods on each one of the axes. This, gives an overview about the agreement in the estimation of the statistical significance between two methods. If a density layer is added, it helps to understand where the methods agree more and where the differences are bigger.

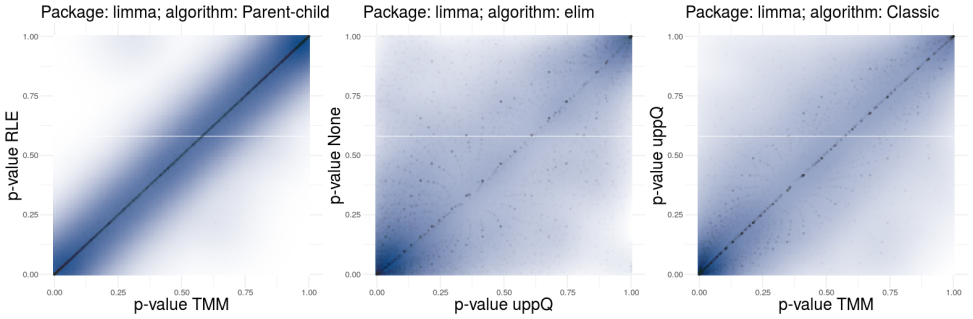

FIGURE 4.3: Dot density scatter plot with the p-values of the GO terms calculated after using different normalization methods. The darker is the blue, the more dots are in the area. These plots were made using the results from `limma` whit the different algorithms.

The plots shown in the FIGURE 4.3 are a sample of all the plots prepared, that comprehend all the possible six combinations of pairs of normalization method for each of the three algorithms for both of the packages, making a total of thirty-six plots. Plots from which three representative have been selected and showed here. All the rest of the plots show a very similar distribution, being the pair of normalization methods compared the variable that makes the difference.

The FIGURE 4.3 corroborates conclusions drawn at the basis of figure FIGURE 4.2, that different methods have different level of agreement. TMM and RLE methods

make similar assumptions, leading to similar results in the posterior functional enrichment analysis. While upperquartile and 'None' methods are more different between them as well as compared to the other methods. It is remarkable that the agreement is higher when the p-values are extreme, both in the higher end (closer to 1) and in the lower end (closer to 0). That could indicate that high amount or lack of information leads to proper analysis irrespectively from the performed method, while cases with lower amount of information are more sensitive for processing approach.

Dependence on the GO analysis algorithm

As explained in the point 1.3.7, different algorithms can be applied to calculate the statistical significance of each GO term. The final p-value of each term is different and it depends on what algorithm is applied.

Some plots have been made, comparing the p-values obtained using the three different algorithms in all the four cases (using the four different normalization methods). Here some of them are shown.

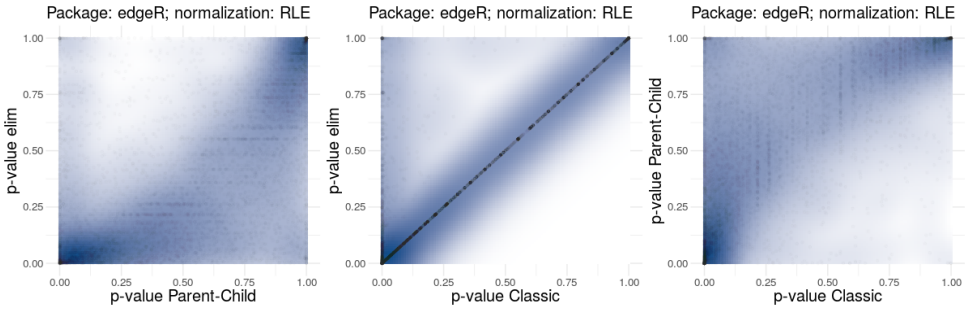

FIGURE 4.4: Dot density scatter plot with the p-values of the GO terms calculated using different methods. The darker is the blue, the more dots are in the area. These plots were made using the results from `edgeR` whit the RLE normalization.

The FIGURE 4.4 illustrates the influence of the chosen algorithm on the final result. Only elim and classic algorithms keep some correlation, which will be analyzed posteriorly. Comparing elim and classic with Parent-child only a low correlation is observed. The level of agreement is higher for both extremes, when p-value is either close to 1 or to 0.

The plots from figures 4.4 and 4.5 were made for all the normalization methods and both for `edgeR` and `limma`, there were no significant differences among them, so they lead to the same conclusions. The rest of the plots can be found in the supplementary data.

The first remark that can be done observing the plots from the FIGURE 4.5 is that the choice of the algorithm is not trivial. Terms that have rather high p-values with one algorithm can be called as significant by another one.

Observing the left plot, which compares p-values obtained by classic and Parent-child methods we can affirm several things. The classic algorithm found more GO terms as significant, this is consistent with what was observed in the FIGURE 4.2. Parent-child algorithm's main objective is to detect and remove overrepresentation, that is why we observe less terms as significant. The big white dots we observe in the lower left corner suggest that most general terms are found as significant by

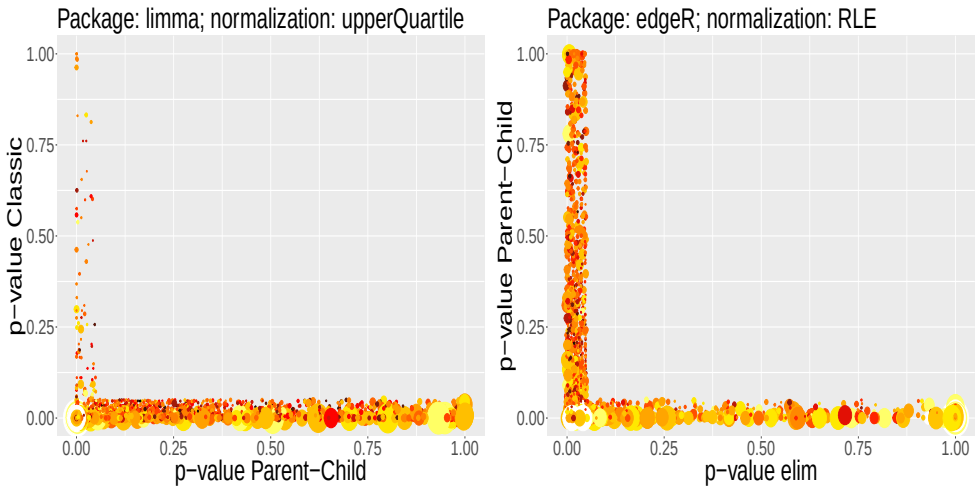

FIGURE 4.5: On the left panel, p-values scatter plot for the parent-child (x axis) and classic (y axis) methods. On the right panel, p-values scatter plot for the elim (x axis) and Parent-child (y axis) methods. Only significant terms in one or the other method have been represented. The size of the dot is proportional to the number of annotated genes for the respective GO term and its coloring represents the depth in the GO hierarchy, with the dark red points being more specific terms than the light yellow ones.

both methods, however when the hierarchy goes deeper we start to see discrepancies between the methods, which was expected as Parent-child removes the overrepresented terms.

Regarding the plot on the right, which compares the p-values obtained applying Parent-child and elim algorithms, some observations can be done. The first thing that drives the attention is that elim method find as significant deeper terms (represented in dark red) that have less annotated terms (smaller circles), which is an exact opposite of what parent- child algorithm reported-it finds less deep and more represented terms. This makes sense if we take into account how these two algorithms work. Elim's objective is to eliminate the terms that are not significant by themselves (those terms that are significant only because they have children terms which are significant). This means that, if a child term is significant it is removed from the analysis of its parents, enriching the analysis in specific terms in detriment of the general ones. In the case of Parent-child the algorithm tend to eliminate the overrepresentation of terms in a less conservative way, keeping more general terms, and it gives more weight to the number of genes annotated to one term, making more general terms easier to appear as significant.

A special comparison is the classic vs elim, there is an expected trend in which most of the p-values calculated by the two methods will be the same. This trend can be observed in the FIGURE 1 of the TopGO viggnete (bioconductor.org, 2018b). Even with the data used in this work, which has a huge amount of DE genes, and, thus a huge amount of significant GO terms this trend can be observed.

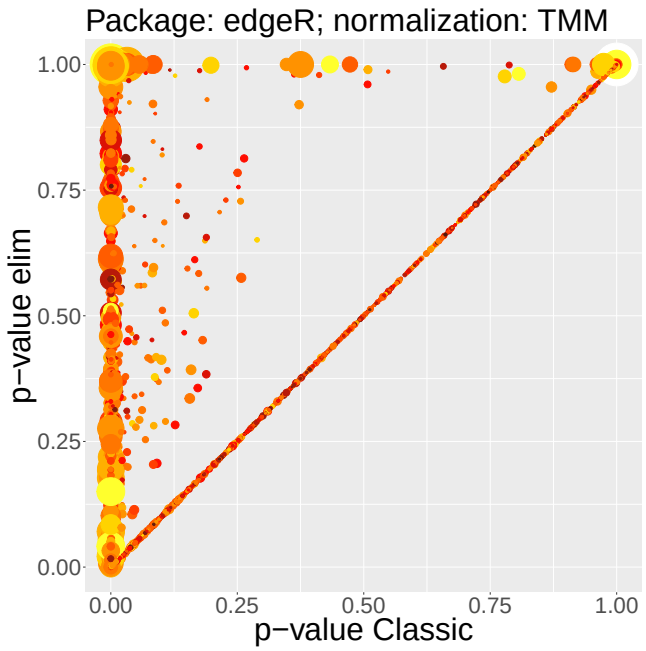

FIGURE 4.6: p-values scatter plot for the classic (x axis) and elim (y axis) methods. The size of the dot is proportional to the number of annotated genes for the respective GO term and its coloring represents the depth in the GO hierarchy, with the dark red points being more specific terms than the light yellow ones.

