Identifying the sequence complexity of miRNAs and their functional impact in small-RNA-seq data

Inês Correia Viegas

Mestrado em Bioinformática e Biologia Computacional
Especialização em Bioinformática

Dissertação orientada por:
Andreia de Jesus Amaral Gomes Barbosa Fonseca
Maria Beatriz Duarte Pereira do Carmo

2017
To my family and friends.
Acknowledgments

First of all I want to say a big thank you to my family: my father Paulo, my mother Maria and my sister Catarina. Thank you for your never ending patience throughout this year. Without your support I couldn't have done it. I would also like to thank my grandparents: Beta, Maria, Jorge and Fernando. You are the best. Kiba, my sweet little dog, I couldn't have made it without you too.

I also want to thank to my supervisors Andreia Amaral and Professor Beatriz Carmo as well as to Professor António Ferreira for all the great help in accomplishing this project. This thesis was partially funded by Biosystems and Integrative Sciences Institute (BioISI, FCT/UID/Multi/04046/2013).
Abstract

Since the development of deep sequencing for small non-coding RNAs (small-RNA-seq), several novel microRNAs (miRNAs) have been identified, which led to the observation that miRNAs can vary in length and/or sequence when comparing to their canonical form. These variants, named isomiRs, appear due to an addition or deletion of one or more nucleotides at the 5' or 3' ends or both. Additionally they can also result from internal editings in their sequence. Nowadays, it is already well-established that microRNAs play an important role as regulators of gene expression across multiple species, being critical for maintaining normal physiology and considered candidate biomarkers, regulators, and therapeutic targets for a wide spectrum of diseases. However, numerous recent studies suggest that isomiRs might regulate the expression of different targets in comparison to their respective canonical. Reports indicating differential functionality for isomiRs are still limited to particular variants, and although isomiRs are common, their biological impact is not yet fully understood. The growing number of available tools to perform small-RNA-seq data analyses shows that the interest in obtaining accurate miRNA annotation and quantification is rapidly increasing. However, several tools fail to provide an accurate identification of all forms of isomiRs and to allow a comprehensive analysis of their function.

Here we present the development of the analysis pipeline embedded within the IsomiR Window tool, a bioinformatics tool for accurate annotation, quantification and functional analysis of microRNAs and their isoforms (isomiRs) from small-RNA-sequencing data. The developed pipeline enables the simultaneous processing of multiple data files for two experimental conditions, identifying all types of small non-coding RNAs present in each dataset. It further detects and categorizes all types of isomiRs, such as 5' and 3' miRNA modifications, internal editings and 3' tailings. In addition, the pipeline includes a functional analysis module, providing information related to the targets of selected isomiRs and their functional impact in the cell genetic program in comparison with its canonical form. Additionally, the pipeline offers the possibility to perform novel miRNA prediction, and to add the novel predicted miRNAs to the database file of the species in question, in order to allow the identification of isomiRs derived from these predicted miRNAs, in a subsequent analysis.

We applied this pipeline to analyze simultaneously six small-RNA-seq datasets from either healthy individuals or individuals with hepatitis B. The investigation led to the rapid and accurate identification, quantification and differential expression of several miRNAs and isomiRs. Our analysis allowed to identify significant changes at isomiR level, which were not previously investigated, as well as to identify genes that have not been previously associated with liver damage.

Keywords: miRNA, isomiR, small-RNA-seq, bioinformatics, pipeline
Resumo

A sequenciação de nova geração tornou-se, nos últimos anos, a tecnologia de eleição para o estudo do transcriptoma. Esta metodologia permite a sequenciação de pequenos RNAs não codificantes (small-RNA-seq) a serem expressos numa amostra, tendo contribuído para o aumento, a um ritmo nunca antes visto, da descrição de novos microRNAs (miRNAs) nos genomas de várias espécies. Principalmente, permite caracterizar a complexidade existente numa amostra de RNA, o que acabou por revelar a existência de pequenas variações no que toca ao comprimento e/ou à sequência de miRNAs quando comparados ao respectivo miRNA canónico. Os miRNAs que apresentam estas variações são denominados de isomiRs, sendo que as variações podem incluir a adição ou remoção de um ou mais nucleotídeos nas extremidades da sequência ou podem resultar de eventos de editing no interior da sequência. Os miRNAs são conhecidos por actuar como reguladores de expressão génica em várias espécies, sendo considerados essenciais para manter um bom funcionamento de inúmeras vias biológicas.

No entanto, estudos recentes sugerem que as variações na sequência dos isomiRs têm como consequência uma alteração nos alvos destes (miRNAs), resultando em alterações ao nível da programação genética da célula. Apesar de vários estudos apontarem para este cenário, o impacto biológico dos isomiRs ainda não é extensivamente conhecido. Várias ferramentas têm sido desenvolvidas para a análise de dados de small-RNA-seq com o intuito de identificar isomiRs, no entanto, a maioria das ferramentas, não permite identificar todos os tipos possíveis de isomiRs. Adicionalmente, muitas das ferramentas disponibilizadas não realizam de um modo automatizado a inferência sobre o impacto da expressão destes isomiRs ao nível funcional, ou seja, estudar o impacto destas sequências ao nível dos pathways e de redes de regulação génica da célula.

Por este motivo, o presente projecto teve como finalidade o desenvolvimento de um pipeline que integra a ferramenta IsomiR Window. A ferramenta em questão permite obter a anotação, quantificação e análise funcional de miRNAs/isomiRs provenientes de dados de small-RNA-seq. O pipeline desenvolvido tem a capacidade de receber múltiplos ficheiros de input para um total de duas condições experimentais, permitindo a identificação e quantificação dos diferentes tipos de pequenos RNAs não codificantes presentes em cada dataset. Posteriormente, é capaz de detectar e categorizar todos os tipos de isomiRs: modificações nas extremidades 3’ e 5’ relativamente ao miRNA canónico, eventos de editing internos, adição de tailings na extremidade 3’ relativamente ao miRNA canónico e possíveis combinações entre os diferentes tipos de isomiRs. Adicionalmente, o pipeline inclui uma etapa de análise de expressão diferencial e análise funcional, fornecendo informação relacionada com os targets de isomiRs específicos e com o seu impacto funcional em diversas vias biológicas. O pipeline integra ainda a funcionalidade de previsão de novos miRNAs. Finalmente, embora não de forma automatizada, é possível adicionar os novos miRNAs previstos à correspondente base de dados das espécies em estudo, permitindo, numa análise subsequente, a identificação de isomiRs derivados de potenciais novos pre-miRNAs.

De modo a validar o pipeline desenvolvido, analisaram-se seis datasets que incluíram amostras de indivíduos saudáveis e amostras de indivíduos infectados com hepatite B. Esta análise incluiu a identificação, quantificação, análise de expressão diferencial e por fim, a previsão de alvos para os isomiRs de interesse. Como resultado, identificámos alterações significativas na expressão de alguns isomiRs que não tinham sido anteriormente reportados. Adicionalmente, a análise funcional permitiu identificar genes, que de acordo com a literatura, não têm sido associados a lesões no fígado.

Palavras-chave: miRNA, isomiR, small-RNA-seq, bioinformática, pipeline
Resumo Alargado

Os microRNAs (miRNAs) são pequenos RNAs não codificantes que regulam a expressão génica ao nível pós-transcripcional. Esta regulação é efectuada por lições de homologia que envolvem toda a sequência do miRNA, ou por homologia parcial, envolvendo só uma região do miRNA localizada entre o segundo e o sétimo nucleótido numa região que se chama seed. Deste modo, um único miRNA regula a expressão de centenas de diferentes genes. Estudos recentes reportam a existência de variantes de miRNAs, nomeados isomiRs, que apresentam alterações nas suas sequências quando comparados ao respectivo canónico. Os isomiRs podem ser classificados em quatro categorias principais: isomiRs com modificações na extremidade 5’, isomiRs com modificações na extremidade 3’, isomiRs com eventos de editing na sequência e isomiRs com tailings na extremidade 3’. Visto que os miRNAs se ligam aos seus alvos (mRNAs) por homologia da sua sequência, os isomiRs podem regular diferentes alvos e implicarem uma alteração nas vias celulares afectadas em comparação com o miRNA canónico.

A rápida evolução das tecnologias de sequenciamento de alto débito (vulgarmente referenciados como NGS) tem potenciado a obtenção de um grande volume de informação que permite um melhor conhecimento sobre o genoma funcional, no qual se inclui o transcriptoma. No entanto, esta evolução não tem sido acompanhada pelo desenvolvimento de ferramentas bioinformáticas que permitam, a utilizadores com fracos ou nenhuns conhecimentos de programação e bioinformática, realizar a análise de dados NGS.

Existem actualmente publicadas cerca de treze ferramentas que se propõem a analisar dados de sequenciamento de miRNAs, sendo que só oito permitem analisar dados provenientes de NGS. Algumas ferramentas permitem a anotação de diferentes tipos de pequenos RNAs não codificantes (sncRNAs) existentes nos datasets. Relativamente à identificação de isomiRs, algumas ferramentas têm a capacidade de identificar modificações nas extremidades 5’ e 3’, no entanto são poucas as que procedem à identificação de eventos de editing e ocorrência de tailings. Apenas três das oito ferramentas incluem a previsão de targets para os miRNAs/isomiRs identificados, sendo que apenas duas prosseguem para a análise ontologias. Foi ainda possível verificar que apenas duas ferramentas, das oito ferramentas comparadas, incluem a previsão de novos miRNAs no seu workflow.

Deste modo, verificou-se que não está disponível para a comunidade científica uma ferramenta que permita realizar uma análise completa de dados de small-RNA-seq, permitindo analisar paralelamente vários conjuntos de dados de NGS de uma forma integrada, isto é, procedendo à anotação de todos os tipos de sncRNAs e à previsão de novos pre-miRNAs e miRNA; fazendo a identificação de todos os tipos de isomiRs, prosseguindo automaticamente para a realização de uma análise de expressão diferencial entre duas condições experimentais e finalmente efectuando o estudo do impacto funcional dos isomiRs identificados como diferencialmente expressos. Neste âmbito, o projecto IsomiR Window envolveu duas importantes vertentes. Uma das vertentes, na qual se centra este trabalho, teve como objectivo o desenvolvimento de um pipeline que integra todas as referidas análises, permitindo a um utilizador com nível inicial de conhecimentos de bioinformática a sua execução. A segunda vertente, consistiu no desenvolvimento de uma interface web (que não se encontra nos objectivos deste projecto) que permite a um utilizador sem conhecimentos de bioinformática analisar dados de NGS com o objectivo de estudar sncRNAs com especial enfoque nos isomiRs. O pipeline desenvolvido tem a capacidade de: (i) receber múltiplos datasets de duas condições experimentais diferentes; (ii) realizar uma caracterização geral da população de pequenos RNAs não codificantes presentes; (iii) identificar e categorizar todos os tipos de
isomiRs; (iv) prever novos miRNAs; (v) proceder à análise de expressão diferencial de isomiRs entre as amostras; (vi) proceder à previsão de alvos dos isomiRs; (vii) identificar as vias biológicas em que os alvos dos isomiRs estão envolvidos.

O pipeline, implementado em PERL, tem a capacidade de receber ficheiros em formato FASTQ para duas condições experimentais, sendo que a totalidade do workflow inclui a utilização de seis scripts, divididos pelos dois módulos de análise: anotação e funcional.

O primeiro passo deste workflow consiste na verificação da existência de inputs válidos, neste caso um FASTQ file. Após a verificação de inputs, segue-se o passo que inclui o alinhamento, utilizando-se, para isso, o software Bowtie. Posteriormente, é efectuada uma verificação ao ficheiro resultante do alinhamento (convencionalmente em formato SAM) de modo a garantir que foi convenientemente criado. Este ficheiro é posteriormente utilizado para a caracterização geral da população de sncRNAs presentes nos conjuntos de dados, utilizando para isso a função HTSeq (biblioteca PYTHON) que compara as coordenadas genómicas do ficheiro SAM com as coordenadas genómicas de sncRNAs disponíveis na base de dados RNA central para a espécie em estudo. Nesta fase da análise é possível proceder à utilização do miRDeep2 para prever novos miRNAs, sendo que este passo da análise é opcional, sendo apenas efectuado se o utilizador assim o desejar. Para a etapa de identificação e classificação dos vários tipos de isomiRs foi desenvolvido um algoritmo baseado na comparação das coordenadas genómicas das sequências obtidas após o alinhamento, com as coordenadas de pre-miRNAs e de miRNAs disponíveis na miRBase para a espécie em estudo. Aquando a identificação de um isomiR é-lhe atribuído um identificador único e intuitivo que permite perceber quais os tipos de modificações que possui relativamente ao respectivo canónico. Ao longo destas primeiras etapas do workflow são produzidos vários ficheiros de resultados, sendo que o pipeline é capaz de retirar as informações consideradas mais relevantes, construindo ficheiros adicionais com estas informações, nomeadamente: tipos de modificações mais abundantes nos datasets, tipos de editings e tailings mais comuns, entre outros, que permitem facilmente obter sumários estatísticos e realizar visualizações de resultados através de gráficos.

A etapa final do módulo da anotação inclui a utilização de um pacote de R, Deseq2, para testar a existência de expressão diferencial entre os isomiRs identificados nos conjuntos de dados das duas condições experimentais do estudo, sendo possível efectuar o teste para um valor-p ajustado de 0.05 ou 0.1. Caso se verifique a presença de isomiRs/miRNAs diferencialmente expressos entre as duas condições é possível prosseguir para o módulo da análise funcional com esta mesma lista de isomiRs/miRNAs.

O módulo de análise funcional consiste na previsão de genes alvo para o conjunto de isomiRs/miRNAs proveniente da etapa anterior. É possível proceder à previsão de genes alvo através da utilização de duas ferramentas: TargetScan e miRanda, sendo que ambas as ferramentas podem ser utilizadas em simultâneo ou individualmente. Por fim, o workflow inclui ainda a análise de enriquecimento de termos ontológicos relativos a processos biológicos para os genes previamente previstos, utilizando um pacote de R - topGO, de modo a que seja possível identificar alterações biológicas relevantes, incluindo por exemplo, alterações de funcionalidade por parte dos isomiRs. A utilização do módulo da análise funcional pode ainda ser feita de modo independente, sendo que pode ser fornecido como input, um ficheiro FASTA não proveniente do módulo da anotação.

Para além dos scripts incluídos no pipeline descrito, foram ainda desenvolvidos mais dois scripts independentes. O primeiro tem como intuito possibilitar a previsão de novos miRNAs pelo miRDeep2 de forma independente do restante workflow. Deste modo é possível proceder à utilização do miRDeep2 posteriormente à restante análise. Adicionalmente, existe um segundo script externo que permite adicionar os miRNAs previstos pelo miRDeep2 à base de dados de miRNAs, para identificação de isomiRs para a
espécie em estudo, facilitando assim a identificação de modificações nestes novos miRNAs, numa próxima análise.

Ao longo do desenvolvimento do pipeline foram efectuados testes para verificar a sua eficiência em termos de tempo e de precisão. De modo a validar o pipeline desenvolvido, foram extraídos datasets de acesso público provenientes de um estudo que teve como intuito identificar miRNAs como biomarcadores para identificar a existência de patologias associadas a lesões no fígado. De todos os datasets disponíveis, foram extraídos 3 datasets de indivíduos saudáveis e 3 datasets de indivíduos infectados com hepatite B. Depois de efectuada a análise integrada do conjunto dos dados foi possível verificar que a maior parte da população de sncRAs é constituída por miRNAs, sendo que o miR92a-3p foi o miRNA identificado como sendo o mais abundante, quer em indivíduos saudáveis, quer em indivíduos com hepatite B. Estes resultados mostraram estar em concordância com os resultados publicados no estudo original, de onde provêm os dados. Adicionalmente, identificaram-se três isomiRs diferencialmente expressos entre as duas condições, sendo que, aquando a realização da análise funcional, foi possível identificar alguns genes que não têm sido associados a sintomatologias relacionadas com lesões ao nível do fígado.
Index

List of Figures ................................................................................................................................. xv
List of Tables .................................................................................................................................. xvii
1. Introduction .................................................................................................................................. 1
   1.1. What are IsomiRs? .................................................................................................................. 1
   1.2. Existing tools for IsomiR analysis .......................................................................................... 3
      1.2.1. Annotation analysis ......................................................................................................... 6
      1.2.1.1. MiRNAs/isomiRs identification ...................................................................................... 6
      1.2.1.2. Differential expression ................................................................................................ 8
      1.2.2. Functional analysis .......................................................................................................... 8
      1.2.2.1. MiRNA-mRNA targeting .............................................................................................. 8
      1.2.2.2. Gene-Enrichment analysis .......................................................................................... 11
      1.2.2.3. Prediction of novel miRNAs ........................................................................................ 11
   1.3. MiRNAs and isomiRs as biomarkers .................................................................................... 12
2. Goals ............................................................................................................................................. 13
3. Materials and methods ................................................................................................................ 13
   3.1. Third-party software integration ........................................................................................... 13
   3.2. Databases ............................................................................................................................. 14
   3.3. Workflow benchmarking ....................................................................................................... 15
4. Results and Discussion ................................................................................................................ 17
   4.1. Workflow implementation ..................................................................................................... 17
      4.1.1. Annotation analysis ......................................................................................................... 19
      4.1.1.1. Alignment and sncRNAs classification ......................................................................... 19
      4.1.1.2. Prediction of novel miRNAs ......................................................................................... 20
      4.1.1.3. IsomiRs identification and classification .................................................................... 21
      4.1.1.4. Differential expression ............................................................................................... 25
      4.1.2. Functional analysis ......................................................................................................... 27
      4.1.3. Reporting results ............................................................................................................ 29
      4.1.4. Pipeline usage ................................................................................................................. 30
   4.2. IsomiR Window pipeline analysis of small-RNA-seq data derived from plasma of hepatitis B patients ......................................................................................................................... 31
5. Conclusions ................................................................................................................................... 43
6. References ..................................................................................................................................... 45
7. Appendices ................................................................................................................................. 53

7.1. Folder organization in the IsomiR Window tool context ................................................. 53

7.2. Usage of the scripts from the command line ................................................................. 54
List of Figures

Figure 1.1. MiRNA biogenesis ................................................................................................. 1
Figure 1.2. Types of isomiRs. .................................................................................................. 2
Figure 1.3. FASTQ format. ..................................................................................................... 3
Figure 1.4. FASTA format ........................................................................................................ 4
Figure 4.1. Schematic representation of the IsomiR Window tool ........................................ 17
Figure 4.2. Flow chart of analysis modules ............................................................................. 18
Figure 4.3. Detailed diagram of the annotation analysis module ............................................. 22
Figure 4.4. The isomiR identifier ............................................................................................ 24
Figure 4.5. Experimental design file for unpaired samples analysis ....................................... 26
Figure 4.6. Detailed diagram of the functional analysis module .............................................. 28
Figure 4.7. Example of Miranda output file ............................................................................ 28
Figure 4.8. TargetScan output file .......................................................................................... 29
Figure 4.9. Comparison of the read length distribution across the 3 samples of each condition. .......................................................... 32
Figure 4.10. General characterization of all types of small non-coding RNAs present across both conditions .................................................................................................................. 33
Figure 4.11. Comparison of the abundance of miRNAs that derived from the 3p or 5p arm of the pre-miRNA ............................................................................................................................................... 33
Figure 4.12. MiRNA’s abundancy across both conditions ....................................................... 34
Figure 4.13. Types of isomiRs present in the samples of each condition .................................. 35
Figure 4.14. Types of tailings that appear in the samples of both conditions ......................... 35
Figure 4.15. Circulating miRNAs/isomiRs among the two different experimental conditions .................................................................................................................. 37
Figure 4.16. Schematic representation of two isomiRs of the hsa-miR-92a-3p ....................... 39
Figure 4.17. Schematic representation of an isomiR of the hsa-miR-92a-3p ............................ 40
Figure 7.1. Folder structure and organization within the IsomiR Window tool ........................ 53
List of Tables

Table 1.1. Overview of the different isomiRs analysis tools. .............................................................. 5
Table 1.2. Brief description of the SAM file format. .............................................................................. 7
Table 1.3. Overview of miRNA target prediction tools. ......................................................................... 9
Table 3.1. External software included in the pipeline. ......................................................................... 14
Table 4.1. Availability of database files for the species accepted by the pipeline. ......................... 19
Table 4.2. Settings defined for variant calling using SAMtools mpileup and BCFtools. .............. 23
Table 4.3. Examples of isomiRs identifiers......................................................................................... 25
Table 4.4. Example of DESeq2 output file.......................................................................................... 26
Table 4.5. Summary files produced for the annotation analysis module. ....................................... 29
Table 4.6. Number of sequences in the FASTQ files before and after quality filtering................. 31
Table 4.7. Summary statistics provided by Bowtie............................................................................. 32
Table 4.8. Table produced by the DESeq2 output (TableDE_*.txt). ............................................... 38
Table 4.9. Gene lists obtained from miRanda target predictor......................................................... 40
Table 4.10. Enriched GO terms that resulted from topGO analysis.............................................. 41
Table 4.11. Gene list obtained from miRanda target predictor for the UAUUGCACUUGUCCCGGC_hsa-
mir-92a-3p_0_4_0_0_0_0_NA_0_NA_F_NA. ................................................................................. 42
Table 7.1 Individual scripts commands and their usage...................................................................... 54
1. Introduction

1.1. What are IsomiRs?

In the past decade, microRNAs (miRNAs) have emerged as important regulators of gene expression among various species. This family of small non-coding RNAs (sncRNAs), with usually ~22 nucleotides of length, has been considered to play an essential role in several biological processes, such as developmental control, apoptosis, cell differentiation and proliferation. Also, they have been frequently implicated in various disease processes thus emerging as potential targets of therapeutic intervention [1]–[3].

The process of miRNA biogenesis includes multiple steps (Figure 1.1). First, the primary miRNA (pri-miRNA) is transcribed by RNA polymerase II [4]. Next, this pri-miRNA undergoes processing by two endoribonucleolytic enzymes. Drosha, an RNase III endonuclease, forms a complex with DGCR8 (DiGeorge syndrome critical region gene 8), being able to bind to stem loops within the pri-miRNA and to release the precursor miRNA (pre-miRNA) [5], [6]. The complex Drosha-DGCR8 is therefore responsible for the first endonucleolytic reaction, whereas the DGCR8 functions as a molecular ruler that positions the Drosha cut site 11 bp from the base of the hairpin stem [7]. The resulting hairpin of pre-miRNA has a length of approximately 70 nt and it is further transported into the cytoplasm by Exportin-5, where it is processed by Dicer (a RNase III endonuclease) giving origin to an approximately 22 nt miRNA-miRNA duplex [8], [9]. This miRNA-miRNA duplex is then loaded into the protein Argonaut 2 (Ago2), where the strand that is complementary to the target mRNA is selected to be incorporated into the RNA Induced Silencing Complexes (RISCs), while the other strand is further degraded [10]. MiRNAs act at the post transcriptional level via base-pairing with complementary sequences mostly on the 3’UTR of its target mRNAs, requiring only homology in the seed sequence (nucleotides 2–7 of the 5′ end), meaning that a single miRNA can regulate the expression of thousands of different genes [11].

![Figure 1.1. MiRNA biogenesis. The figure shows the main stages of miRNA biogenesis from the nucleus to the cytoplasm [11].](image-url)
Since Next-Generation Sequencing (NGS) technology has emerged, the development of small-RNA-seq enabled the discovery of novel small non-coding RNAs species, revealing novel insights regarding the biology of these molecules. This group of non-coding RNAs includes transfer RNA (tRNA), ribosomal RNA (rRNA), microRNA (miRNA), small nucleolar RNA (snoRNA), small nuclear RNA (snRNA), small interfering RNA (siRNAs), and piwi-interacting RNA (piRNA) [12], [13]. Furthermore, several previous studies, in which small-RNA-seq libraries were generated from different mammalian and plant tissues, reported the production of miRNA variants of the canonical mature sequences, called isoforms of miRNAs (isomiRs) [14], [15], [16].

IsomiRs can be classified into four main categories: 5’ isomiRs, 3’ isomiRs, isomiRs with internal modifications, and isomiRs with 3’ tailings (Figure 1.2). Within 5’ and 3’ we can identify two types of isomiRs, whether they suffered, in relation to the canonical, differential Dicer processing which consequently resulted in the addition or trimming of one or more nucleotides at the 5’ or 3’ ends or both. Such alterations are referred to as ‘templated’, since the miRNA sequences will display total homology to their corresponding pre-miRNA [17], [18]. Furthermore, there are two classes of non-templated isomiRs, (that do not display total homology with the pre-miRNA): isomiRs with 3’ tailings (mainly adenylation or uridylation events) as a result from enzymatic modification [19] and isomiRs holding internal editions. Considering this, an isomiR can be referred to as templated or non-templated whether it has complete homology to the pre-miRNA or not. Besides the types of modifications mentioned above, internal editing events within the miRNA sequence have been reported as being a result from enzymatic modification [20]. Importantly, internal editions can only be considered after discarding the occurrence of nucleotide polymorphisms (SNPs). Although miRNAs holding SNPs are not considered as being part of the isomiR class, identifying SNPs in miRNAs is really crucial, since SNPs in the seed region (bases 2-7) are shown to affect phenotypes and disease vulnerability [21]. Therefore, to be able to identify SNPs in the seed-region could help in the further comprehension of the molecular mechanism of gene dysregulation.

Although the most frequently observed type of isomiRs in animals and plants is the 3’ isomiR, in terms of both the number of miRNAs displaying these variations and their overall abundance, 5’ isomiRs represent a significant proportion of the population of some miRNAs [22]–[24], and since the seed region is located in the 5’end, these isomiRs may potentially regulate a different set of genes in comparison with their corresponding canonical feature and, if highly expressed, may have great impact in the genetic programming of the cell [25].

**Figure 1.2. Types of isomiRs.** The figure shows a schematic representation of a miRNA precursor, its possible mature sequence, and the variations that can occur regarding the respective canonical sequence. Non-templated additions at the 3’ end (tailing) are represented in red, a nucleotide change inside the sequence is represented in yellow (internal editing) and templated additions and trimmings are represented in dark grey.
1.2. Existing tools for IsomiR analysis

Several tools have been developed aiming to identify isomiRs, which include: miRGator [26] and its more recent version, miRGAtor v3.0 [27], miRanalyzer [28], SeqBuster [29], CPSS [30], isomiRID [31], isomiRex [32], miRspring [33], IsomiRage [34], YM500 v2 [35], miRMOD [36], DeAnnIso [20], miR-isomiRExp [37] and isomiR-SEA [38]. By comparing all of these 13 tools it became evident that, although they all intend to investigate the presence of isomiRs, each tool presents different features. For instance, in Table 1.1, it is shown that, some of the tools are also able to detect other types of small non-coding RNAs that are present in a specific sample, such as miRanalyzer and CPSS, while others are only specialized in isomiR detection and additional analysis of their expression profiles in small-RNA-seq data, namely miRGAtor v3.0 and DeAnnIso. Among all of these tools, only a few allow to use small-RNA-seq data as input (Table 1.1): miRanalyzer, CPSS, DeAnnIso, isomiRex, miR-isomiRExp, mirSpring, IsomiRage and isomiRID.

However, miRanalyzer, for instance, does not accept directly the output from NGS techniques, since it provides a PERL script to turn this data, of usually several gigabytes (GB), to a multifasta format file with only a few megabytes (MB). On the other hand, CPSS manual specifies that this tool has no limit of size for the input file, however, it only takes FASTA files as input. Both CPSS and DeAnnIso only allow the analysis of one FASTA file at a time and, contrary to CPSS, DeAnnIso establishes a maximum size of 50 MB, which is not a typical size for NGS data. Both isomiRex and miR-isomiRExp are described as being tools that take small-RNA-seq raw data as input, however, at the present time, the provided sites for accessing the tools are offline. All the previously mentioned tools are web based tools, which may justify some of the limitations for accepting large input files. Stand-alone tools, such as mirSpring, IsomiRage and IsomiRID do not report a limitation regarding input file size. MirSpring and IsomiRage, however, only take previously mapped reads as inputs for further analysis. IsomiRID, on the other hand, accepts sequence files in FASTQ (Figure 1.3) or FASTA format (Figure 1.4).