As it was expected, a general trend can be observed in the FIGURE 4.6, comprising especially the specific terms. At the same time we can confirm that there is no term which p-value is bigger for classic than for elim. The disagreement can be observed for both the terms that are deep and high in the hierarchy, however it draws attention that most of the discrepancies are related to the more general terms, which often contain more genes associated with them. This was expected as elim basically removes the general terms that do not have enough significance by themselves.

4.2 Packages comparison

An approach similar to the one realized with the normalization methods has been done to compare the three different DE calling packages `limma`, `edgeR` and `DESeq2`. Firstly, the DE genes have been compared. In a posterior analysis the significance of the GO terms associated with these genes is compared among packages utilizing the three algorithms described in the subsection 1.3.7.

4.2.1 Comparison of the DE genes

Depending on the package utilized to perform the DE analysis, different results are expected. Each one of them has its own assumptions, applies its normalization method and utilizes a different approach to calculate the level of significance (p-value) of each gene being DE.

To give an overview about which method is more ore less sensitive to the fold change between conditions and to the average counts of a gene an MA plot can be used. In

this plot the x axes represents the average expression of a gene while the y axes represents the logarithmic fold change between conditions.

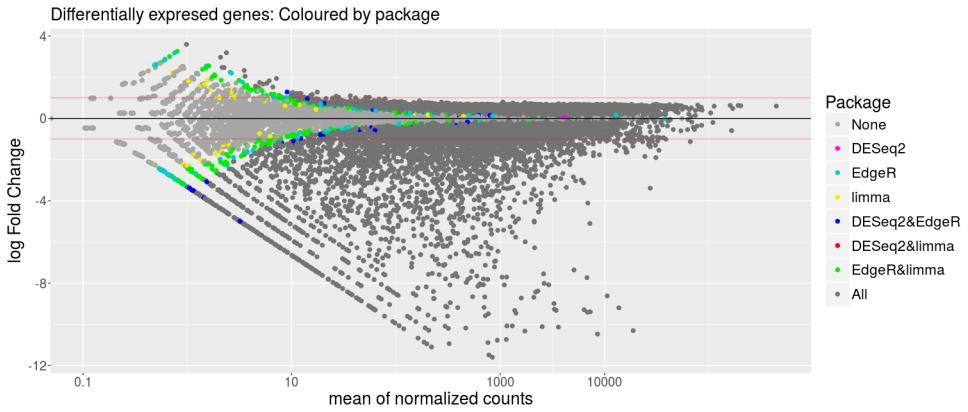

FIGURE 4.7: MA plot representing the logarithmic fold-change in the y axes and the mean of normalized counts in the x axes. The coloring of the different points of the scatter-plot depend on the package that detects them as DE.

The FIGURE 4.7 illustrates the common behavior that the packages have, showing how the two main factors affect all of them in general. The fold-change is one of them and the bigger it is, the more prone are the packages to detect the gene as DE. This makes sense because it is an indicator of the gene expression change between conditions. The other one is the mean of normalized counts, which indicates how strong is the average expression of the gene in all the samples. Regarding this last factor, the higher it is, the less big the fold-change is needed to be in order to detect a gene as DE.

Also in the FIGURE 4.7 we can analyze the differences among methods. Most of the discrepancies are located in the border line that separates the genes found as DE by all the methods (dark gray) from the ones that are not detected as DE by any of them (light gray). This is what would be expected, since all the packages are used for the same purposes results must be similar, however, the fact that different packages are being analyzed and they use different algorithms makes to expect some little differences. The main difference among methods is that the `limma` (yellow) and the `edgeR` (cyan) packages are less sensitive to the mean of counts than `DESeq2`. This can be inferred from the cyan, yellow and green (combination of `limma` and `edgeR`) dots that can be observed when the average counts are less than 10.

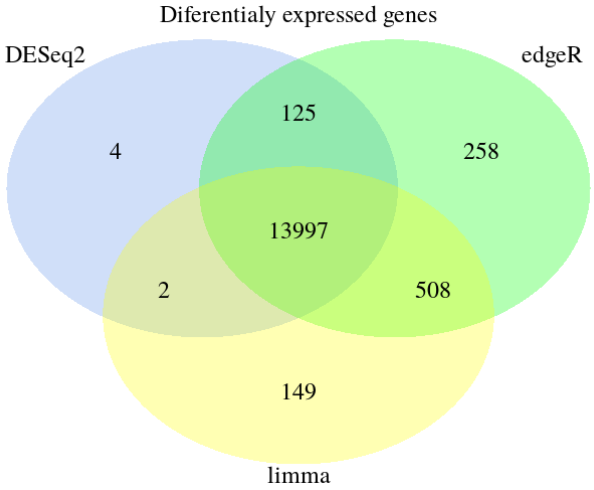

FIGURE 4.8: Venn diagram representing the different DE genes found depending on the package used.

While the FIGURE 4.7 allowed to differentiate where the discrepancies among methods where, the Venn diagram of the FIGURE 4.8 permits to quantify this disagreement. The first thing that drives the attention is that the general agreement is considerably great, it is even bigger than when normalization methods were compared (FIGURE 4.1). Another feature worth to highlight is that `DESeq2` is more conservative (as it was observed in the FIGURE 4.7), finding less genes as DE than both `limma` and `edgeR`; it also shows a higher agreement with `edgeR` than with `limma`.

4.2.2 Comparison of GO terms significance

The agreement at the gene level was high. This leads to conclusion that agreement at the functional enrichment level is expected. To check if this is the case, some Venn diagrams and scatter plots have been made.

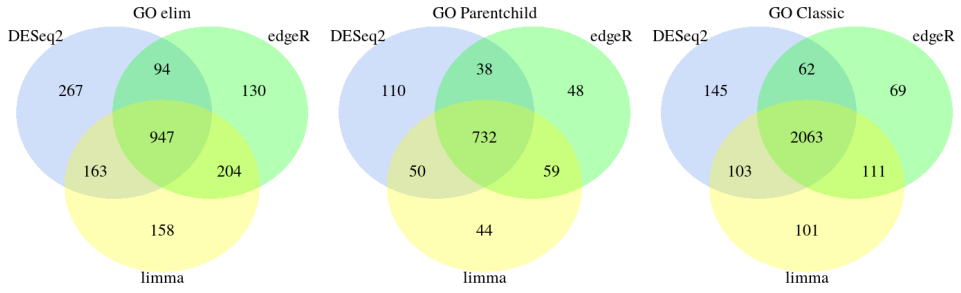

FIGURE 4.9: Venn diagram representing the number of significant GO terms found when using each one of the three different packages. Three different diagrams, corresponding with different GO term analysis algorithms, are represented.

What is visible on the FIGURE 4.9 is that `limma` and `edgeR` show a higher level of agreement with one another than they do with `DESeq2`. This was expected, because the same happened at the DE gene level. The level of agreement in general is lower than it was when different normalization methods were compared (FIGURE 4.2). This is caused by the fact that there are more disagreements among methods detecting DE genes than among normalization methods.

Regarding the comparison among algorithms, a similar behavior with the one visible on the FIGURE 4.2 is observed. Classic algorithm finds as significant more GO terms than elim and parent-child do. The differences are also bigger when the elim algorithm is applied than when parent-child is used; the classic algorithm also shows here the lowest level of discrepancy. The reasoning of why this is happening is parallel to the one in the subsection 4.1.2.

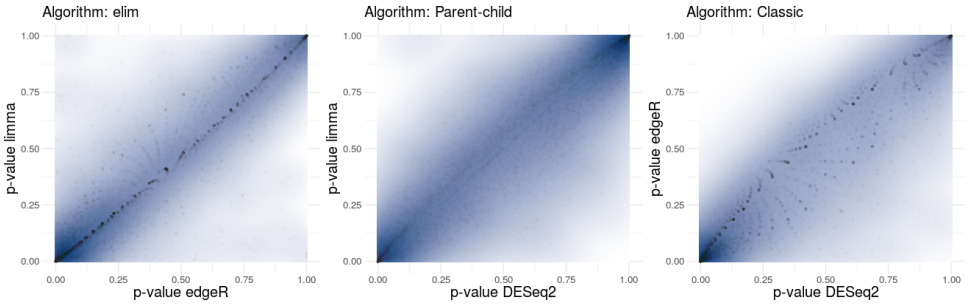

FIGURE 4.10: Dot density scatter plot with the p-values of the GO terms calculated using different packages. The darker is the blue, the more dots are in the area.

The scatter plot of the FIGURE 4.10 helps to understand where the different cases agree more. The same plots were made utilizing different combinations of packages and algorithms (they can be found in the supplementary data). The three representative cases show a strong correlation between methods. The strongest correlation is visible between `edgeR` and `limma`, as it was expected. The most dense areas are both corners of the diagonal, which means that most of the terms have either very low or very high p-values and, since these genes are in the diagonal, a correlation in these cases can be observed.

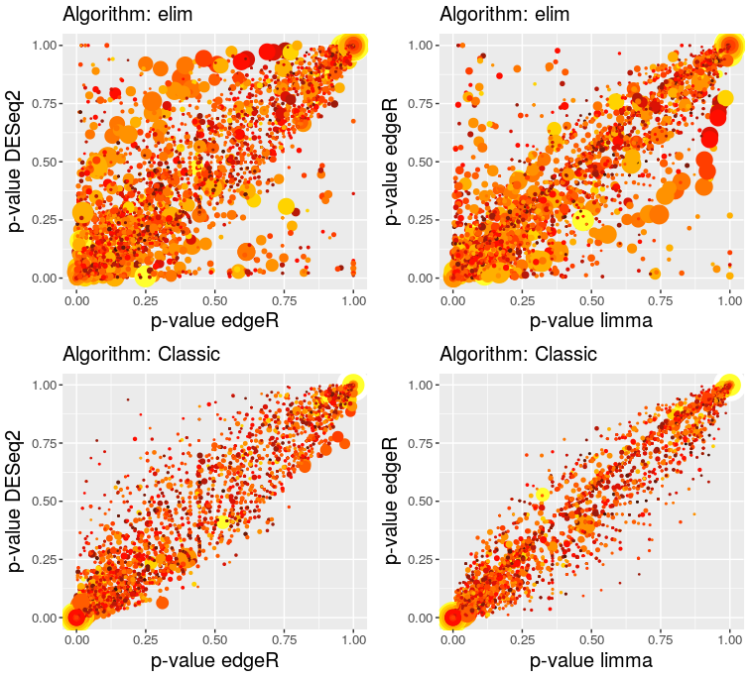

FIGURE 4.11: p-values scatter plot for GO terms calculated using the data from different packages. The size and the colour of the dots represent the hierarchy and the number of annotated genes (see FIGURE 4.5).

The correlation among packages is not perfect and some discrepancies appeared. To observe where in the hierarchy this discordance appears a plot similar to figures 4.5 and 4.6 can be plotted.

In the FIGURE 4.11 two comparisons have been made. `DESeq2` with `edgeR` and `edgeR` with `limma`. Both of the comparisons were made for elim and classic algorithms. As it was expected, the classic algorithm shows more correlation between packages also the correlation is stronger between `limma` and `edgeR`, which agrees with the results of previous exploratory analyses.

It should be noted that the difference between elim and classic algorithm is quite important. In the case of the classic, in both comparisons we can observe a rather good correlation, however, in the case of elim, numerous terms have very different p-values depending on the package used to call the DE genes. This is more remarkable in the case of the comparison between `limma` and `edgeR`; they only differ in few genes and yet, these one or two extra genes in one term can result in huge differences. Small differences in the gene list used as an input for the GO term enrichment analysis can make the difference in the related GO terms if elim method is applied.

Conclusions

To summarize and finish up this exertion, a few observations regarding the whole of the project are condensed in this section.

- The differences at the gene level are lower than at the functional enrichment level. This might come from unusual samples which were compared as these were artificial samples. In the case of the typical biological samples, we expect to have some "main driving biological" signal. Thus, in a typical case, the agreement on functional level would be expected to be higher than on the gene level.
- The influence of the normalization method is not very high if we take into account the most used RLE and TMM.
- The influence of the algorithm used in the GO terms enrichment is considerable, being classic much more general; parent-child does not assume that if a gene is assigned to a term it is also assigned to all parents of this term; and elim removes the children in any parents analysis
- The influence of the package chosen is crucial, mostly for genes with low average counts. This bias is not so important if we take into account that most of the approaches include a filtering of the low expressed genes. The packages also differ in the differential expression calling when fold-change is low and average counts is high. This is also avoided in common analyses by removing genes with low fold-change from the analysis.
- To summarize, it is important to understand how the bioinformatic tools work in order to better choose the ones that fit your objectives the best. Depending on what you want to focus in, you have to be very selective with your approach.

Bibliography

ADAMS, M.D., M. DUBNICK, A.R. KERLAVAGE, R. MORENO, J.M. KELLEY, T.R. UTTERBACK, J.W. NAGLE, C. FIELDS, and J.C. VENTER (1992). "Sequence identification of 2,375 human brain genes". In: *Nature* 355.6361, pp. 632–634. DOI: 10.1038/355632a0.

BIOCONDUCTOR.ORG (2018a). *Bioconductor; [Accessed: 12th May 2018]*. URL: http://www.bioconductor.org/.

BIOCONDUCTOR.ORG (2018b). *TopGO viggnete; [Accessed: 29th April 2018]*. URL: https://bioconductor.org/packages/release/bioc/vignettes/topGO/inst/doc/topGO.pdf.

BRAY, N.L., H. PIMENTEL, P. MELSTED, and L. PACHTER (2016). "Near-optimal probabilistic RNA-seq quantification". In: *Nature Biotechnology* 34.5, 525–527. DOI: 10.1038/nbt.3519.

CARNEIRO, M.O., C. RUSS, M.G. ROSS, S.B. GABRIEL, C. NUSBAUM, and M.A. DEPRISTO (2012). "Pacific biosciences sequencing technology for genotyping and variation discovery in human data". In: *BMC Genomics* 13. DOI: 10.1186/1471-2164-13-375.

CHIDGEAVADZE, Z.G., R.S. BEABEALASHVILLI, A.M. ATRAZHEV, V. AZHAYEV, and A.A KRAYEVSKY (1984). "2', 3'-dideoxy-3' Aminonucleoside 5'-triphosphates are the Terminators of DNA-synthesis Catalyzed by DNA-polymerases". In: *Nucleic Acids Research* 12.3, 1671–1686. DOI: 10.1093/nar/12.3.1671.

DOBIN, A., C.A. DAVIS, F. SCHLESINGER, J. DRENKOW, C. ZALESKI, S. JHA, P. BATUT, M. CHAISSON, and T.R. GINGERAS (2013). "STAR: ultrafast universal RNA-seq aligner". In: *Bioinformatics* 29.1, 15–21. DOI: 10.1093/bioinformatics/bts635.

EID, J. et al. (2009). "Real-Time DNA Sequencing from Single Polymerase Molecules". In: *Science* 323.5910, 133–138. DOI: 10.1126/science.1162986.

FISHER, R.A (1922). "On the interpretation of χ^2 from contingency tables, and the calculation of P". In: *Journal of the Royal Statistical Society* 85, 87–94. DOI: 10.2307/2340521.

FU, X., N. FU, S. GUO, Z. YAN, Y. XU, H. HU, C. MENZEL, W. CHEN, Y. LI, R. ZENG, and P. KHAITOVICH (2009). "Estimating accuracy of RNA-Seq and microarrays with proteomics". In: *BMC Genomics* 10. DOI: 10.1186/1471-2164-10-161.

GINZINGER, D.G. (2002). "Gene quantification using real-time quantitative PCR: An emerging technology hits the mainstream". In: *Experimental Hematology* 30.6, 503–512. DOI: 10.1016/S0301-472X(02)00806-8.

GOODWIN, S., J.D. MCPHERSON, and W.R. MCCOMBIE (2016). "Coming of age: ten years of next-generation sequencing technologies". In: *Nature Reviews Genetics* 17.6, 333–351. DOI: 10.1038/nrg.2016.49.

GOODWIN, S., J. GURTOWSKI, S. ETHE-SAYERS, P.a DESHPANDE, M.C. SCHATZ, and W.R. MCCOMBIE (2015). "Oxford Nanopore sequencing, hybrid error correction, and de novo assembly of a eukaryotic genome". In: *Genome Research* 25.11, 1750–1756. DOI: 10.1101/gr.191395.115.

HAROLDPIMENTEL.WORDPRESS.COM (2018). *What the FPKM? A review of RNA-Seq expression units; [Accessed: 25th May 2018]*. URL: https://haroldpimentel.wordpress.com/2014/05/08/what-the-fpkm-a-review-rna-seq-expression-units/.

HEATHER, J.M. and B. CHAIN (2016). "The sequence of sequencers: The history of sequencing DNA". In: *Genomics* 107.1, 1–8. DOI: 10.1016/j.ygeno.2015.11.003.

HRDLICKOVA, R., M. TOLOUE, and B. TIAN (2016). "RNA-Seq methods for transcriptome analysis". In: *Wiley Interdisciplinary Reviews: RNA* 8.1, e1364. DOI: 10.1002/wrna.1364.

HUANG, Y., Y. HU, C.D. JONES, J.N. MACLEOD, D.Y. CHIANG, Y. LIU, J.F. PRINS, and J. LIU (2013). "A Robust Method for Transcript Quantification with RNA-Seq Data". In: *Journal of Computational Biology* 20.3, 167–187. DOI: 10.1089/cmb.2012.0230.

HUBBARD, R. and J.S. ARMSTRONG (2006). "Why We Don't Really Know What Statistical Significance Means: Implications for Educators". In: *Journal of Marketing Education* 28, 114–120. DOI: 10.1177/0273475306288399.

KIM, D., B. LANGMEAD, and S.L. SALZBERG (2015). "HISAT: a fast spliced aligner with low memory requirements". In: *Nature Methods* 12.4, 357–U121. DOI: 10.1038/NMETH.3317.