![Figure 1.3. FASTQ format. A FASTQ format file displays a repetition pattern every four rows, sequence identifier, sequence, “+” symbol, which can be followed by a repetition of the sequence identifier and the corresponding sequence quality scores in ASCII format.](image-url)
From the extensive analysis of these available tools it was possible to understand that there are a variety of important features that should be considered when we intend to analyze small-RNA-seq data in a comprehensive manner, in this case with particular emphasis on isomiRs identification. These features are mainly divided into two major classes of analysis: annotation and functional impact. As part of the annotation analysis, the following features should be included: sequence annotation (classification of reads according to the corresponding type of sncRNA: miRNAs and other non-coding RNAs - snoRNA, snRNA, rRNA, etc), detection of all types of IsomiRs (templated and non-templated), novel miRNA prediction and differential expression. The functional analysis should comprise isomiR and canonical miRNA target prediction, identification of GO terms for the identified targets and gene set enrichment analysis, which will allow to understand the impact of isomiR production in the cell genetic program. We next present, in detail, a review of the ability of the available tools to perform this set of analysis as well as regarding the available third party software.
Table 1.1. Overview of the different isomiRs analysis tools. Comparison of the major features of 8 tools that intend to perform small-RNA-seq data analysis.

<table>
<thead>
<tr>
<th>Feature</th>
<th>miRanalyzer</th>
<th>CPSS</th>
<th>isomiRID</th>
<th>isomiRex</th>
<th>miRspring</th>
<th>IsomiRage</th>
<th>DeAnnIso</th>
<th>miR-isomiRExp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length distribution of reads</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Annotation of reads</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MiRNA detection</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Other non-coding RNA detection (snoRNA, snRNA, rRNA...)</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>miRNA editing detection</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>miRNA modification detection</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Novel miRNA prediction</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Differentially expressed miRNA identification</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>miRNA target prediction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GO analysis for miRNA targets</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Pathway analysis for miRNA targets</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>
1.2.1. Annotation analysis

1.2.1.1. MiRNAs/isomiRs identification

From the 8 tools that are able to process small-RNA-seq data, only three are described as being able to detect all kinds of isomiRs (Table 1.1). These include: CPSS, DeAnnIso and isomiRID. However, only isomiRID is in fact able to categorize isomiRs as having 3’-end and 5’-end modifications, isomiRs with internal editings and isomiRs with non-templated additions at the 3’ end (tailings).

CPSS performs the identification of 3’-end 5’-end isomiRs and isomiRs with internal editings by matching query sequences to the miRBase repository, being able to detect all types of modifications. However, in order for them to be considered as true modifications, each modification has to be counted at least 3 times in the data and to contribute in more than 10% to the total number of all reads annotated to the same miRNA. Additionally, this tool estimates a Z-score to ascertain the error associated with identified nucleotide changes in the sequence.

The second tool, DeAnnIso, for the detection of 3’-end and 5’-end isomiRs uses the same type of strategy as CPSS, using BLAST [20] to match the genome mapped reads with precursor sequences obtained from miRBase and also to match precursor sequence matched reads with mature miRNA downloaded from miRBase. To finish this initial step of the workflow all the detected isomiRs are aligned with their canonical miRNAs in order to extract the variant nucleotides for further classification. Regarding the detection of isomiRs with internal editings, after BLAST application to match the genome mapped reads with precursor sequences from miRBase, this tool performs another BLAST to match precursor sequence matched reads with mature miRNA from miRBase. Although BLAST alignment is more accurate, due to the use of Smith-Waterman algorithm, it results in more processing time, not ideal for small-RNA-seq data [39]. Following this procedure, all the detected isomiRs are aligned to their canonical miRNAs in order to extract the variant nucleotides. From this alignment, isomiRs with internal modifications are classified into two categories: internal modifications with seed shifting and internal modifications without seed shifting. Afterwards, it is important to determine if these found variations are indeed internal editings or if they are originated from SNPs. To resolve this matter, the tool performs a match of these variations to the SNP database (dbSNP), and if they are already identified as being a SNP, information about the SNP is shown in the results page.

The third mentioned tool, isomiRID, is an available command line Python script that can analyze FASTA or FASTQ files, being able to detect all types of isomiRs, including modifications, editings and tailings. This tool follows a different method for isomiR detection, through an iterative process, the algorithm starts by searching for perfect matches between the query sequence and the miRNA and pre-miRNA dataset, identifying 3’ and 5’ end isomiRs and then iteratively starts increasing the number of mismatches allowing the identification of isomiRs with internal nucleotide modifications and tailings.

Other tools, such as IsomiRage and IsomiRex are only able of detecting some types of isomiRs (3’ end and 5’ end modifications), using a similar strategy as CPSS and DeAnnIso, making use of a reference dataset consisting of all known mature and precursor sequences of miRNAs downloaded from miRBase and mapping reads using Bowtie [32], [34]. The usage of Bowtie is less accurate, when comparing to BLAST, however, it requires less processing time [40], being more adequate for NGS data and producing an alignment file in SAM format as output, which is a generic alignment format for storing read alignments against reference sequences that includes a header section and an alignment section (Table
1.2). IsomiRage is only able to detect modifications when they do not occur in both 5’ and 3’ ends simultaneously using a different strategy based in sequence matching. Finally, MirSpring and miR-isomiRExp are both able to identify and determine the percentage of 5’ and 3’ isomiRs in a small-RNA-seq dataset, but they are not capable of categorizing each isomiR into its corresponding type.

Regarding isomiRs with internal editings, only CPSS, isomiRID and DeAnnIso have the ability to detect them. CPSS classifies reads into miRNA editing if it appears at least 3 times in the data and if each editing contributes in more than 10% to the total number of all reads annotated into the same miRNA. IsomiRID is able to detect these editings by allowing a single mismatch when mapping is performed. In the end of the workflow a file is generated containing sequences with one mismatch in the 5’ end, in the 3’ end or in the middle. These nucleotide variations detected as a middle mismatch are then considered as being a SNP or an A-I editing depending if these exist or not in dbSNP. Finally, DeAnniso, CPSS and isomiRID are the only tools that are able to detect tailings, however, this nomenclature is never used, being only referred as non-templated additions at the 3’ end. Furthermore, it is never mentioned in the description of this tools, the number of nucleotides that they consider for the extension to be a tailing.

Table 1.2. Brief description of the SAM file format. Following the header lines, the SAM file includes the alignment lines, comprising several columns described in the table.

<table>
<thead>
<tr>
<th>Column</th>
<th>Field</th>
<th>Brief description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>QNAME</td>
<td>Query template name</td>
</tr>
<tr>
<td>2</td>
<td>FLAG</td>
<td>bitwise FLAG</td>
</tr>
<tr>
<td>3</td>
<td>RNAME</td>
<td>Reference sequence name</td>
</tr>
<tr>
<td>4</td>
<td>POS</td>
<td>1-based leftmost mapping position</td>
</tr>
<tr>
<td>5</td>
<td>MAPQ</td>
<td>Mapping Quality</td>
</tr>
<tr>
<td>6</td>
<td>CIGAR</td>
<td>CIGAR string</td>
</tr>
<tr>
<td>7</td>
<td>RNEXT</td>
<td>Reference name of the mate/next read</td>
</tr>
<tr>
<td>8</td>
<td>PNEXT</td>
<td>Position of the mate/next read</td>
</tr>
<tr>
<td>9</td>
<td>TLEN</td>
<td>Observed Template length</td>
</tr>
<tr>
<td>10</td>
<td>SEQ</td>
<td>Segment sequence</td>
</tr>
<tr>
<td>11</td>
<td>QUAL</td>
<td>ASCII of Phred-scaled base QUALity+33</td>
</tr>
</tbody>
</table>
1.2.1.2. **Differential expression**

Since isomiRs can play important roles under both physiological and disease conditions, it is important that a tool of analysis is able to infer which isomiRs and miRNAs are being differentially expressed in the experiment. Of the 8 tools presented in Table 1.1, 7 of them present this feature. IsomiRID is the only one that is not able to detect differentially expressed isomiRs. MiRanalyzer, isomiRex and miR-isomiRExp are all able to identify differentially expressed isomiRs species using the DESeq package [41] of Bioconductor [42], which tests for differential expression by the use of the negative binomial distribution and a shrinkage estimator for the distribution’s variance [43]. CPSS is also able to detect differentially expressed isomiRs in paired samples, through a different method. To do so, the counts of mapped sequences are normalized using the Reads per Million method (RPM), which it is also referred to as library-size normalization, and then compared between the samples. CPSS uses two statistical test methods according to two different conditions. When the sequence count is smaller than 5, Fisher's exact test is used for judgment, and chi-square test is used instead if the count is larger than 5. In IsomiRage, sequences are mapped to a reference genome using Bowtie and, sequence counts are then normalized using RMA [44] and RPKM [45] methods. Both CPSS and IsomiRage use normalization methods that are not the most suitable for analyzing small-RNA-seq data. The referred methods used by these tools to perform count normalization perform well for comparison of gene expression within a library but tend to perform poorly when samples have heterogeneous transcript distributions [46].

Finally, DeAnnIso applies Wilcoxon Rank-sum test to infer the statistically significant differences for a particular experimental group and mirSpring only detects differently expressed isomiRs within miRNA clusters.

1.2.2. **Functional analysis**

Following isomiR identification, understanding their functional implications in the control of biological pathways in the cell is crucial. These questions are usually addressed by performing target prediction followed by gene enrichment analysis. To do so, many computational methods have already been developed. A few examples of how to perform miRNA-target prediction and gene enrichment analysis will be described below.

1.2.2.1. **miRNA-mRNA targeting**

Regions of high complementarity have been observed in almost all known miRNA-mRNA interactions in animals [47]. As mentioned before, the interaction occurs between the miRNA “seed” region, which is located at the 5’ part of the miRNA between 2-7 nucleotides and the 3’UTR of the mRNA (although less frequent interactions to other regions of the mRNA molecules have been reported) [48]. Although the exact rules regarding the interaction of miRNA molecules with their targets are not fully understood, the different prediction algorithms currently available are based on different principles, being categorized into 3 different groups (Table 1.3): sequence-based, energy-based and machine learning-based groups. The tools included in first group, take the complementarity of the sequences into
greater account, including TargetScan [49] and miRanda [50]. On the other hand, the second group of tools, which includes RNAlhybrid [51], PITA [52] and RNA22-GUI [53], use thermodynamics as the main principle to perform target prediction. Finally, the tools belonging to the third group bring together different types of features and apply machine learning techniques to find the feature patterns shared by true miRNA-target interactions, including mirTarget2 [54] and DIANA microT-CDS [55]. Among all the mentioned tools for target prediction, the following have the source code available to download, in order to be possible its inclusion in a pipeline: TargetScan, miRanda, PITA and RNAlhybrid. Due to the nature of miRNA and mRNA interactions, thousands of targets are identified for a single miRNA. Performing experimental validation of all these targets is not yet feasible and therefore, researchers rely in the targets that are commonly identified by several predictors as a way to prioritize experimental validations.

Table 1.3. Overview of miRNA target prediction tools. A comparison between the type of method, organisms’ availability and required input, of miRNA target prediction tools is shown in the table.

<table>
<thead>
<tr>
<th>Tools</th>
<th>Type of method</th>
<th>Organisms</th>
<th>Input Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>TargetScan</td>
<td>sequence-based</td>
<td>Human, mouse, rat, chimpanzee, rhesus, cow, dog, opossum, chicken, frog, fly, zebrafish and worm</td>
<td>MiRNA name, gene name or miRNA family</td>
</tr>
<tr>
<td>MiRanda</td>
<td>sequence-based</td>
<td>Human, mouse, rat, frog, worm</td>
<td>User-supplied miRNA sequence and UTR sequence for command line</td>
</tr>
<tr>
<td>RNAlhybrid</td>
<td>energy-based</td>
<td>Any</td>
<td>User-supplied data for miRNA sequence and mRNA sequence</td>
</tr>
<tr>
<td>PITA</td>
<td>energy-based</td>
<td>Human, mouse, fly, and worm</td>
<td>MiRNA and gene names or NCBI RefSeq mRNA accession numbers.</td>
</tr>
<tr>
<td>RNA22-GUI</td>
<td>energy-based</td>
<td>Human, mouse, fly, and worm</td>
<td>MiRNA name, Ensembl gene ID, Ensembl transcript ID, or gene name</td>
</tr>
<tr>
<td>MirTarget2</td>
<td>machine learning-based</td>
<td>Human, mouse, rat, dog and chicken</td>
<td>MiRNA name, gene name, National Center for Biotechnology Information (NCBI) RefSeq mRNA accession number, gene ID or GenBank accession number</td>
</tr>
<tr>
<td>DIANA microT-CDS</td>
<td>machine learning-based</td>
<td>Human, mouse, fly, and worm</td>
<td>MiRNA name, gene name, Ensembl ID, KEGG description</td>
</tr>
</tbody>
</table>

TargetScan was first proposed in 2003 as a target prediction tool that mainly takes the complementarity of the seed region into account [49]. This is one of the most used algorithms for miRNA target prediction and it has been constantly updated [56]. TargetScan is considered an easy to use tool since it does not need the adjustment of advanced settings, however, this can also be considered a disadvantage for advanced users. TargetScan is usually used along with another target prediction algorithm, miRanda [50]. By contrast with TargetScan, miRanda considers matching throughout the entire miRNA sequence, although it takes into greater account the matches in the seed region. Just like
TargetScan, miRanda is regularly updated and it has been considered easy to compile and run. Still, it provides a reasonable number of adjustable parameters that may be useful for investigating specific miRNA targets, which can turn out to be an obstacle for a less advanced user [50], [57].

Energy-based algorithms include RNAhybrid, PITA and RNA22-GUI. Although they are all put together in the same group of algorithms, RNAhybrid predicts potential binding sites of miRNAs by looking for the energetically most favorable hybridization sites between miRNAs and their target mRNAs [51], [58], while PITA uses target-site accessibility as major feature for performing miRNA target prediction [52], [59] and RNA22-GUI uses a pattern-based approach to find miRNA binding sites [60]. Both RNAhybrid and PITA allow the user to define the seed region, also giving the chance to the user to define more advanced settings, which makes these tools more appropriate for advanced users. Also regarding RNA22-GUI, this algorithm represents miRNA:mRNA target interactions in a cDNA map that could be considered difficult to navigate [53], [59].

Finally, the third group of target prediction algorithms include mirTarget2 and DIANA-microT-CDS. Both algorithms use a machine-learning approach to identify new features significantly correlated with miRNA:mRNA target interactions. MirTarget2 is a target prediction tool based on support vector machines (SVMs) that makes use of public microarray data while DIANA-microT-CDS uses photo activatable-ribonucleoside-enhanced crosslinking and immunoprecipitation (PAR-CLIP) data [54], [55]. MirTarget2 gives special attention to features such as seed conservation, seed match, base composition in the regions flanking the seed pairing sites, secondary structure, and location of the site within the 3’ UTR [54]. DIANA-microT-CDS gives more importance to features such as: distance to the nearest end of the region or to an adjacent binding site, the predicted free energy of the duplex and AU content. It is a considered user-friendly web interface that also allows the download of a plugin that lets us use a non-web interface with additional options.

Although there are several tools for predicting miRNAs targets, it is hard to consider that one of them is able to detect all the exact targets of a specific miRNA, since they use different algorithms that consider various features. For instance, a study from 2010 compared miRanda, TargetScan and RNAhybrid, as they are three of the most frequently used target prediction algorithms. The results showed that, when tested individually, each one of the algorithms is better in a specific feature, while, when using integration strategies, the results revealed that the application of a probabilistic model for classification (Bayesian Network) on the features calculated from multiple prediction methods significantly improved target prediction accuracy [61]. However, such combinations of predictions have also been shown to perform worse than the prediction by one single algorithm, possibly due to the trade-off between specificity and sensitivity [62].

From the 8 tools that were compared, only 2 perform target prediction: CPSS and DeAnnIso. CPSS allows performing target prediction using a great set of target prediction tools, including RNAhybrid, miRanda and TargetScan, however, the authors of CPSS recommend that one should keep in mind that the more tools are used simultaneously, less results are obtained, since the result corresponds to the overlap of all predictions. When using CPSS, target prediction for a single sample is performed for the ten most abundant novel miRNAs and known miRNAs, while all the differentially expressed miRNAs will be subjected to target prediction for paired samples allowing the selection between eight different target predictors. The output corresponds to the overlap of targets predicted by the selected algorithms. DeAnnIso allows the user to select isomiRs of their interest (range from 2 to 8) for target analysis directly. For predicting the targets of each isomiR, this tool uses one of two possible target prediction algorithms: miRanda or RNA hybrid.
1.2.2.2. **Gene-Enrichment analysis**

After the identification of isomiR targets, the following step to obtain insights regarding the impact of isomiR expression is to infer, from the obtained gene list, which biological processes are being affected. Performing gene set enrichment analysis and functional enrichment analysis is a commonly used strategy that allows identifying, in a gene list, classes of genes or proteins that are over-represented through a statistical approach. A study from 2008 did an extensive analysis and comparison of a total of 68 tools that perform functional enrichment analysis and classified them in three classes: singular enrichment analysis (SEA); gene set enrichment analysis (GSEA); and modular enrichment analysis (MEA). In SEA-based approach, annotation terms of a subset of genes are assessed one at a time against a list of background genes. GSEA approaches are similar, but consider all genes during the enrichment analysis, instead of a pre-defined threshold based genes as in the SEA approach. Finally, tools based on MEA are considered to inherit the basic enrichment calculation found in the SEA and to also incorporate extra network discovery algorithms by considering the term-to-term relationships, being considered the type of analysis that is closer to the nature of biological data structure, obtaining results with more sensitivity and specificity [63].

Using an R package for this step of the analysis allows to easily integrate this analysis into a bioinformatics pipeline. Two of the most known R packages available to perform gene set enrichment analysis include topGO and GOstats. TopGO algorithm was first described in 2006 [64] and has been constantly updated [65]. This package is designed to facilitate semi-automated enrichment analysis for GO terms and performs the analysis using two different statistical tests, Fisher’s exact test which is based on gene counts, and a Kolmogorov-Smirnov like test which computes enrichment based on gene scores. GOstats, on the other hand, applies a Hypergeometric calculation for over or under-representation of GO terms in a specified gene list, being constantly updated as well [66]. Although both packages are extremely used, GOstats is inserted in the first classification of enrichment tools (SEA) and topGO is classified into a modular enrichment tool (MEA), taking into account network interactions allowing the identification of more specific term in the GO ontology hierarchy (low level terms).

Only two of the tools described in Table 1.1 perform pathway enrichment analysis. Both CPSS and DeAnnIso include the GO analysis in their pipelines. CPSS obtains the enriched biological processes using a Hypergeometric test, where a GO term will be identified as key term and subjected to continue analysis when its ratio of enrichment is >2 and p-value is <0.05. DeAnnIso is also able to perform enrichment analysis. This step is carried out for isomiR-affected targets and the enrichment p-values is quantitatively measured by Fisher’s exact test.

1.2.2.3. **Prediction of novel miRNAs**

Besides being able to detect the several types of different non-coding RNAs, it is also important to evaluate if there are potential novel miRNAs present in small-RNA-seq datasets. After a brief analysis of the previous performed studies concerning prediction of novel miRNAs in small-RNA-seq datasets it was possible to determine that one of the most used algorithm in this field is miRDeep [67]–[69]. miRDeep was first established in 2008 as an algorithm that uses a probabilistic model of miRNA biogenesis to score
compatibility of the position and frequency of sequenced RNA with the secondary structure of the miRNA precursor. This algorithm starts by filtering all the candidates whose structure and read signature are inconsistent with Drosha/Dicer processing, followed by the use of Bayesian statistics to score the fit of the structure and signature to an explicit model of miRNA biogenesis. Also, it includes strict statistical controls to estimate the false positive rate and the sensitivity of miRDeep predictions, which makes it possible for the users to estimate the quality of their results [67].

Since it is a very used tool, miRDeep has been updated and improved throughout the years. For instance, MiRDeep2, an updated version of miRDeep, can identify miRNA candidates with improved confidence, producing stronger results, with a notable upgrading in accuracy in the identification of non-canonical miRNAs like anti-sense miRNAs [70], [71]. For a single analysis with miRDeep2, the tool produces a single .html page that links to all results generated by the module. This file includes a survey of miRDeep2 performance for varying score cut-offs, providing estimates of sensitivity and number of true positive novel miRNAs. Also, it includes a table of novel miRNAs discovered in the sequencing data [71].

Regarding the tools in comparison (Table 1.1), only CPSS and miRanalyzer are able to perform novel miRNA prediction. For the sequences that could be mapped to the reference genome but could not be classified, CPSS allows the user to select two different tools to perform novel miRNA prediction: miRDeep and mireap, a tool that has no documentation but can be accessed at http://sourceforge.net/projects/mireap/. The other tool capable of obtaining predictions of novel miRNAs, miRanalyzer, doesn’t make use of an external tool for this purpose, since it integrates a specifically developed machine learning algorithm for the prediction of novel microRNAs.

1.3. MiRNAs and isomiRs as biomarkers

From the time when miRNAs were shown to be present in human body fluids, they have arisen as being potential non-invasive biomarkers [72]. This is due to the fact that, besides appearing dysregulated in several pathological conditions [72], they present high stability in fluids such as plasma, serum, saliva and urine [73]. The altered expression of miRNAs in different types of cancer have been extensively studied in the last few years. For instance, the let-7a miRNA has been reported as being differently expressed in lung cancer [74], as let-7f in ovarian cancer [75] and miR-18a in colorectal cancer [76]. Besides the severe investigation that has been made in the field of cancer, other studies suggest that miRNAs are also dysregulated in viral infections [77], cardiovascular diseases [78] and other disorders such as the ones of the nervous system [79]. This leads to the conclusion that using these differently expressed miRNAs as biomarkers for diseases might be considered a valuable diagnostic strategy. Furthermore, due to this rising interest in miRNAs as potential biomarkers, a database (miRandola) that presents a comprehensive list of all known extracellular ncRNAs (including 1002 miRNAs), was recently updated, gathering information from a total of 314 papers [80].

A recently reported investigation included an examination of serum miRNAs of patients with liver pathogenesis and showed that, in fact, a total of 179 miRNAs exhibited altered serum levels across the diseased subjects. Additionally, and in line with the present work, they also identified a high percentage of isomiRs, which indicates that the cellular response to stress or disease might be more complex than we actually consider [81].
2. Goals

This thesis aimed to develop a pipeline for the analysis of deep sequencing data, being able to detect the different types of small non-coding RNAs present in a sample (tRNA, miRNA, snoRNA, snRNA, siRNAs) while being able to simultaneously identify all possible types of isomiRs present in small-RNA-seq data (including combinations of isomiR type in a same molecule). The developed pipeline should: (i) receive multiple datasets from two different experimental conditions, (ii) perform a general characterization of small-RNA-seq datasets, (iii) identify all existing miRNA variants (isomiRs), (iv) predict novel miRNAs and identify their variants, (v) analyze the differential expression of isomiRs between samples, (vi) perform isomiR target prediction, (vii) identify the pathways in which the isomiR targets are involved. The last stage of this project included benchmarking the pipeline by performing analysis of small-RNA-seq datasets publicly available.

3. Materials and methods

The pipeline to identify all possible types of isomiRs was written in Perl (v5.10.1) and for the remaining features the pipeline incorporates third-party software.

3.1. Third-party software integration

The pipeline integrates more than 10 tools and utilities that are available for freely usage and have been well established in the field (Table 3.1). Bowtie was selected as the software for performing sequence alignment [82]. To determine the presence and frequency of all types of sncRNAs (excluding miRNAs), htseq-counts function of HTseq Python modules [83] was selected. To perform variant calling, two utilities were used in combination - SAMtools mpileup [84] and BCFtools [85]. SAMtools provides various utilities for manipulating alignment SAM files. SAMtools was used to identify single nucleotide variants which are filtered by using BCFtools in order to remove false positives.

To further annotate the previously identified variants according to dbSNP records, GATK VariantAnnotator was used [86], and its dependencies had also to be installed (Java and PicardTools - http://broadinstitute.github.io/picard/). For novel miRNA prediction, MiRDeep2 was used [71]. For performing differential gene expression analysis we used DESeq2 R package, which is able to estimate variance-mean dependence in count data from deep sequencing datasets and test for differential expression based on a model using the negative binomial distribution [87]. For performing target prediction, two different tools were included in the pipeline: miRanda [88] and TargetScan [89]. Finally, the TopGO R package was used to accomplish Gene set enrichment analysis [64].
Table 3.1. External software included in the pipeline.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bowtie</td>
<td>1.2.1.1.</td>
</tr>
<tr>
<td>HTSeq</td>
<td>0.6.1p1</td>
</tr>
<tr>
<td>SAMtools</td>
<td>1.2</td>
</tr>
<tr>
<td>Java</td>
<td>1.8.0_131</td>
</tr>
<tr>
<td>PicardTools</td>
<td>2.9.4</td>
</tr>
<tr>
<td>BCFtools</td>
<td>1.3.1</td>
</tr>
<tr>
<td>GATK</td>
<td>3.7-0</td>
</tr>
<tr>
<td>miRDeep2</td>
<td>2.0.0.8</td>
</tr>
<tr>
<td>DeSeq2</td>
<td>1.14.1</td>
</tr>
<tr>
<td>miRanda</td>
<td>3.3a</td>
</tr>
<tr>
<td>TargetScan</td>
<td>7.0</td>
</tr>
<tr>
<td>topGO</td>
<td>2.26.0</td>
</tr>
</tbody>
</table>

3.2. Databases

The pipeline was prepared to perform analysis for a total of 13 species: *Homo sapiens*, *Mus musculus*, *Drosophila melanogaster*, *Sus scrofa*, *Bos Taurus*, *Gallus gallus*, *Capra hircus*, *Equus caballus*, *Ovis aries*, *Pan troglodytes*, *Rattus norvegicus*, *Canis familiaris* and *Danio rerio*. Since many databases such as RNACentral, have not yet available the genomic information for the referred species, it is only possible to perform the integrated analysis for *Homo sapiens*.