KIM, D., Geo P., C. TRAPNELL, H. PIMENTEL, R. KELLEY, and S.L. SALZBERG (2013). "TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions". In: *Genome Biology* 14.4. DOI: 10.1186/gb-2013-14-4-r36.

KORPELAINEN, E. (2015). "RNA-seq data analysis: a practical approach". In: *Lipoproteins and Cardiovascular Disease: Methods and Protocols*. Ed. by CRC PRESS, TAYLOR & FRANCIS GROUP; BOCA RATON.

LABAJ, P.P. and D.P. KREIL (2016). "Sensitivity, specificity, and reproducibility of RNA-Seq differential expression calls". In: *Biology Direct* 11. DOI: 10.1186/s13062-016-0169-7.

LIAO, Y., G.K. SMYTH, and W. SHI (2013). "The Subread aligner: fast, accurate and scalable read mapping by seed-and-vote". In: *Nucleic Acids Research* 41.10. DOI: 10.1093/nar/gkt214.

LOMAN, N.J., R.V. MISRA, T.J. DALLMAN, C. CONSTANTINIDOU, S.E. GHARBIA, J. WAIN, and M.J. PALLEN (2012). "Performance comparison of benchtop high-throughput sequencing platforms". In: *Nature Biotechnology* 30.5, 434+. DOI: 10.1038/nbt.2198.

LOVE, M.I., W. HUBER, and S. ANDERS (2014). "Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2". In: *Genome Biology* 15.12. DOI: 10.1186/s13059-014-0550-8.

MOONEY, M. et al. (2013). "Comparative RNA-Seq and Microarray Analysis of Gene Expression Changes in B-Cell Lymphomas of Canis familiaris". In: *Plos One* 8.4. DOI: 10.1371/journal.pone.0061088.

NCBI (2018). *Query DataSets for GSE47774; [Accessed: 10th January 2018]*. URL: https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE47774.

PATRO, R., G. DUGGAL, M.I. LOVE, R.A. IRIZARRY, and C. KINGSFORD (2017). "Salmon provides fast and bias-aware quantification of transcript expression". In: *Nature Methods* 14.4, 417+. DOI: 10.1038/nmeth.4197.

PERTEA, M., D. KIM, G.M. PERTEA, J.T. LEEK, and S.L. SALZBERG (2016). "Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown". In: *Nature Protocols* 11.9, 1650–1667. DOI: 10.1038/nprot.2016.095.

R-PROJECT.ORG (2018). *The R Project for Statistical Computing; [Accessed: 12th May 2018]*. URL: https://www.r-project.org/.

ROBINSON, M.D., D.J. MCCARTHY, and G.K. SMYTH (2010). "edgeR: a Bioconductor package for differential expression analysis of digital gene expression data". In: *Bioinformatics* 26.1, 139–140. DOI: 10.1093/bioinformatics/btp616.

ROTHBERG, J.M. et al. (2011). "An integrated semiconductor device enabling non-optical genome sequencing". In: *Nature* 475.7356, 348–352. DOI: 10.1038/nature10242.

SMYTH, G.K. (2004). "Linear models and empirical bayes methods for assessing differential expression in microarray experiments". In: *Stat Appl Genet Mol Biol* 3, 1–26. DOI: 10.2202/1544-6115.1027.

SU, ZQ. et al. (2014). "A comprehensive assessment of RNA-seq accuracy, reproducibility and information content by the Sequencing Quality Control Consortium". In: *Nature Biotechnology* 32.9, 903–914. DOI: 10.1038/nbt.2957.

SU, ZQ., Z. LI, T. CHEN, Q.Z. LI, H. FANG, D. DING, W. GE, B. NING, H. HONG, R.G. PERKINS, W. TONG, and L. SHI (2011). "Comparing Next-Generation Sequencing and Microarray Technologies in a Toxicological Study of the Effects of Aristolochic Acid on Rat Kidneys". In: *Chemical Research in Toxicology* 24.9, 1486–1493. DOI: 10.1021/tx200103b.

SULTAN, M. et al. (2008). "A global view of gene activity and alternative splicing by deep sequencing of the human transcriptome". In: *Science* 321.5891, 956–960. DOI: 10.1126/science.1160342.

VELCULESCU, V.E., L. ZHANG, B. VOGELSTEIN, and K.W. KINZLER (1995). "Serial analysis of gene-expression". In: *Science* 270, 484–487. DOI: 10.1126/science.270.5235.484.

WAGNER, E.M. (2013). "Monitoring Gene Expression: Quantitative Real-Time RT-PCR". In: *Lipoproteins and Cardiovascular Disease: Methods and Protocols*. Ed. by FREEMAN, LA. Vol. 1027. Methods in Molecular Biology. Humana Press INC, 19–45. DOI: 10.1007/978-1-60327-369-5_2.

WU, R.Q., X.F. ZHAO, Z.Y. WANG, M. ZHOU, and Q.M. CHEN (2011). "Novel Molecular Events in Oral Carcinogenesis via Integrative Approaches". In: *Journal of Dental Research* 90.5, 561–572. DOI: 10.1177/0022034510383691.

XU, W. et al. (2011). "Human transcriptome array for high-throughput clinical studies". In: *Proceedings of the National Academy of Sciences of the United States of America* 108.9, 3707–3712. DOI: 10.1073/pnas.1019753108.

Complementary figures

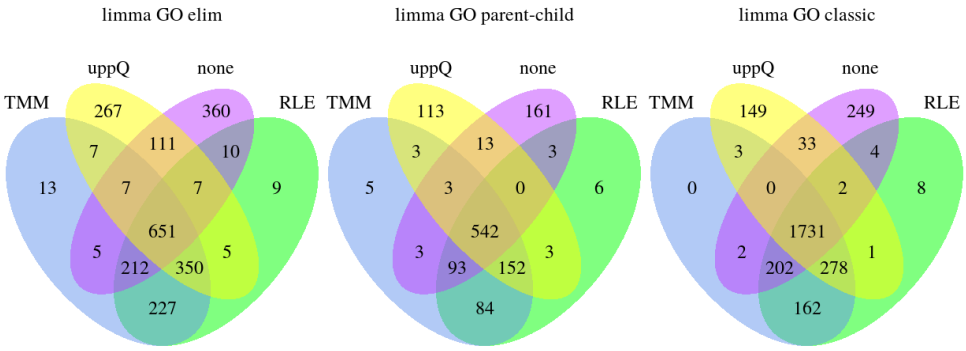

FIGURE A.1: Venn diagram representing the number of significant GO terms found when using each one of the four normalization methods. Three different diagrams, corresponding with different GO term analysis algorithms, are represented. The package utilized to calculate the DE genes in the previous step was `limma`.