The *Homo sapiens* databases included: a) genome assembly hg18, NCBI 36.1 ([http://www.ensembl.org/info/data/ftp/index.html](http://www.ensembl.org/info/data/ftp/index.html)); b) genomic coordinates of small non-coding RNAs RNACentral in the GFF file format ([ftp://ftp.ebi.ac.uk/pub/databases/RNAcentral](ftp://ftp.ebi.ac.uk/pub/databases/RNAcentral)) which include tRNA, miRNA, snoRNA, snRNA, siRNAs; c) mature microRNA and pre-miRNA sequences from mirBase version 21.0 ([ftp://mirbase.org/pub/mirbase/CURRENT/genomes/](ftp://mirbase.org/pub/mirbase/CURRENT/genomes/)); d) Single Nucleotide Polymorphism database (dbSNP) in VCF format ([ftp://ftp.ncbi.nih.gov.snp/organisms/](ftp://ftp.ncbi.nih.gov.snp/organisms/)), from which the file containing the most common SNPs in humans was downloaded; d) 3’ UTR sequences, were obtained from TargetScan (version 7.1) site ([http://www.targetscan.org/cgi-bin/targetscan/data_download.vert71.cgi](http://www.targetscan.org/cgi-bin/targetscan/data_download.vert71.cgi)) and genomic coordinates obtained from RNA central and from miRBase 21 were developed based in genome assembly hg18. PiRNAs were not included due to lack of genomic coordinates for this species of sncRNAs.
3.3. **Workflow benchmarking**

From the Gene Expression Omnibus database we obtained the datasets with the accessions GSE90028 or GSE59565 [81]. These data were generated in a study aiming to identify miRNAs as biomarkers of the occurrence of different types of liver impairment. As a result the authors identified signatures of circulating miRNAs specific to different liver injury phenotypes. A complexity of isomiR molecules was reported, although the details regarding the identified isomiRs along with their biological impact were not. The analysis of a subset of samples derived from this study was performed in order to identify differential isomiR expression of hepatitis B infected patients in comparison with healthy individuals. For this analysis we selected 3 samples from healthy controls (GSM2395967, GSM2395968 and GSM2395969) and 3 samples from patients with hepatitis B (GSM2395985, GSM2395989 and GSM2395990). The specified files were downloaded using the fastq-dump functionality of the SRA Toolkit (https://trace.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?cmd=show&f=software&m=software&s=software). Dataset quality was inspected using FASTQC (https://www.bioinformatics.babraham.ac.uk/projects/fastqc/). After the download using fastq-dump, the sequence adapters were identified by running a short script, and the data was pre-processed to clip the identified adapters using fastx_clipper from the FASTX-Toolkit (http://hannonlab.cshl.edu/fastx_toolkit/), using the following command: `bin/fastx_clipper -a TGGAATTCTCGGGT -Q33 -i <input file> -o <output file>`. Quality selection was performed using a short PERL script in order to remove sequences holding homopolymers and with an average phred quality score lower than 20.
4. Results and Discussion

We present here a novel analysis pipeline developed in PERL programming language that enables the identification of snRNA and of all types of isomiRs, furthermore including analysis of functional impact of isomiR expression into the cell genetic program. The pipeline consists of a multiplatform collection of automated and parallelized scripts to analyze small-RNA-seq data (isomirwindow.campus.ciencias.ulisboa.pt). Importantly, the pipeline combines a set of well-established external software with a newly developed algorithm for the identification of all types of isomiRs. It receives FASTQ files as input without restriction of file size, performs the alignment, identifies the species of small non-coding RNAs present in a data file, identifies all types of isomiRs, reports summary statistics for read annotation and identifies differently expressed miRNAs/isomiRs. Furthermore, it allows performing functional analysis for the differently expressed isomiRs and reports regarding their functional impact. The pipeline is currently integrated in a graphical user interface (GUI) tool called IsomiR Window (not in the scope of this thesis), which allows generating, from inputs created by the pipeline, different types of visualization of results (Figure 4.1) [90].

![IsomiR Window tool]

*Figure 4.1. Schematic representation of the IsomiR Window tool. The tool integrates the developed pipeline with a GUI that was not developed in the scope of this thesis [90].*

4.1. Workflow implementation

The workflow of the developed pipeline (Figure 4.2) is divided into two distinct analytical branches that may be executed independently to analyze distinct small-RNA-seq data: annotation and functional analysis. The parameters have been selected based on third software developer's recommendations and based in our daily experience while performing routine analysis of this type of data. The pipeline fully automates small-RNA-seq analysis, reducing human errors and improving the reproducibility of small-RNA-seq studies.
Figure 4.2. Flow chart of analysis modules. Flow chart for (A) Annotation analysis with or without novel miRNA prediction and for (B) Functional analysis module. For each module input, internal workflow and output is shown. Input files are represented by a parallelogram, processes are represented rectangles. Files that are derived from the novel algorithm for isomiR detection are represented by a rectangle with a curved bottom. Dotted lines indicate optional steps. The individual description of all steps is explained below.
4.1.1. Annotation analysis

4.1.1.1. Alignment and sncRNAs classification

The first step of the annotation analysis module (Figure 4.3) includes mapping the reads provided by the FASTQ input files, followed by the detection of small non-coding RNAs, other than miRNAs, that are present in these files. Both tasks are achieved when running the first script of the pipeline – `find_ncRNAs.pl` (Figure 4.4). In order for the script to be executed without errors, the script requires the following inputs, name of input file, species identifier, number of mismatches (1-3) and the number of multiple hits (1-3) allowed for read mapping in the corresponding species genome (Appendices 7.1 and 7.2). Only a setting of number of mismatches (2-3) will allow the identification of isomiRs with 3’ end tailing downstream in the workflow. The script is prepared to accept the identifiers and corresponding genomes shown in Table 4.1.

<table>
<thead>
<tr>
<th>Species</th>
<th>Species identifier</th>
<th>Genome assembly version</th>
<th>miRBase version</th>
<th>RNACentral version</th>
<th>3’ UTRs fasta file</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homo sapiens</td>
<td>homoSapiens</td>
<td>GRCh38</td>
<td>GRCh38</td>
<td>GRCh38</td>
<td>Available</td>
</tr>
<tr>
<td>Mus musculus</td>
<td>musMusculus</td>
<td>GRCm38</td>
<td>GRCm38</td>
<td>GRCm38</td>
<td>Available</td>
</tr>
<tr>
<td>Drosophila melanogaster</td>
<td>drosophilaMelanogaster</td>
<td>BDGP6</td>
<td>BDGP5.0</td>
<td>BDGP6</td>
<td>Available</td>
</tr>
<tr>
<td>Sus scrofa</td>
<td>susScrofa</td>
<td>Sscrofa10.2</td>
<td>Sscrofa10.2</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Bos taurus</td>
<td>bosTaurus</td>
<td>UMD3.1</td>
<td>UMD3.1</td>
<td>UMD3.1</td>
<td>NA</td>
</tr>
<tr>
<td>Gallus gallus</td>
<td>gallusGallus</td>
<td>Gallus-gallus-4.0</td>
<td>Gallus-gallus-4.0</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Capra hircus</td>
<td>capraHircus</td>
<td>NA</td>
<td>CHIR_1.0</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Equus caballus</td>
<td>equusCaballus</td>
<td>EquCab2</td>
<td>EquCab2.0</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Ovis aries</td>
<td>ovisAries</td>
<td>Oar_v3.1</td>
<td>OARv3.1</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Canis familiaris</td>
<td>canisFamiliaris</td>
<td>CanFam3.1</td>
<td>CanFam3.1</td>
<td>CanFam3.1</td>
<td>NA</td>
</tr>
<tr>
<td>Danio rerio</td>
<td>danioRerio</td>
<td>GRCz10</td>
<td>Zv9</td>
<td>GRCz10</td>
<td>NA</td>
</tr>
<tr>
<td>Pan troglodytes</td>
<td>panTroglodytes</td>
<td>CHIMP2.1.4</td>
<td>PanTro2.1.4</td>
<td>CHIMP2.1.4</td>
<td>NA</td>
</tr>
<tr>
<td>Rattus norvegicus</td>
<td>rattusNorvegicus</td>
<td>Rnor_6.0</td>
<td>Rnor_5.0</td>
<td>Rnor_6.0</td>
<td>NA</td>
</tr>
</tbody>
</table>
The script starts by checking input files format, by verifying if the number of lines is a multiple of 4. Secondly, reads are mapped using Bowtie [82]. After running Bowtie, the script performs a checkpoint to evaluate if the read mapping step was correctly finished in order to proceed to the next step. The checkpoint consists in verifying if the number of unique identifiers of a sequence matches in both input file (FASTQ) and alignment file (SAM).

After the production of a complete SAM alignment file, the script proceeds to the identification and quantification of all types of small non-coding RNAs, not including miRNAs. By comparison of genomic coordinates of mapped reads and of sncRNAs annotated in the RNACentral database [91], the script invokes the htseq-count function allowing to identify and quantify the following types of sncRNAs: RNase_P_RNA, guide_RNA, antisense_RNA, lncRNA, snRNA, RNase_MRP_RNA, rRNA, tRNA, scRNA, ribozyme, SRP_RNA, telomerase_RNA, other, snoRNA, vault_RNA and misc_RNA. PiRNAs have not been considered since genomic coordinates are not yet available for this type of sncRNAs. A file with unassigned reads is produced in SAM format (Filtered_SAM_*.sam) and further used as input for the next step of the pipeline.

4.1.1.2. Prediction of novel miRNAs

The prediction of novel miRNAs is optional, due to the fact that the selected algorithm, miRDDeep2, is time consuming. This step is performed by run_miRDeep2.pl script, which invokes miRDDeep2. The script run_miRDeep2.pl was developed following the implementation advised by the authors that developed miRDDeep2 [71]. Running the core miRDDeep2 module involves the previous creation of a few input files, which are all automatically produced within the run_miRDee2.pl script. First, the script produces a merged SAM file, including all the alignment files previously created for each sample (All_Filtered_mirdeep_*.sam). Additionally, to invoke miRDDeep2, it is also required a collapsed FASTA file and an ARF file, which are produced with the following names: reads_collapsed_*.fa and reads_collapsed_vs_genome_*.arf, respectively. Finally, the script also needs to have access to a FASTA file of the corresponding genome, a FASTA file with known miRNAs of the analyzing species and a FASTA file of known miRNAs of related species, which, when considering an analysis for humans, we used the following species: *Pan troglodytes*, *Pan paniscus*, *Gorilla gorilla* and *Pongo pygmaeu*.

MiRDDeep2 output includes a CSV and an html file, both including the same information: an overview of all detected miRNAs in the deep sequencing input data. Each line includes several information regarding a single novel miRNA candidate: the miRDDeep2 score, the probability that the miRNA candidate is genuine given the evidence from the sequencing data, sequence and read count summaries, a link to a graphic representation of structure and read signature, a link to the UCSC genome browser for the species analyzed, and a link to NCBI blast results for the candidate precursor sequence.

An additional script, add_novel_miRNAs.pl, was developed with the purpose of facilitating the inclusion of the novel predicted miRNAs, by miRDDeep2, to the miRBase file. This script is not included in the pipeline, however, it gives the possibility to detect variations on the novel predicted miRNAs. For more information regarding the usage of this script see Appendix 7.2.
4.1.1.3. **IsomiRs identification and classification**

The third script of the pipeline – `find_isomiRs.pl` (Figure 4.4) has the main purpose of assigning reads to miRNAs by comparison of genomic coordinates between reads and the species corresponding collection of pre-miRNAs and derived canonical miRNAs using the file `Filtered_SAM_*.sam` as input (Appendices 7.1 and 7.2). The algorithm, developed in PERL, is able to detect and quantify all types of isomiRs, templated and not templated and all possible combinations between these categories. The algorithm starts by comparing the genomic coordinates of mapped reads and annotated pre-miRNAs. If a read is within the pre-miRNA interval, without redundancy it will proceed to the next level of analysis. If a read falls into the genomic interval of more than one pre-miRNA molecule, the read is classified as ambiguous and it is written into a file (`Filtered_SAM_*sam_ambiguous.txt`). Reads within a pre-miRNA genomic location and classified as non-ambiguous proceed to the isomiR classification stage.

The isomiR classification starts by taking into account if the sequence was aligned with 0, 1, 2 or 3 mismatches. If the sequence aligns with 0 mismatches, the tool estimates the shifts of the start and end locations of the read in relation to the genomic location of the canonical miRNA (first testing from which arm - 5p or 3p - of the hairpin the read derives) in order to classify the read as 5’ or 3’ end additions or trimming, or even all possible combinations of these templated isomiRs. If the sequence was mapped with 1 mismatch, the algorithm uses SAMtools mpileup function to detect single nucleotide polymorphisms (SNPs) followed by BCFtools to select the highly significant identified SNPs. The settings defined within the script are shown in Table 4.2.
Figure 4.3. Detailed diagram of the annotation analysis module. From left to right, the figure depicts the sequence of the scripts that compose the annotation analysis module. The arrows indicate actions of processing that each script performs.
Table 4.2. Settings defined for variant calling using SAMtools mpileup and BCFtools.

<table>
<thead>
<tr>
<th><strong>SAM mpileup settings</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The minimum number of gapped reads for an indel candidates (-m) was set at 1;</td>
<td></td>
</tr>
<tr>
<td>The minimum mapping quality for an alignment to be used (-q) was set at 20;</td>
<td></td>
</tr>
<tr>
<td>The minimum base quality for a base to be considered (-Q) was set at 20.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>BCFtools settings</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>The SNP/indel calling flag (call);</td>
<td></td>
</tr>
<tr>
<td>A flag to report only SNPs (--skip-variants indels);</td>
<td></td>
</tr>
<tr>
<td>A flag to call only multi-allelic variants (--multiallelic-caller);</td>
<td></td>
</tr>
<tr>
<td>A flag to report only multi-allelic variants and exclude sites without alternate alleles (--variants-only)</td>
<td></td>
</tr>
<tr>
<td>An additional filter is piped to the command by using a functionality of a perl script (vcfutils.pl varFilter) establishing the minimum number of alternate bases (-a3) and the minimum read depth (-d10).</td>
<td></td>
</tr>
</tbody>
</table>

The identified SNPs are then annotated by comparison with the species correspondent dbSNP database using GATK VariantAnnotator. As a result of this comparison, the algorithm classifies significant variations existent in dbSNP as SNPs (-SNP) and the remaining ones as internal editing events (-Ed). Finally, if a nucleotide change is observed in the file (Filtered_SAM_*.sam) with a frequency above 100 but it was not classified as significant by SAMtools, it is still considered to be relevant, being reported with the code -Alt (alteration).

At this stage, the algorithm is able to produce the annotation of templated and of non-templated isomiRs with internal editings and all possible combinations, leaving the identification of non-templated isomiRs with 3’ end tailing for the next step. For this purpose, the algorithm checks if the mismatches are in the 3’end and then it searches for the repetition of the same nucleotide at the same position. The algorithm generates for each miRNA/isomiR an identifier that allows an intuitive identification of all the modifications that were identified in the miRNA (Figure 4.3). The final output with this information is a file called Filtered_SAM_*.sam_not_ambiguous.txt, which is a tab delimited file that includes a column with the identifier of a miRNA/isomiR followed by a second column with its frequency in the respective sample.
Figure 4.4. The isomiR identifier. The figure depicts the identifier generated by the find_isomiR algorithm for a canonical miRNAs. This identifier was developed in order to allow different queries and to generate summary statistics. The identifier should be read from left to right and displays the transcript sequence, the identifier of the canonical miRNA from which it derives, followed by a code that distinguishes the different types of frame-shifts encountered in comparison with the respective canonical miRNAs.

In the first column we can find the sequence identifier which starts by displaying the sequence corresponding to the transcript (in sequences deriving from pre-miRNAs transcribed from the minus strand, the mapper algorithm maps and reports the sequence in reverse complement, the developed algorithm takes this into consideration for proper estimation of frameshifts and for developing a user-friendly isomiR identifier). Then the identifier is followed by a code that quantifies the nucleotide shifts observed at 3’ and 5’ end respectively. As shown in Table 4.2, canonical miRNAs will be classified as “0_0_0_0” in this part of the identifier. Then the identifier allows understanding if the nucleotide changes are observed, specifying the type of change followed by the classification of this change performed as previously described into editings (Ed), SNPs or nucleotide alterations (Alt), and finally specifying the position in the read where this nucleotide change is observed. A canonical miRNA displays the following identifier in this section, “NA_0_NA”. The final section of the identifier aims to provide information regarding 3’end tailings. Reports if a tailing is observed (using TRUE or FALSE annotation and the type of tailing observed). A canonical miRNA, as shown in Figure 4.4 and Table 4.3, should display “F_NA” in this section of the identifier. The different items of the identifier are separated by “_” which allows easily querying and producing summary statistics using bash scripting.
### Table 4.3. Examples of isomiRs identifiers.

<table>
<thead>
<tr>
<th>IsomiR ID</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>UGGGAAAGAGAAAGAACAAGU_hsa-miR-6739-5p_0_0_0_0_NA_0_NA_F_NA</td>
<td>hsa-miR-6739-5p with no variations (canonical form)</td>
</tr>
<tr>
<td>CUGGACACCAGCUCAGCCGGCGCCG_hsa-miR-4638-3p_0_1_1_0_NA_0_NA_F_NA</td>
<td>Trimming of 1 nucleotide at the 3’ end</td>
</tr>
<tr>
<td>AAAUUAAGUAUAUCCAGGAUAGG_hsa-miR-26a-5p_3_0_0_2_NA_0_NA_TUUU</td>
<td>Addition of 3 nucleotides at the 3’ end</td>
</tr>
<tr>
<td>UACCACAGGUAGAACCACCGAAA_hsa-miR-140-3p_3_0_0_0_C/A_23_Alt_F_NA</td>
<td>Addition of 3 nucleotides at the 3’ end</td>
</tr>
</tbody>
</table>

#### 4.1.1.4. Differential expression

The pipeline further investigates the existence of differential expression of isomiRs between two experimental conditions (Figure 4.3) – `deseq.pl + Rdeseq.R` (Appendices 7.1 and 7.2). Testing for differential expression requires two input files that are produced within the deseq.pl script: a file containing all the frequency of each isomiR/miRNA across all the samples for each condition (*Deseq_IsomiR_*.txt), and a file specifying the experimental design (paired or not paired samples) to take into further consideration - *ExperimentalDesign_*.txt. The first mentioned file includes an initial column with the identifier of the miRNA/isomiR, followed by several columns including the frequency of this miRNA/isomiR across all the samples for each condition, the condition 1 followed by condition 2. Thus, the total number of columns varies according to the number of files for each condition. The file containing information regarding the experimental design is created accordingly to the type of analysis given as input (“Unpaired_Sample” or “Paired_Sample”) to the deseq.pl script. This file has a specific format that DESq2 needs as input. An example of the file produced for unpaired samples analysis is showed in Figure 4.5. If “Paired_Sample” is selected instead, it adds a column specifying from each individual the sample derives.
Figure 4.5. Experimental design file for unpaired samples analysis. This file is an example of the produced ExperimentalDesign_*.txt file when invoking the DESeq2.pl script. The first column includes an identifier for each sample, the second column indicates the name of the present file, and the third column indicates from which clinical condition the sample derives.

Additionally, it is possible to perform the analysis using two different levels of statistical significance, choosing between adjusted p-value <0.05 or <0.1. Furthermore, the pipeline uses functions from DESeq2 R package [87] to test for differential expression for the specified experimental design defined while invoking the deseq.pl script (see above). The pipeline starts by performing a minimal pre-filtering to remove rows that display a sum of read count smaller or equal than 1. Then the differential expression test is applied, with the previous selected level of statistical significance, adjusted p-value < 0.05 or adjusted p-value < 0.1 (also defined by the user when invoking the script deseq.pl). The direct output from DESeq2 includes a table (CompleteTableDE_*.txt) showing the expression results for all miRNAs/isomiRs included in the input file (Table 4.4).

Table 4.4. Example of DESeq2 output file. The table shows the first rows of the CompleteTableDE_*.txt file produced by DESeq2. The baseMean column indicates the average of the normalized counts taken over all samples, the second column indicates the log2 fold change between the groups (a value of 2 means the expression has increased 4-fold change), followed by the column lfcSE, which indicates the standard error of the log2FoldChange. The stat column shows the Wald statistic estimation and the final two columns show the obtained p-value and adjusted p-value.

<table>
<thead>
<tr>
<th>baseMean</th>
<th>Log2FoldChange</th>
<th>lfcSE</th>
<th>stat</th>
<th>pvalue</th>
<th>padj</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUGACCUAUGAAUUGACA GAC_hsa-miR-215-5p_0_0_0_0_NA_0_NA_F_NA</td>
<td>22.7513746</td>
<td>-3.97575998256246</td>
<td>0.78260243</td>
<td>3.77080605</td>
<td>8.9745184224</td>
</tr>
<tr>
<td></td>
<td>150466</td>
<td>150466</td>
<td>150466</td>
<td>150466</td>
<td>150466</td>
</tr>
<tr>
<td>AAUGCGCACCACUGGGU G_hsa-miR-652-3p_0_2_0_0_NA_0_NA_F_NA</td>
<td>267.059606</td>
<td>1.89794642465596</td>
<td>0.37327767</td>
<td>3.68512206</td>
<td>8.9745184224</td>
</tr>
<tr>
<td></td>
<td>35603</td>
<td>35603</td>
<td>35603</td>
<td>35603</td>
<td>35603</td>
</tr>
<tr>
<td>UUAUAAAGCAAUGAGACU GAUU_hsa-miR-340-5p_0_0_0_0_NA_0_NA_F_NA</td>
<td>152.927684</td>
<td>-1.71921619172834</td>
<td>0.41360565</td>
<td>3.22940519</td>
<td>0.0051239895</td>
</tr>
<tr>
<td></td>
<td>097944</td>
<td>097944</td>
<td>097944</td>
<td>097944</td>
<td>097944</td>
</tr>
</tbody>
</table>
4.1.2. Functional analysis

The second module of the analysis (Figure 4.6) corresponds to the functional analysis module. For performing functional analysis, the script `functional.pl` needs to be executed (Appendices 7.1 and 7.2). This script takes as input the sequence information of isomiRs to be studied in FASTA format. When using this pipeline as a standalone tool, the user needs to create this input before using the functional.pl script. Therefore, the input FASTA file should include all the identifiers and corresponding sequences of the miRNAs/isomiRs that were considered as relevant to proceed to the functional analysis module. At this stage, the pipeline allows the usage of two different target predictors – miRanda and TargetScan. Target prediction may be performed selecting both or just one of them. Once again, when using the tool outside the graphical interface the user needs to provide this selection as arguments of the functional.pl script (for more detailed information on this regard see Appendix 7.1 and Appendix 7.2). However, bear in mind that this pipeline was developed with the purpose of being integrated in a GUI, and that within the IsomiR Window tool the user will be able to just select the desired isomiRs through this interface, without the need of preparing the input FASTA file, which is produced in an automated manner.

Following the establishment of all the necessary parameters, the script starts by performing target prediction using the selected target prediction algorithm. The execution of miRanda, produces a file with the target site alignments (Figure 4.7), which is processed by the script, first to select only the predictions with a total score above 150 and a total energy less than -7, producing a final gene list. The execution of TargetScan produces a tabular output showing the predicted genes (Figure 4.8), from which the script extracts the predicted gene list. If both target predictors are selected the scripts produces a gene list that results from the overlap of the two obtained gene lists. This final gene list is then used as input to the gene set enrichment analysis that is performed by the topGO R package [64]. For this purpose, the functional_isomiR.pl script invokes a short R script (`R_topGO.R`) that is ready to receive the list of unique gene IDs (one gene ID per row in HUGO format) as input and produces the statistical analysis of functional enrichment. The output produced is a table with the enriched GO terms along with the corresponding statistical significance.
For the functional analysis module, the functional.pl + R_topGO.R scripts produce the output files for the used target predictors (TargetScan or/and miRanda). The script processes the obtained outputs, a gene list that is used as input to topGO is accessible with the name: topgo_input*.txt. Furthermore, when performing gene enrichment analysis, topGO produces a table including the most significant GO terms for the input dataset (TableTopGO_*.txt).

Figure 4.6. Detailed diagram of the functional analysis module. From left to right, the figure depicts the functions programmed for the functional analysis module. The arrows indicate actions of processing of each step of the scripts flow.

Figure 4.7. Example of Miranda output file. The figure displays a sample of the output produced by the miRanda target predictor, showing a potential target site and the respective alignment scores based on sequence complementarity and thermodynamic stability.
4.1.3. Reporting results

Throughout the workflow, an extensive list of files is produced. However, in order to facilitate the analysis of the results of the annotation analysis module, an additional script was developed – **charts.pl**. This script processes output files produced by the annotation module and produces files with summary statistics in tab delimited format, which can be further used to produce graphical visualizations such as pie charts and bar charts (Table 4.5) using Excel. Furthermore, through the GUI developed in the scope of the IsomiR Window project (not in the present project), the visualizations are automatically created.

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read_size_barchart.txt</td>
<td>Contains the frequency for each size of the reads after performing alignment with Bowtie.</td>
</tr>
<tr>
<td>Total_noncoding.txt</td>
<td>Contains the frequency for each type of sncRNA in the dataset.</td>
</tr>
<tr>
<td>Counts_miRNAs.txt</td>
<td>Contains the frequency of each miRNA present in the dataset.</td>
</tr>
<tr>
<td>Counts_3p5p.txt</td>
<td>Contains the frequency of the miRNAs that derived from the 3p or 5p arm of the pre-miRNA.</td>
</tr>
<tr>
<td>Types_isomiRs.txt</td>
<td>Contains the frequency of all types of isomiRs.</td>
</tr>
<tr>
<td>Counts_tailings.txt</td>
<td>Contains the frequency for each type of tailing present in the sequences of the isomiRs.</td>
</tr>
<tr>
<td>Counts_editings</td>
<td>Contains the frequency for each type of editing present in the sequences of the isomiRs.</td>
</tr>
</tbody>
</table>
Furthermore, regarding differential expression analysis, invoking the deseq.pl + Rdeseq.R also produces an additional table, besides CompleteTableDE_*.txt (Appendices 7.1 and 7.2). More specifically, if the test finds that there are isomiRs/miRNAs differentially expressed between the samples, with an adjusted $p$-value lower than the selected value, an additional table is produced, including only these results (TableDE_*.txt). On the other hand, if no differentially expressed isomiRs/miRNAs are found, the script produces a file containing only the first 100 rows (with lower adjusted $p$-value) of the complete table (TableDE_*.txt). This is due to the fact that the CompleteTableDE_*.txt usually contains a huge number of entries, therefore facilitating visualizations with these smaller produced tables. Alongside these tables, the script also produces a heatmap (heatmap_*.png) using the gplots package in R [92], for further enhancing the visualization of differential expression across the samples.

4.1.4. Pipeline usage

All the developed scripts can be invoked by the command line and their respective arguments and input files are described in Appendix 7.2. However, the scripts were developed already aiming for the future integration in a graphical interface. For this reason, it can be a complex process to try to run this scripts outside the IsomiR Window tool, since they are designed to expect the specific structure of the Web application (Appendix 7.1). Additionally, since it is a wrapper tool, outside the IsomiR Window tool, the user would have to install all the tools and further dependencies that are used within this pipeline, which is also facilitated within the IsomiR Window tool. Nevertheless, it is possible to obtain more information regarding the folder structure in which this pipeline is based on by consulting the following project: “The IsomiR Window: the interface that bridges the complexity of miRNAs and their functional impact”, which includes the details of the developed GUI for this pipeline [90] and Appendices 7.1 and 7.2.
4.2. IsomiR Window pipeline analysis of small-RNA-seq data derived from plasma of hepatitis B patients

The developed pipeline was used to investigate the effect of infection in miRNA biogenesis, in publicly available datasets derived from plasma. Here we report the results obtained while comparing 3 sample controls (Condition 1 – C1) and 3 samples derived from individuals with hepatitis B (Condition2 – C2), datasets available through the public repository. The following table (Table 4.6) contains information regarding the number of sequences present in the FASTQ files, before and after quality filtering (see methods). The 6 filtered FASTQ files were used as input for the pipeline, and the entire analysis took about 2h45m.

<table>
<thead>
<tr>
<th>Sample</th>
<th>Identifier</th>
<th>Number of sequences</th>
<th>Number of sequences after quality filtering</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSM2395967</td>
<td>C1R1</td>
<td>6 179 023</td>
<td>6 108 474</td>
</tr>
<tr>
<td>(SRR5034632.fastq)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GSM2395968</td>
<td>C1R2</td>
<td>10 528 310</td>
<td>8 054 070</td>
</tr>
<tr>
<td>(SRR5034633.fastq)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GSM2395969</td>
<td>C1R3</td>
<td>4 445 188</td>
<td>4 400 342</td>
</tr>
<tr>
<td>(SRR5034634.fastq)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GSM2395985</td>
<td>C2R1</td>
<td>5 824 555</td>
<td>5 686 386</td>
</tr>
<tr>
<td>(SRR5034650.fastq)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GSM2395989</td>
<td>C2R2</td>
<td>10 176 643</td>
<td>10 112 274</td>
</tr>
<tr>
<td>(SRR5034654.fastq)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GSM2395990</td>
<td>C2R3</td>
<td>8 298 235</td>
<td>8 214 246</td>
</tr>
<tr>
<td>(SRR5034655.fastq)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The parameters selected for read mapping were: 3 allowed mismatches (-v3) and 3 multiple hits (-m3). An important mapping quality parameter is the percentage of mapped reads, which can be considered an indicator of the overall sequencing accuracy. For example, a recently published survey of best practices for RNA-seq analysis suggests that we should expect between 70% and 90% of regular RNA-seq reads to map onto the human genome (depending on the read mapper used) [46]. For the present analyzed samples, ~96% of the reads aligned to the human genome. However, when looking at the summary statistics reported by Bowtie (Table 4.7), it is possible to verify that the majority of the reads mapping was suppressed by the 3 allowed multiple hits. The same happened when performing the alignment with 5 multiple hits (results not shown), showing that, it is likely that a large proportion of the data corresponds to repeat-derived RNAs, or even piRNAs.
Table 4.7. Summary statistics provided by Bowtie. The table displays the percentage of reads that aligned to the reference genome, the percentage of reads that failed to align, the percentage of alignments suppressed by the number of genomic multiple hits (-m) and the final number of aligned reads that were reported to the output file.