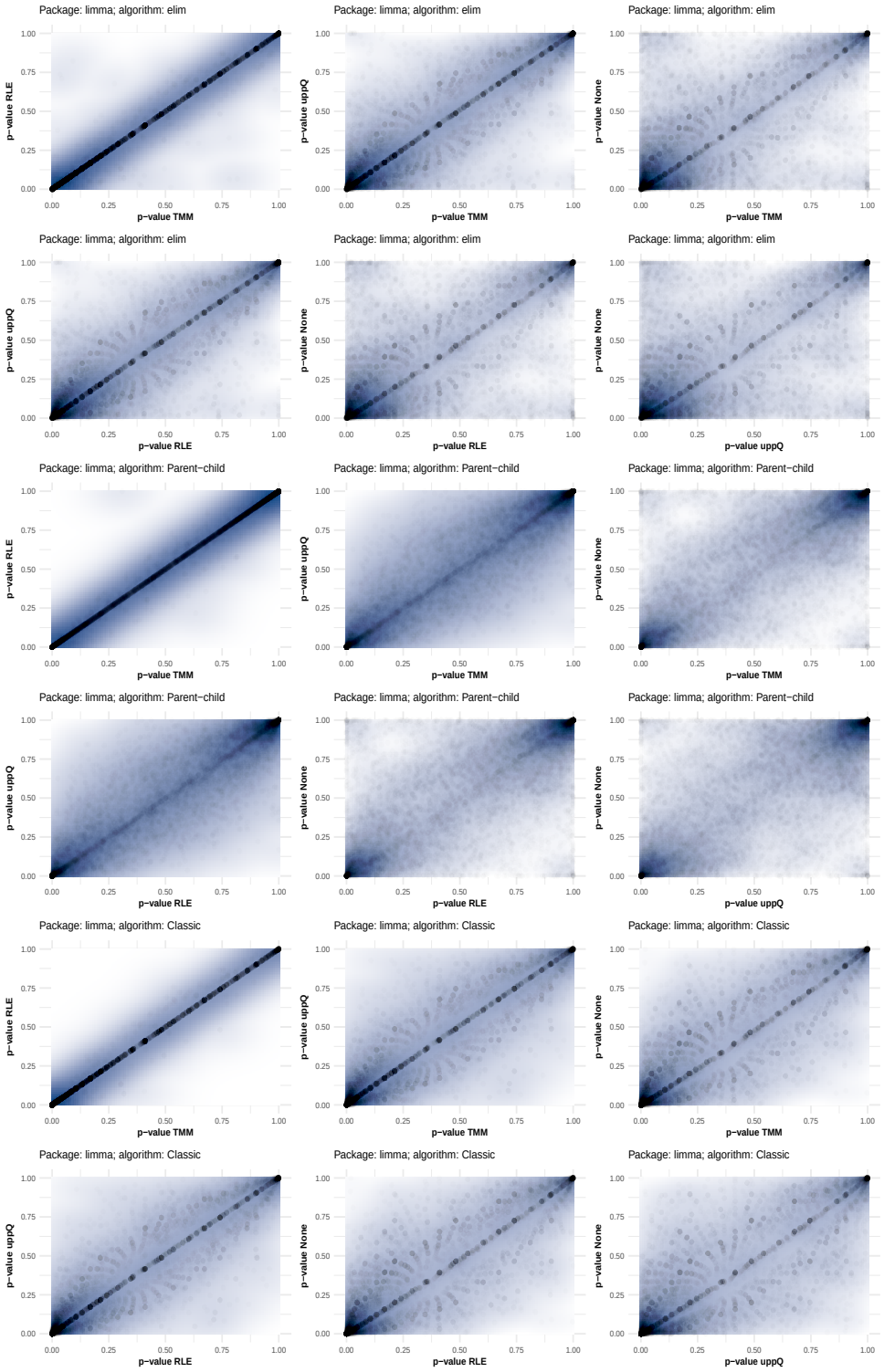

FIGURE A.2: Dot density scatter plot with the p-values of the GO terms calculated after using different normalization methods. The darker is the blue, the more dots are in the area. Faded black dots represent the terms' p-values. These plots were made using the results from `limma` whit the different algorithms.

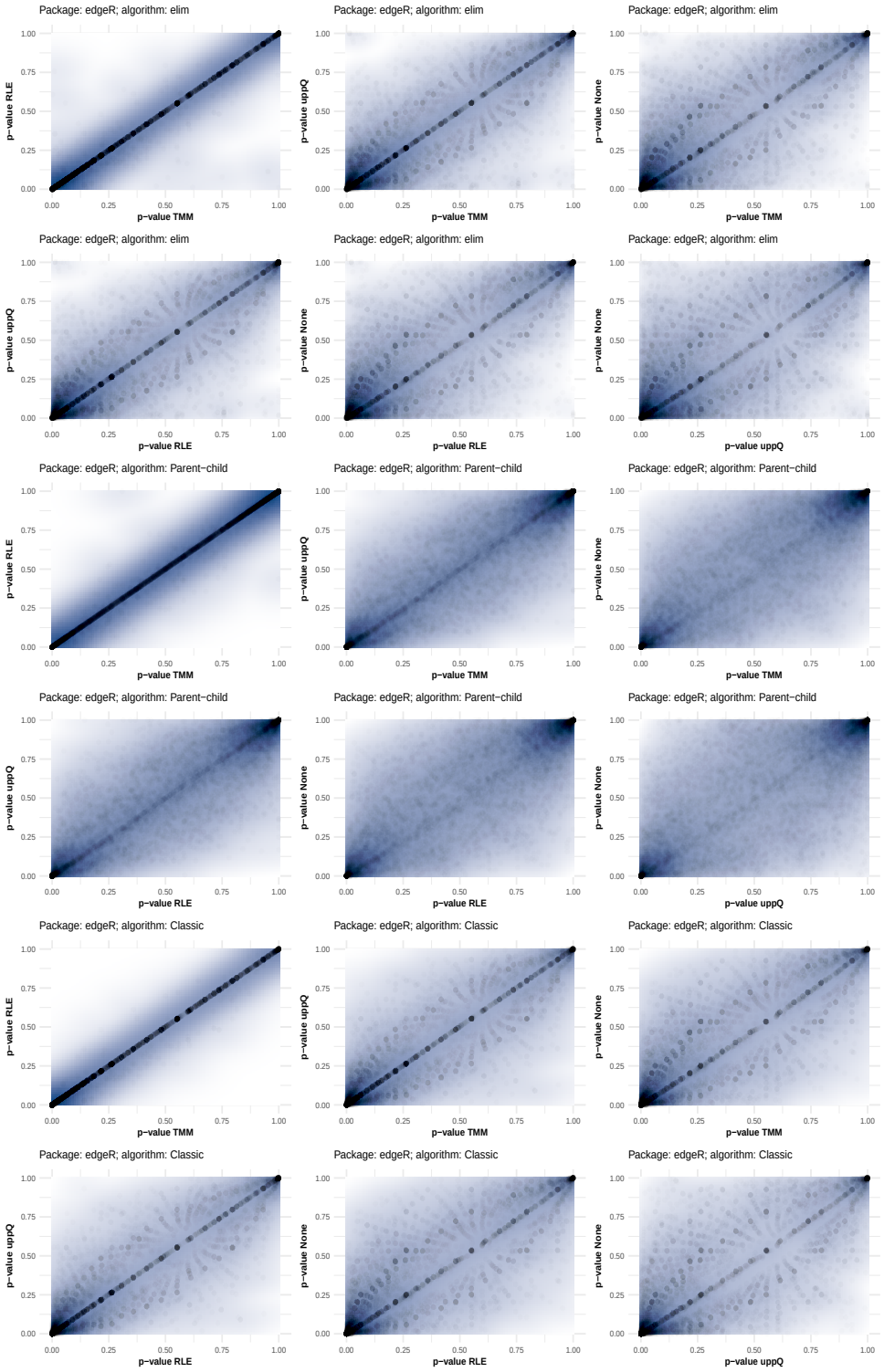

FIGURE A.3: Dot density scatter plot with the p-values of the GO terms calculated after using different normalization methods. The darker is the blue, the more dots are in the area. Faded black dots represent the terms' p-values. These plots were made using the results from `edgeR` whit the different algorithms.

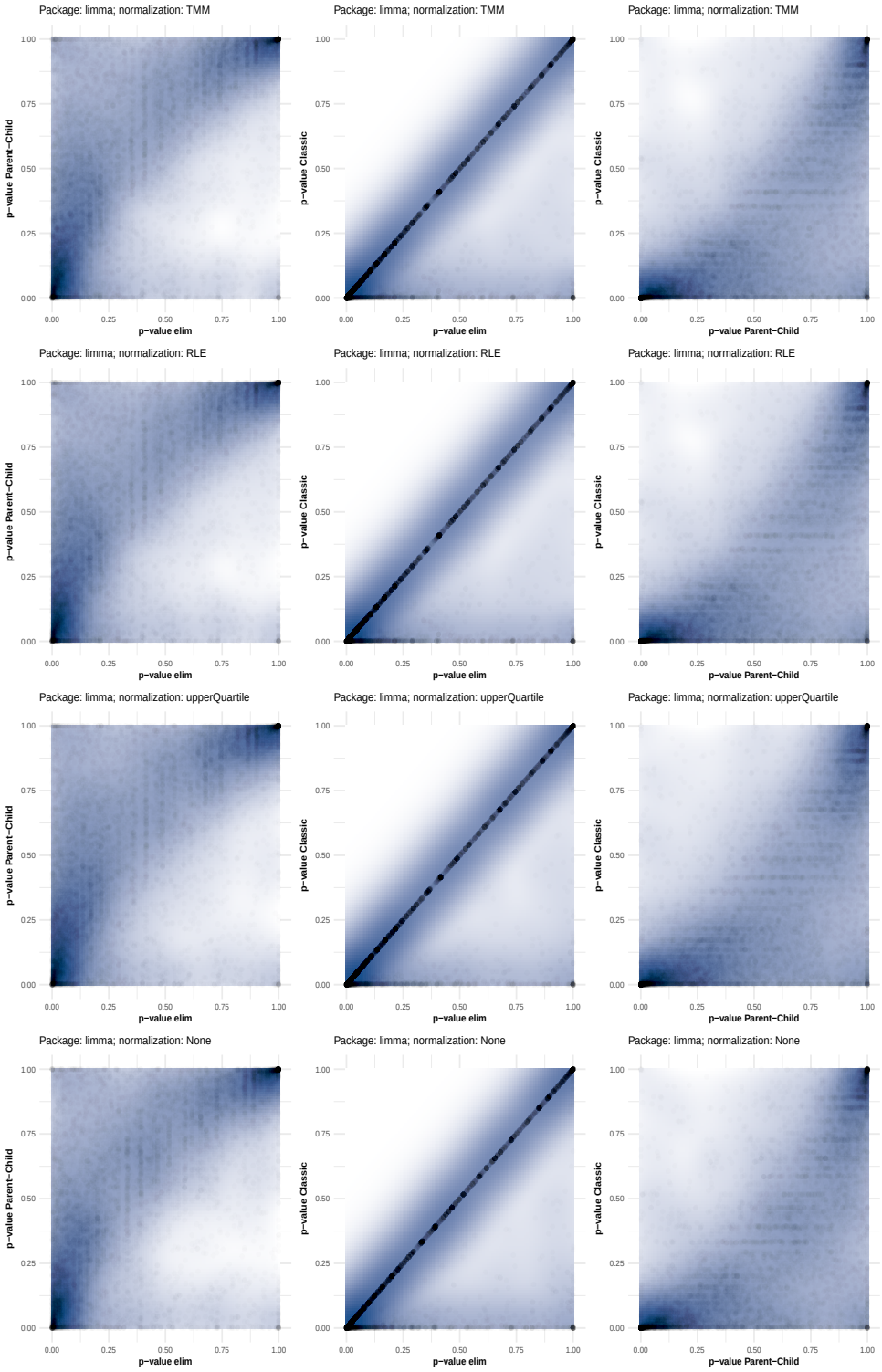

FIGURE A.4: Dot density scatter plot with the p-values of the GO terms calculated using different methods. The darker is the blue, the more dots are in the area. Faded black dots represent the terms' p-values. These plots were made using the results from `limma` whit the the different normalization methods.