<table>
<thead>
<tr>
<th>Reads processed</th>
<th>6108474</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reads with at least one reported alignment</td>
<td>2750180 (45.02%)</td>
</tr>
<tr>
<td>Reads that failed to align</td>
<td>223322 (3.66%)</td>
</tr>
<tr>
<td>Reads with alignments suppressed due to -m</td>
<td>3134972 (51.32%)</td>
</tr>
</tbody>
</table>

Following the alignment, the first obtained result shows us the distribution of read length across the samples of both conditions (Figure 4.9). As it can be verified, the majority of the reads have a size between 21-23 nucleotides, however, for condition 1 (healthy individuals), we have a more irregular distribution, finding a great set of sequences with 14-15 nucleotides. This might indicate that adapter trimming should have been further optimized, however, the adapter sequence was not provided in the original study from which the data derived.

The general characterization of the population of sncRNAs present in the datasets (considering only mapped reads) is displayed through the pie charts represented in Figure 4.10. As expected, the great majority of this population is composed by miRNAs (~70%), followed by IncRNAs (19%) and tRNAs (13%) in condition 1 and condition 2, respectively. Although IncRNAs were initially considered as being junk RNA, they are in fact involved in numerous biological and pathological processes [93]. As we can verify, the percentage of IncRNAs showed a decrease of 10% in the patients with hepatitis B, in comparison with healthy individuals. This indicates that there could exist an association between the expression of some IncRNAs and infection with hepatitis B. Such issue has already been addressed in recent published investigations [94], [95], which analyzed IncRNAs expression in patients with hepatitis B and also in patients with hepatocellular carcinoma related to hepatitis B. Both studies reported the
existence of several lncRNAs differently expressed between healthy individuals and individuals with hepatitis B related hepatocellular carcinoma. Such as miRNAs/isomiRs, further investigation on this subject might identify lncRNAs as being accurate diagnostic markers in some pathological conditions. Although we also detected a variation in the percentage of tRNAs across both conditions, no studies conducted on this subject were found.

![Figure 4.10](Image)

*Figure 4.10. General characterization of all types of small non-coding RNAs present across both conditions. (A) Represents condition 1 (Control) and (B) represents condition 2 (hepatitis B). These pie charts were created using the Total_noncoding.txt file produced by the pipeline.*

Regarding the selection of mature miRNAs origin (5p or 3p arm), it is believed that this selection is influenced by hydrogen-bonding selection and that the arm with the freer 5' end is preferentially incorporated into RISC to serve as the mature miRNA [96]. However, recent studies have reported that miRNAs derived from the -5p and -3p arms can be preferentially selected among different tissues, including variations between normal liver tissue and tumor liver tissue [97]. For this reason, we decided to include this information as a result of our developed pipeline. From Figure 4.11 it is possible to observe that almost no changes were verified between the two conditions, being the 3p arm, with a slightly difference, the most abundant origin of the mature miRNAs.

![Figure 4.11](Image)

*Figure 4.11. Comparison of the abundance of miRNAs that derived from the 3p or 5p arm of the pre-miRNA. (A) Represents condition 1 (Control) and (B) represents condition 2 (hepatitis B). These pie charts were created using the Counts_3p5p.txt file produced by the pipeline.*
As for the frequency of miRNAs in a general way, without taking into consideration their isoforms, the pipeline produced a file that allowed the construction of the bar charts displayed in Figure 4.12. This figure gives us a general view of the most abundant miRNAs in the two experimental conditions, from which we can observe that miR-92 is the miRNA with the higher read frequency, whether we are looking at the healthy individuals results (C1) or at the results of the patients with hepatitis B (C2). This is in accordance with the study from which the data derived, since they reported that miR-92a-3p was one of the most abundant miRNAs, whether in healthy individuals and in patients with hepatitis B [81].

![Figure 4.12. MiRNA's abundance across both conditions. (A) Represents condition 1 (Control) and (B) represents condition 2 (hepatitis B). Only miRNAs with a medium minimum frequency of 4 000 reads are displayed in the charts. These charts were created using the Counts_miRNAs.txt file produced by the pipeline.](image)

Concerning isomiRs, Figure 4.13 shows their distribution in both experimental conditions. No great differences are observed between the two conditions, being the 3’ end trimming the most abundant category of isomiRs. Once again, this fact is in accordance with the previously reported results and with the analysis published for the used datasets, which showed that the most abundant miRNA isoform (besides the canonical form) was the 3’ trimmed isomiR [81]. It was also possible to verify that the most predominant population of miRNAs was indeed the canonical form of miRNAs, which accounted for ~63% of all the identified sequences. This percentage didn’t vary across both conditions, which is also in accordance with the published results.

Finally, we could also evaluate the most frequent type of tailings that appear in the samples, as well as the type of internal editings. However, for the evaluated samples, no internal editings were identified, being the most uncommon type of modification. For this reason, no chart for this result is displayed.
On the other hand, it was possible to identify the most frequent type of tailings that appear in the samples (Figure 4.14). The addition of two adenine nucleotides was the most abundant tailing in both experimental condition, followed by the addition two uracil nucleotides. These have been previously considered the most common types of miRNA tailing, being carried out by nucleotidyl transferases such as non-canonical PolyA Polymerases and Terminal Uridylyl Transferases [98].

Figure 4.14. Types of tailings that appear in the samples of both conditions. (A) Represents condition 1 (Control) and (B) represents condition 2 (hepatitis B). These charts were created using the Counts_tailings.txt file produced by the pipeline.
The already published analysis for the used datasets also concluded that the percentage of isomiRs and their respective canonical form, showed a huge difference across individual miRNAs. To assess whether our pipeline was correctly classifying these two different populations, we also investigated two additional miRNAs in our results: hsa-miR-194-5p and hsa-miR-221-3p. A brief analysis allowed us to verify that, for hsa-miR-194-5p, the prevalent form was the canonical sequence (61%), followed by 3’ end modifications (39%), mainly including 3’ additions. On the other hand, for the hsa-miR-221-3p, the canonical sequence represented only ~20% of the population of this miRNA, since almost ~75% accounted for 3’ end trimming. The obtained results are in accordance with the previously published analysis of these datasets, however, our pipeline did not classify, in this case, any 3’ tailings isomiRs for the two mentioned miRNAs, which were reported in the published investigation. This may be due to the fact that the developed algorithm classifies as 3’ tailings, isomiRs with non-templated additions that have 2 or 3 nucleotides. Nevertheless, the algorithm was able to identify all 3’ non-templated additions of one single nucleotide. Although the publication does not specify what type of 3’ tailings isomiRs were found, this identified difference might be regarding non-templated 3’ additions of more than three nucleotides. However, previous reports indicate that 3’ tailings isomiRs display a maximum number of three nucleotides [99].

Finally, the last step of the annotation analysis module includes testing for differential expression between the two studied conditions. This test was performed by the DESeq2 R package, with a selected adjusted p-value of 0.05. As previously described, the pipeline uses the output of DESeq2 data as input for creating a heatmap, which is displayed in Figure 4.15. On the other hand, the results obtained directly from the DESeq2 output are shown in Table 4.8. By first analyzing the produced heatmap it is possible to visualize that samples are clearly clustering by condition, however one of the samples from condition 2 (individuals with hepatitis B) (C2R2) shows an evident different expression when comparing to samples from healthy individuals (condition 1). However, if we consider only the last isomiRs deriving from miR-92a-3p, here the differences between the samples of the studied conditions are contrasting.
Figure 4.15. Circulating miRNAs/isomiRs among the two different experimental conditions. The heatmap presents the normalized expression values for the miRNAs/isomiRs differentially expressed (p<0.05) comparing healthy individuals (C1) and individuals with hepatitis B (C2).
### Table 4.8. Table produced by the DESeq2 output (TableDE_*_.txt).

The table displays the output produced by DESeq2, which shows the statistical results for each isomiR/miRNA with the selected adjusted p-value < 0.5.

<table>
<thead>
<tr>
<th>IsomiR/miRNA identifier</th>
<th>baseMean</th>
<th>log2FoldChange</th>
<th>lfcSE</th>
<th>stat</th>
<th>pvalue</th>
<th>padj</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUGACCUAUGAAGAUGACAGAC_hsa-miR-215-5p_0_0_0_0 NA_0 NA_F NA</td>
<td>22,751</td>
<td>-3.976</td>
<td>0.783</td>
<td>-5.080</td>
<td>3.77E-07</td>
<td>8.97E-05</td>
</tr>
<tr>
<td>AAUGGCCGCCACUAGGUG_hsa-miR-652-3p_0_2_0_0 NA_0 NA_F NA</td>
<td>267,060</td>
<td>1.898</td>
<td>0.373</td>
<td>5.085</td>
<td>3.69E-07</td>
<td>8.97E-05</td>
</tr>
<tr>
<td>UUAUAAGGCAUGAGACUGAUU_hsa-miR-340-5p_0_0_0_0 NA_0 NA_F NA</td>
<td>152,928</td>
<td>-1.719</td>
<td>0.414</td>
<td>-4.157</td>
<td>3.23E-05</td>
<td>0.005124</td>
</tr>
<tr>
<td>UGUAAACAUCCUGACUGGAAGCU_hsa-miR-30a-5p_2_0_0_0 NA_0 NA_F NA</td>
<td>438,855</td>
<td>-1.349</td>
<td>0.330</td>
<td>-4.083</td>
<td>4.44E-05</td>
<td>0.005285</td>
</tr>
<tr>
<td>UACCACGUUGGGACUUGUCU_hsa-miR-106b-3p_0_2_0_0 NA_0 NA_F NA</td>
<td>92,607</td>
<td>1.530</td>
<td>0.394</td>
<td>3.880</td>
<td>0.000104</td>
<td>0.008285</td>
</tr>
<tr>
<td>UGUAAACAUCCUGACUGGAAGC_hsa-miR-30a-5p_1_0_0_0 NA_0 NA_F NA</td>
<td>78,642</td>
<td>-1.600</td>
<td>0.412</td>
<td>-3.883</td>
<td>0.000103</td>
<td>0.008285</td>
</tr>
<tr>
<td>UGAGAGAUAGACUGUAGCUCA_hsa-miR-143-3p_1_0_0_0 NA_0 NA_F NA</td>
<td>115,547</td>
<td>-1.902</td>
<td>0.523</td>
<td>-3.637</td>
<td>0.000276</td>
<td>0.018576</td>
</tr>
<tr>
<td>UGAGUAGAUGUUUGACUA_hsa-let-7g-5p_0_3_0_0 NA_0 NA_F NA</td>
<td>34,025</td>
<td>2.102</td>
<td>0.583</td>
<td>3.605</td>
<td>0.000312</td>
<td>0.018576</td>
</tr>
<tr>
<td>UGAGUAGAUGACUGUAGCUCA_hsa-miR-143-3p_0_0_0_0 NA_0 NA_F NA</td>
<td>397,280</td>
<td>1.717</td>
<td>0.485</td>
<td>-3.538</td>
<td>0.000403</td>
<td>0.021314</td>
</tr>
<tr>
<td>UAUGUCACUUGUCCCGGCUGUA_hsa-miR-92a-3p_1_0_0_0 U/A_23 Alt F NA</td>
<td>4077,215</td>
<td>2.122</td>
<td>0.356</td>
<td>3.429</td>
<td>0.000606</td>
<td>0.028866</td>
</tr>
<tr>
<td>UGAGUACUGUAGUGAUGU_hsa-let-7a-5p_0_1_0_0 NA_0 NA_F NA</td>
<td>352,867</td>
<td>-0.497</td>
<td>0.417</td>
<td>-3.498</td>
<td>0.000685</td>
<td>0.029302</td>
</tr>
<tr>
<td>UAUGUCACUUGUCCCGGCUGUA_hsa-miR-92a-3p_1_0_0_0 G/A_23 Alt F NA</td>
<td>3233,792</td>
<td>1.226</td>
<td>0.363</td>
<td>3.375</td>
<td>0.000739</td>
<td>0.029302</td>
</tr>
<tr>
<td>UCUCACACCUUGUCCAG_hsa-miR-150-5p_0_2_0_0 NA_0 NA_F NA</td>
<td>419,804</td>
<td>-1.399</td>
<td>0.417</td>
<td>3.353</td>
<td>0.0008</td>
<td>0.029302</td>
</tr>
<tr>
<td>UAUGUCACUUGUCCCGGC_hsa-miR-92a-3p_0_4_0_0 NA_0 NA_F NA</td>
<td>38,751</td>
<td>1.719</td>
<td>0.523</td>
<td>3.286</td>
<td>0.001016</td>
<td>0.032912</td>
</tr>
<tr>
<td>UGUAAACAUCCUGACUGGAA_hsa-miR-30a-5p_0_1_0_0 NA_0 NA_F NA</td>
<td>113,264</td>
<td>-1.270</td>
<td>0.387</td>
<td>-3.280</td>
<td>0.001037</td>
<td>0.032912</td>
</tr>
<tr>
<td>AAUGACAGCAUCUCGCCGGAG_hsa-miR-425-5p_1_0_0_0 NA_0 NA_F NA</td>
<td>73,563</td>
<td>1.472</td>
<td>0.455</td>
<td>3.236</td>
<td>0.001121</td>
<td>0.036032</td>
</tr>
<tr>
<td>UUCACAGUGCCAUAGGUUCU_hsa-miR-27b-3p_0_2_0_0 NA_0 NA_F NA</td>
<td>88,427</td>
<td>-1.264</td>
<td>0.395</td>
<td>-3.202</td>
<td>0.001364</td>
<td>0.038183</td>
</tr>
<tr>
<td>UUUGCGAAGUGGAGAAAGCA_hsa-miR-182-5p_0_2_0_0 NA_0 NA_F NA</td>
<td>518,985</td>
<td>1.456</td>
<td>0.461</td>
<td>3.158</td>
<td>0.001587</td>
<td>0.041979</td>
</tr>
<tr>
<td>UGUCAGUUGUCAAAUuAC_hsa-miR-223-3p_0_4_0_0 NA_0 NA_F NA</td>
<td>24,951</td>
<td>-1.909</td>
<td>0.610</td>
<td>-3.129</td>
<td>0.001756</td>
<td>0.044005</td>
</tr>
<tr>
<td>UCCUCACACCUUGUACCA_hsa-miR-150-5p_0_3_0_0 NA_0 NA_F NA</td>
<td>68,158</td>
<td>1.667</td>
<td>0.539</td>
<td>3.093</td>
<td>0.00198</td>
<td>0.047116</td>
</tr>
<tr>
<td>AUGACCUAGAAGUAGCACAG_hsa-miR-215-5p_0_0_0_0 NA_0 NA_F NA</td>
<td>22,751</td>
<td>-3.976</td>
<td>0.783</td>
<td>-5.080</td>
<td>3.77E-07</td>
<td>8.97E-05</td>
</tr>
</tbody>
</table>
The analysis of both the heatmap and the DESeq2 output table did not identify any of the reported canonical miRNAs in the published report as significantly changed upon hepatitis B infection. This may be due to somewhat different data pre-processing while trimming adaptors (this information was not provided in the original publication and we had to infer it), or to the fact that were are only analyzing a subset of the available datasets, along with the important fact that our analysis was performed at isomiR level and not at miRNA level. This differential expression analysis, at isomiR level, led us to conclude that two isoforms of the hsa-miR-92a appear, with clear evidence, up-regulated in the samples derived from patients with hepatitis B, when comparing to the samples of healthy individuals: UAUUGCACUUGUCCCGCCUGUA_hsa-miR-92a-3p_1_0_0_0_U/A_23_Alt_F_NA, which presents a 3’ end addition and a change of nucleotide (U/A) in the position 23 of the sequence, and UAUUGCACUUGUCCCGCCUGUA_hsa-miR-92a-3p_1_0_0_0_G/A_23_Alt_F_NA, which also presents a 3’ end addition and a change of nucleotide at this same position (G/A), in comparison with the canonical sequence (Figure 4.16). In order to assess if this identifier was being correctly created, we performed a manually search through the alignment file (SAM) to find these sequences, therefore verifying that they were both present in the file, and with the specified variations. By analyzing both these sequences it was possible to validate that they result in isomiRs with one 3’end nucleotide non-templated addition. However, the template nucleotide is different (Figure 4.16). This is explained by the fact that first maps in chromosome 13 (“+” strand), deriving from hsa-miR-92a-1 and the second maps to chromosome X (“-” strand), deriving from hsa-miR-92a-2. Importantly the algorithm is classifying these isomiRs and isomiRs displaying non-templated additions at the 3’ end, not identifying these as a single nucleotide 3’ tailing event. Indeed the algorithm, although as it is the case, finds these types of isomiRs, only 2-3 nucleotide 3’ tailings are classified as such, which should be improved in the near future. It could be of further interest to add the name of the pre-miRNA to the isomiR identifier, therefore facilitating identifying the precursor from which the isomiR derives.