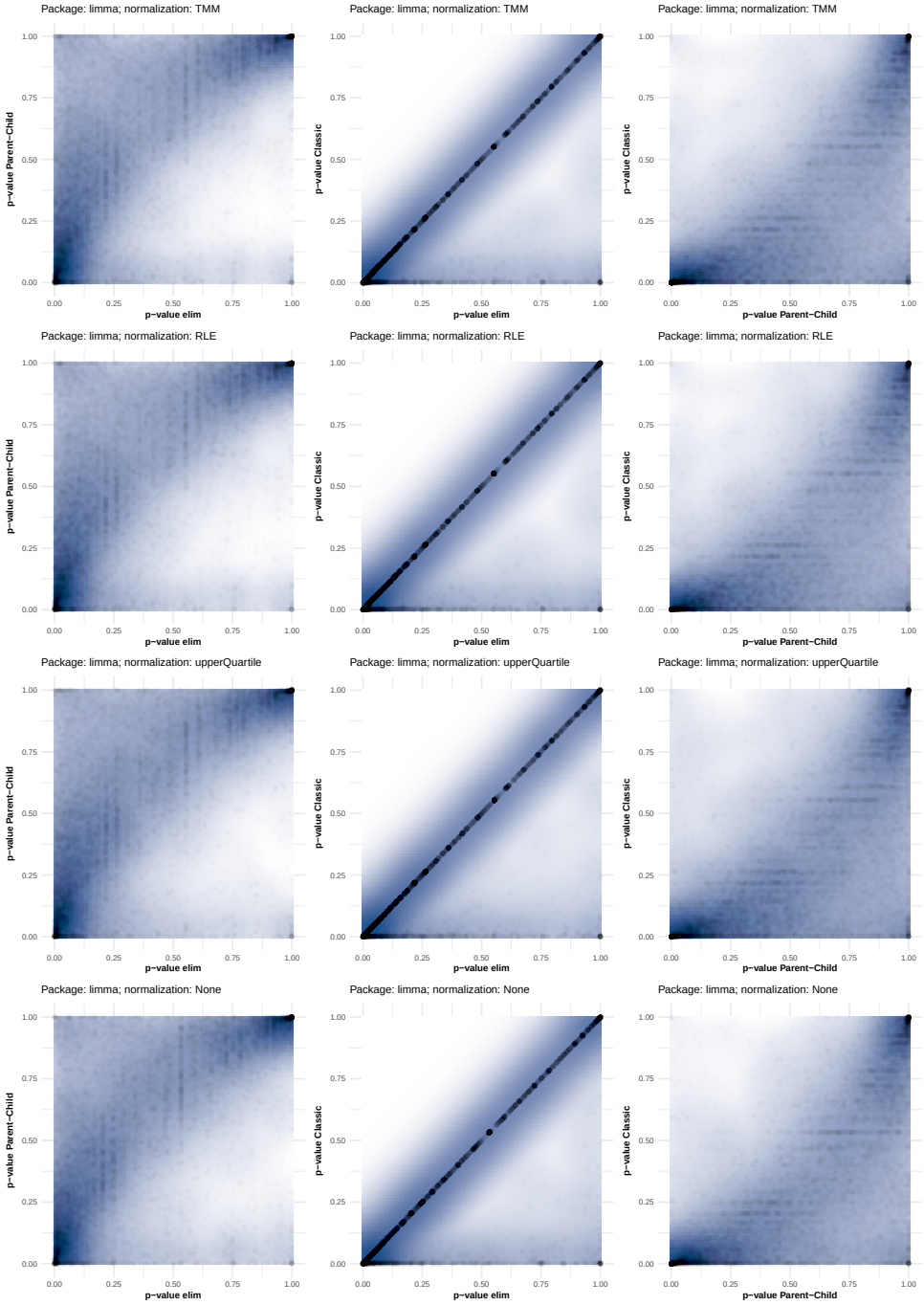

FIGURE A.5: Dot density scatter plot with the p-values of the GO terms calculated using different methods. The darker is the blue, the more dots are in the area. Faded black dots represent the terms' p-values. These plots were made using the results from `edgeR` whit the the different normalization methods.

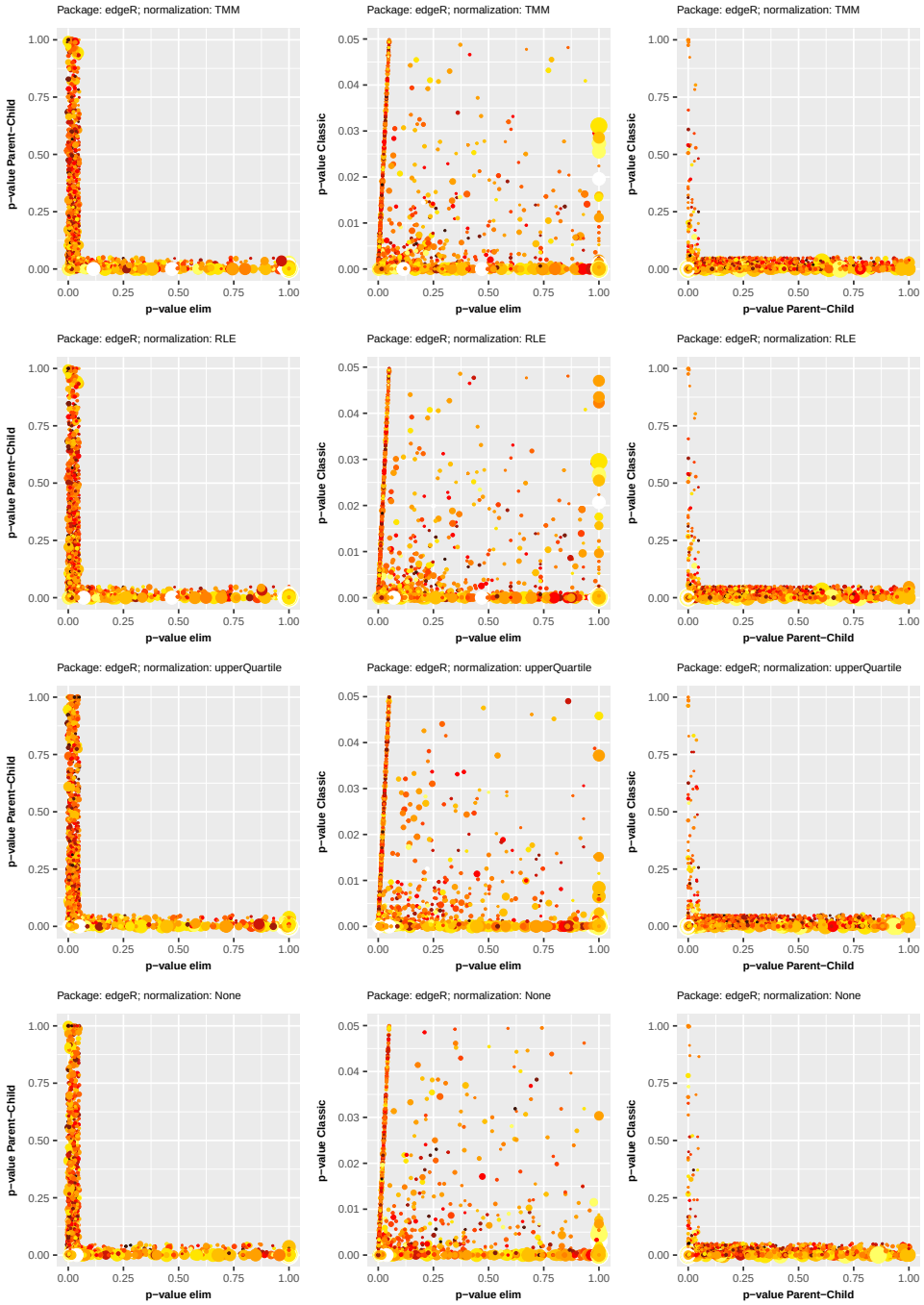

FIGURE A.6: p-value scatter plot algorithm vs algorithm, only significant terms in one or the other method have been represented. The size of the dot is proportional to the number of annotated genes for the respective GO term and its coloring represents the depth in the GO hierarchy, with the dark red points being more specific terms than the light yellow ones. These plots were made using the results from `limma` whit the the different normalization methods.

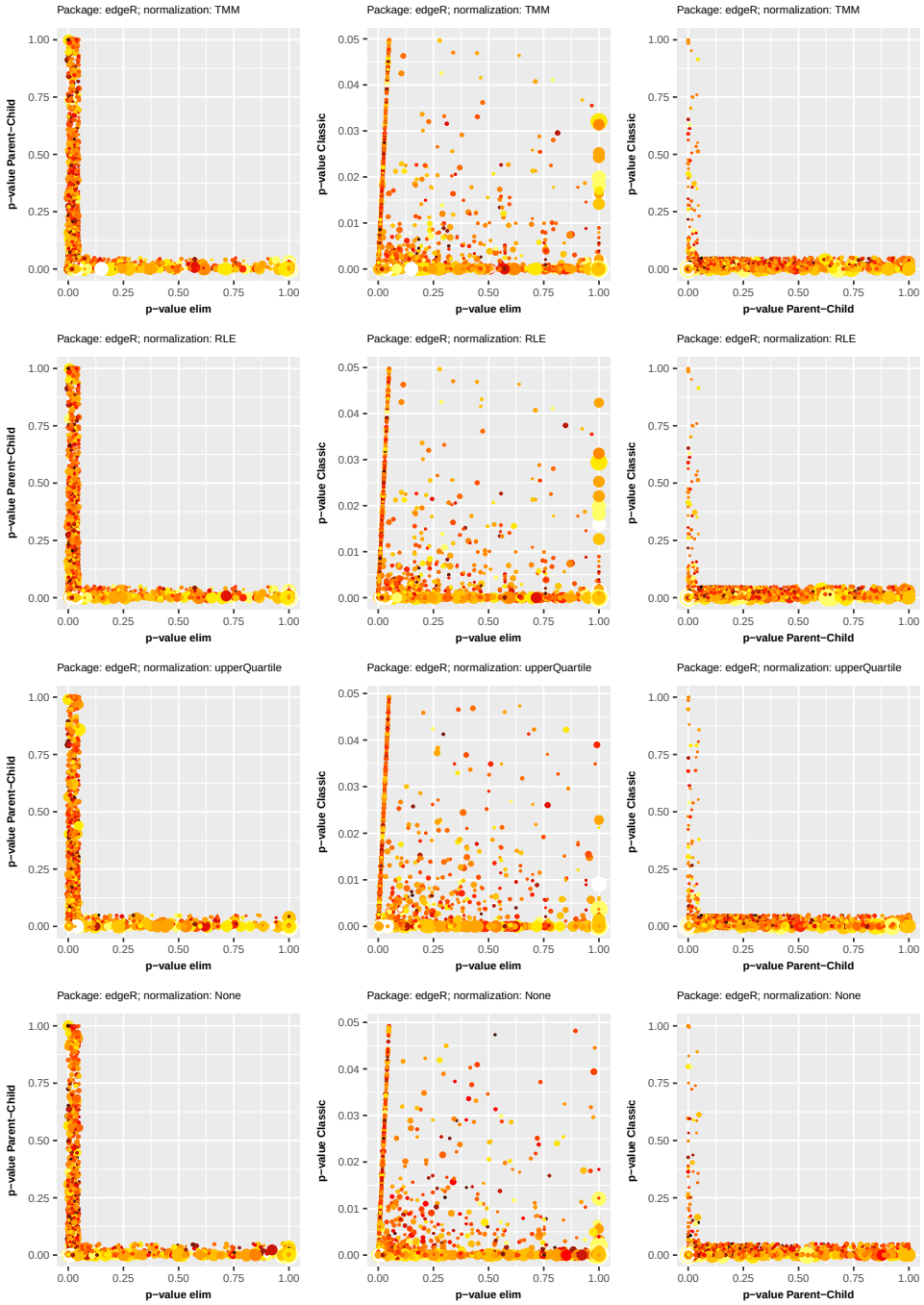

FIGURE A.7: p-value scatter plot algorithm vs algorithm, only significant terms in one or the other method have been represented. The size of the dot is proportional to the number of annotated genes for the respective GO term and its coloring represents the depth in the GO hierarchy, with the dark red points being more specific terms than the light yellow ones. These plots were made using the results from `edgeR` whit the the different normalization methods.

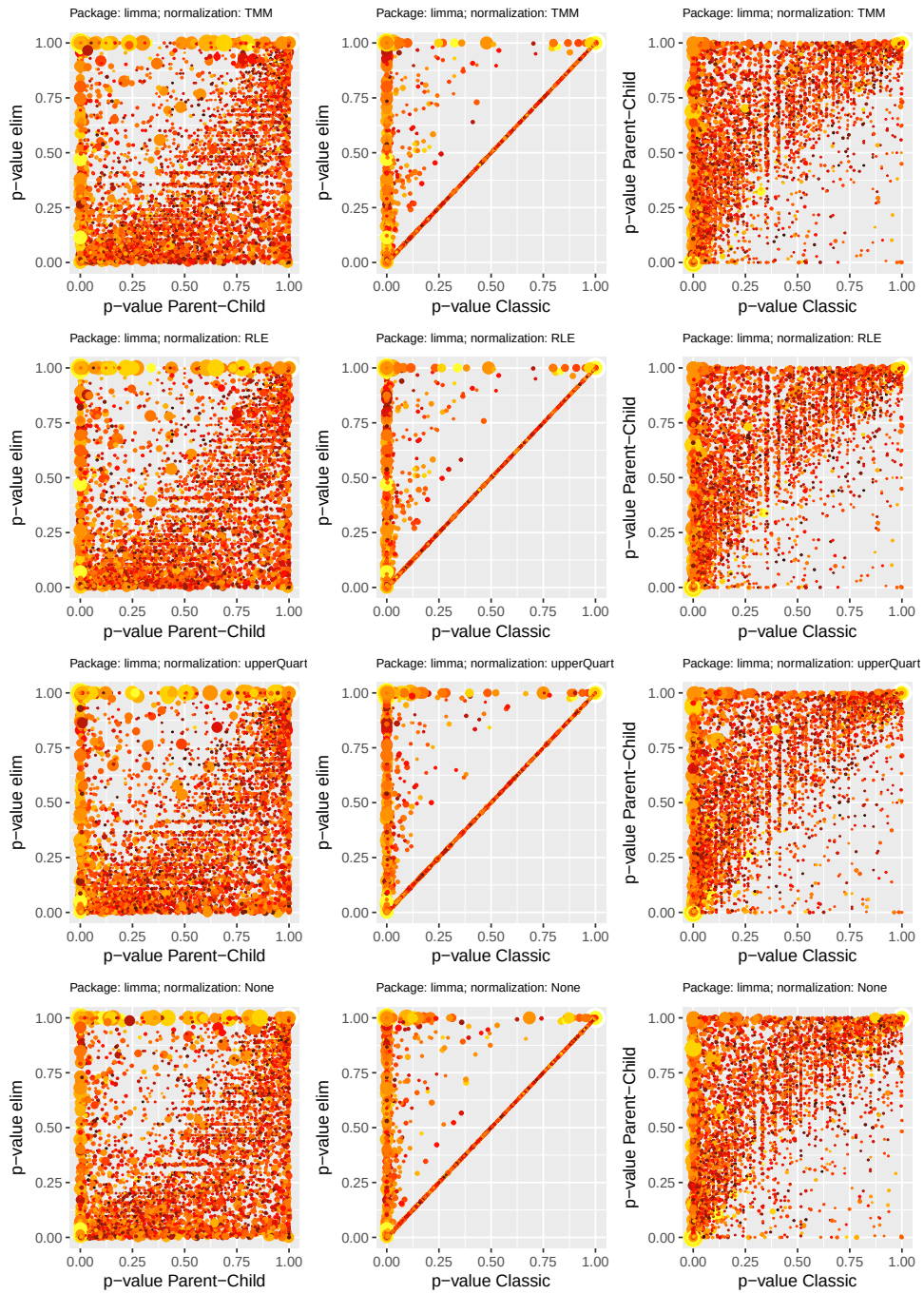

FIGURE A.8: p-value scatter plot algorithm vs algorithm, all the terms in one or the other method have been represented. The size of the dot is proportional to the number of annotated genes for the respective GO term and its coloring represents the depth in the GO hierarchy, with the dark red points being more specific terms than the light yellow ones. These plots were made using the results from `limma` whit the the different normalization methods.

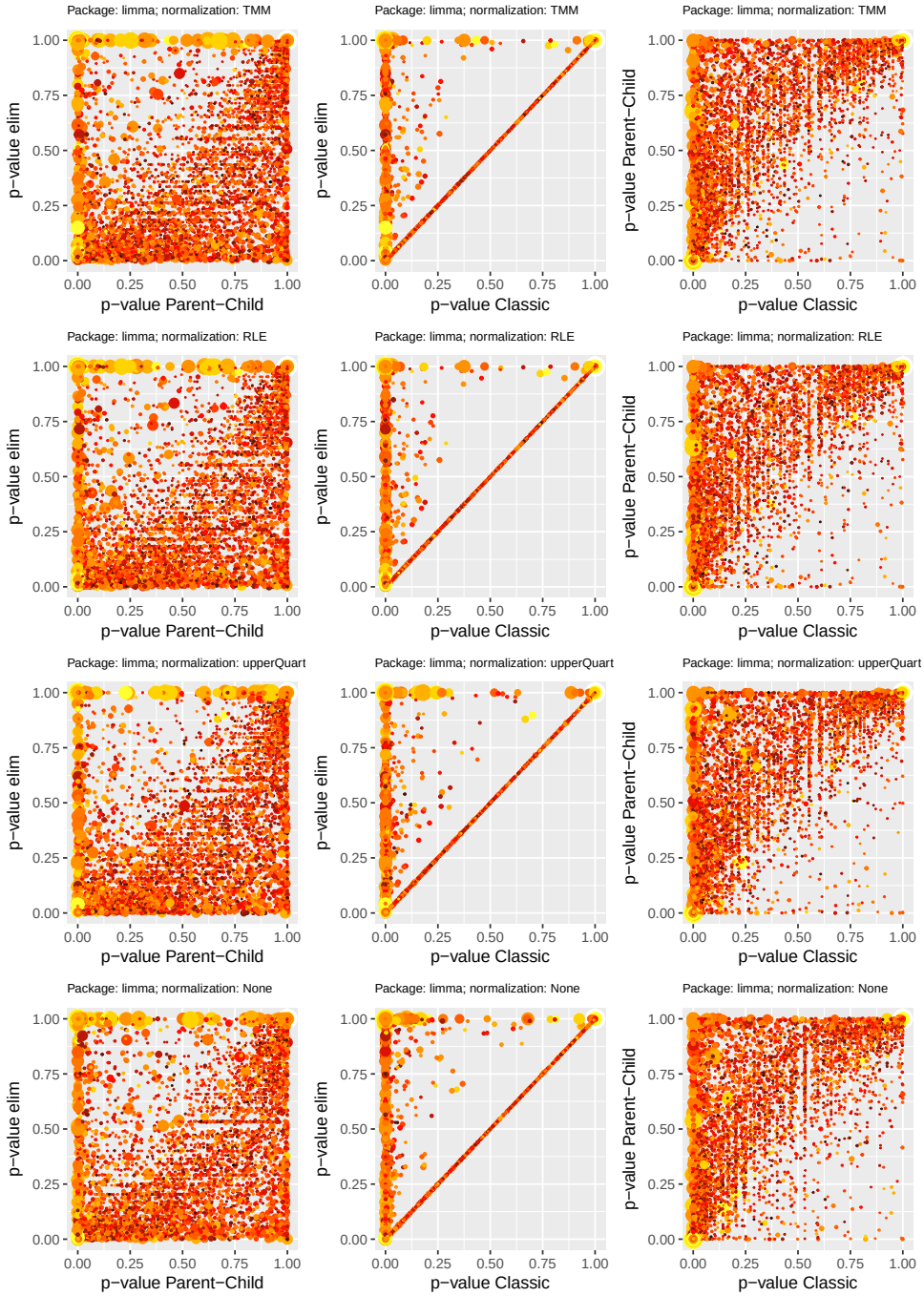

FIGURE A.9: p-value scatter plot algorithm vs algorithm, all the terms in one or the other method have been represented. The size of the dot is proportional to the number of annotated genes for the respective GO term and its coloring represents the depth in the GO hierarchy, with the dark red points being more specific terms than the light yellow ones. These plots were made using the results from `edgeR` whit the the different normalization methods.

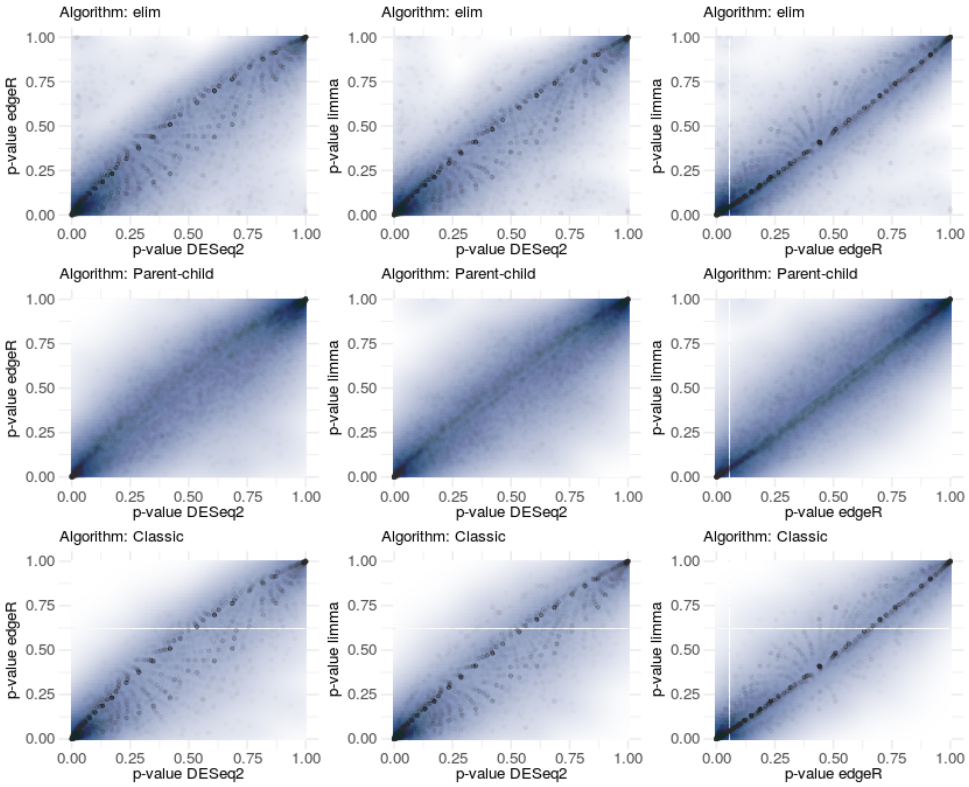

FIGURE A.10: Dot density scatter plot with the p-values of the GO terms calculated using different packages. The darker is the blue, the more dots are in the area. Faded black dots represent the terms' p-values.

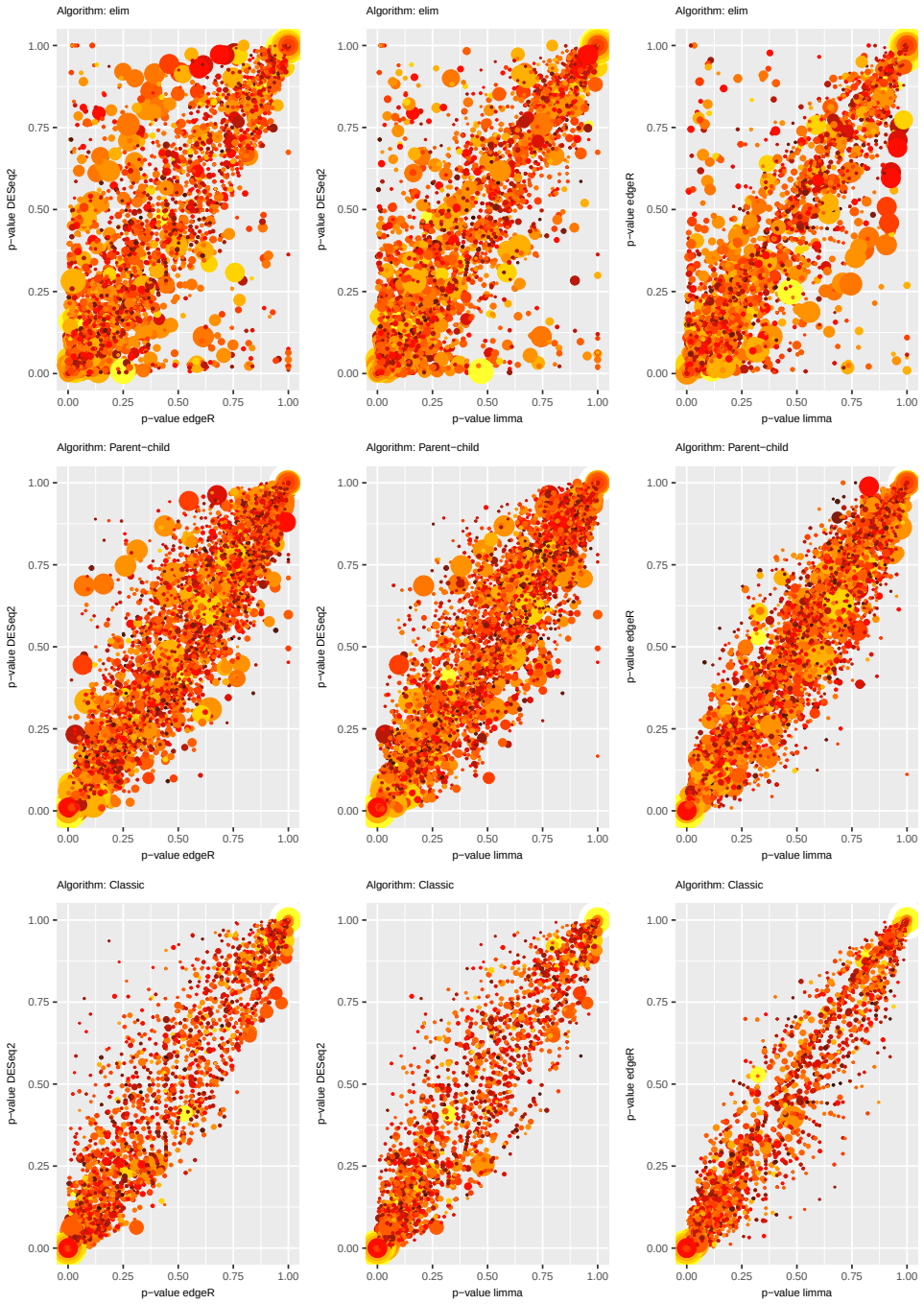

FIGURE A.11: p-values scatter plot for GO terms calculated using the data from different packages. The size and the colour of the dots represent the hierarchy and the number of annotated genes (see figure 4.5).