Figure 4.16. Schematic representation of two isomiRs of the hsa-miR-92a-3p (A) IsomiR of the hsa-miR-92a-3p that derives from hsa-miR-92a-1. (B) IsomiR of the hsa-miR-92a-3p that derives from hsa-miR-92a-2. The top sequence corresponds to the pre-miRNA sequence, the sequence bellow corresponds to the identified isomiR and the nucleotide change is represented in red.

Another isomiR of hsa-miR-92a was also considered to be differentially expressed between the two clinical conditions: UAUUGCACUUGUCCCGCCUC_hsa-miR-92a-3p_0_4_0_0_NA_0_NA_F_NA (Figure 4.17). This isomiR presents a trimming of 4 nucleotides at the 3’ end when comparing to its correspondent canonical sequence.
Figure 4.17. Schematic representation of an isomiR of the hsa-miR-92a-3p. The pre-miRNA sequence is shown first, being the mature derived sequence represented in bold blue (retrieved from the miRBase). In light grey are represented the identified variations from the canonical sequence, which in this case is a deletion of the specified nucleotides.

For this reason, it was of our interest to further perform functional analysis for these three isomiRs, and also for their correspondent canonical form, in order to investigate if these variations have any effects on their biological functions. Since their modifications don’t really affect the seed region (situated at positions 2-7 from the miRNA 5´ end) we are not expecting contrasting changes respecting their targets. Nevertheless we performed a functional analysis, first for the 3 isomiRs of hsa-miR-92a-3p altogether and, secondly, for its canonical form. For this analysis we tested which was the best strategy for target prediction, to use miRanda, TargetScan, or both simultaneously. When using both target predictors we obtained the same results as when performing target prediction with only miRanda, so we decided to only use this target prediction tool for this specific analysis. From the gene lists obtained by miRanda it was possible to verify that two additional genes appeared in the list that had the isomiRs sequences as input, when comparing to the gene list obtained for the canonical sequence (Table 4.9). Since this analysis revealed two new targets, there was a possibility that the gene set enrichment analysis could reveal new functions for the identified isomiRs.

Table 4.9. Gene lists obtained from miRanda target predictor. The first column shows the targets predicted for the canonical form of the hsa-mir-92a-3p, and the second column shows the targets predicted for the isomiRs of the hsa-mir-92a-3p. In red, are represented the additional two targets predicted for the isomiRs.
Gene set enrichment analysis was then performed by the functional.pl + R_topGO.R, as previously described (see methods). This analysis showed us that the most relevant identified GO terms for both the isomiRs and the canonical form of the hsa-miR-92a-3p were the same, without any changes (Table 4.10). Also, they corresponded to very general biological GO terms, such as cellular component and cytoplasm. This non-specific results are not according to expected, since topGO is a network based algorithm that should allow the obtainment of more specific GO terms for Biological processes. This issue makes us believe that the usage of topGO within the pipeline may require further adjustment in order to provide more informative results. Additionally, the pipeline should also include the analysis of enriched pathways, which should have probably resulted in more specific results.

However, with the provided comparison of the gene lists it’s still possible to see which genes might be affected for these isoforms of the hsa-miR-92a-3p. ENSG00000059804 and ENSG00000068394 correspond to the SLCA3 gene and the GPKOW gene, respectively. The SLCA3 gene, which is an important gene involved in the transport of glucose and other sugars, has been previously identified, in an in vitro study, as being related to hepatitis C, since it appeared down-regulated when inducing this condition [100]. On the other hand, the GPKOW gene is identified as being essential for pre-mRNA splicing events [101]. However, none of these genes have been previously related to hepatitis B.

Table 4.10. Enriched GO terms that resulted from topGO analysis.

<table>
<thead>
<tr>
<th>GO ID</th>
<th>Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>GO:0005575</td>
<td>cellular_component</td>
</tr>
<tr>
<td>GO:0005737</td>
<td>cytoplasm</td>
</tr>
<tr>
<td>GO:0043227</td>
<td>membrane-bounded organelle</td>
</tr>
<tr>
<td>GO:0043231</td>
<td>intracellular membrane-bounded organelle</td>
</tr>
<tr>
<td>GO:0044464</td>
<td>cell part</td>
</tr>
<tr>
<td>GO:0044446</td>
<td>intracellular organelle part</td>
</tr>
<tr>
<td>GO:0016020</td>
<td>membrane</td>
</tr>
<tr>
<td>GO:0005622</td>
<td>intracellular</td>
</tr>
<tr>
<td>GO:0005634</td>
<td>nucleus</td>
</tr>
<tr>
<td>GO:0044444</td>
<td>cytoplasmic part</td>
</tr>
</tbody>
</table>

Since it was possible to verify that alterations on the sequence of the hsa-miR-92a-3p resulted in new target genes for this miRNA, it was also of our interest to investigate which specific modifications
caused this change of target genes. Therefore, functional analysis was performed for each identified isomiR of the hsa-miR-92a-3p individually. The resulting enriched GO terms were exactly the same for each isomiR and also for the canonical hsa-miR-92a-3p, as already shown in Table 4.7. Nevertheless, the resulting gene lists showed different results. For the two isomiRs with a modification of only one nucleotide (UAUUGCACUUGUCCGGCGUGUA_hsa-miR-92a-3p_1_0_0_0_U/A_23_Alt_F_NA and UAUUGCACUUGUCCGGCCUGUA_hsa-miR-92a-3p_1_0_0_0_G/A_23_Alt_F_NA) the produced gene lists contained exactly the same genes presented in the list produced for the 3 isomiRs altogether (Table 4.6). On the other hand, for the UAUUGCACUUGUCCCGGC_hsa-miR-92a-3p_0_4_0_0_NA_0_NA_F_NA, the obtained predicted target list was quite shorter (Table 4.11). Comparing this list to the list of the canonical targets, it was possible to verify that the trimming of the 4 nucleotides at the 3’ end affected its functionality, since it lost a lot of targets such as ENSG00000012061 (ERCC1), which is a 5′-3′ structure-specific endonuclease that is involved in a number of DNA repair pathways in mammalian cells [102]. This shows that these changes in the nucleotide sequence might have great affect when it takes to miRNA functionality.

Table 4.11. Gene list obtained from miRanda target predictor for the UAUUGCACUUGUCCCGGC_hsa-miR-92a-3p_0_4_0_0_NA_0_NA_F_NA.

<table>
<thead>
<tr>
<th>Gene List Obtained for the UAUUGCACUUGUCCCGGC_HSA-MIR-92A-3P_0_4_0_0_NA_0_NA_F_NA</th>
</tr>
</thead>
<tbody>
<tr>
<td>RGPD5</td>
</tr>
<tr>
<td>B4GALT7</td>
</tr>
<tr>
<td>MKRN2</td>
</tr>
<tr>
<td>INMT</td>
</tr>
</tbody>
</table>

Finally, prediction of novel miRNAs was performed separately, as an independent module from the remaining analysis, however no novel miRNAs were predicted by miRDeep2 for this input data. The original investigation in which these datasets were analyzed also used miRDeep2 for novel miRNA prediction and were able to identified 47 novel miRNAs. This result variation may arise from the great difference in the number of analyzed samples (they analyzed data from 72 serum samples) and also from the different strategy used for running miRDeep2. More specifically, they only used the reads that remained after all the annotation process, contrasting to our approach, that uses all the sequences present in all the Filtered_SAM*.sam files. Although some tools do not follow our approach, we decided to go with this method after performing some tests throughout the development of the pipeline.

Overall, this integrated analysis allowed us to show the ability of IsomiR Window pipeline to annotate RNA-seq data, identify miRNAs, isomiRs and also other small non-coding RNAs, also providing an automated expression analysis and functional analysis of selected isomiRs/miRNAs. When comparing to the previously reported investigation, our tool facilitates the analysis by turning the entire analysis in an automated process. Additionally, the published investigation for these datasets used the IsomiRage tool for isomiR detection, therefore not being able to identify isomiRs with both 5′ and 3′ end modifications, which is done by our pipeline.
The current project showed the necessity to take into more consideration the variations that occur in the miRNAs sequence, and how to they affect the cell’s biological pathways. As reported by the previous investigation, our analysis showed that, for several species of miRNAs, the great percentage of its population is composed by a specific type of isomiRs. This is the case of hsa-miR-221-3p, for which the most predominant form, in hepatitis B infection, is the 3’ end trimmed isomiR. Although the increased proportion of trimmed is evident, studies regarding the effects of these alterations in miRNAs functions are still lacking.

Additionally, given the high potential of miRNAs as biological markers, being useful for clinical studies and disease diagnosis, it will be extremely relevant to also consider the possibility of including an isomiR as a molecular marker, since it may represent the most expressed form of the respective miRNA.

5. Conclusions

Various software tools have emerged for small-RNA-seq data analysis. This shows the growing necessity for the facilitation and automation of this analysis. Therefore, the pipeline described throughout this project is based on a reliable and fully automated workflow, in order to facilitate an efficient and complete analysis of small-RNA-seq data, which is, nowadays, the type of data produced by the most recent high-performance next generation techniques. The pipeline integrates multiple open source resources combined to obtain the most accurate results, allowing the examination of multiple samples. It is capable of performing a precise detection of miRNAs and of all types of isomiRs and all possible combinations. Additionally, it also guarantees a general characterization of the entire population of sncRNAs existent in the input datasets. Furthermore, it also offers the possibility to test for differential expression analysis, followed by further functional analysis, which can be used to reveal possible different functional characteristics between isomiRs and their respective canonical forms. Functional analysis can also be used as an entire separate module to investigate isomiRs’ roles in certain biological processes, without having to necessarily run the annotation module.

In the future, it should be of great interest to continue improving the tool, by searching more suitable methods or statistically approaches such as for gene enrichment analysis or differentially expression analysis, in order to try and follow the recommendations made by the scientific community. Also, since the tool is already programmed to be able to receive data for a great number of species, it will be possible to incorporate all these species for performing analysis whenever the different required databases become available.
6. References


7. Appendices

7.1. Folder organization in the IsomiR Window tool context

Figure 7.1. Folder structure and organization within the IsomiR Window tool. This figure was retrieved from the project regarding the developed GUI that integrates the developed pipeline, and shows the folder structure within the IsomiR Window tool [90].
# Usage of the scripts from the command line

Table 7.1 Individual scripts commands and their usage. This table shows and describes all the arguments needed to invoke each script of the pipeline.

<table>
<thead>
<tr>
<th>Script</th>
<th>Command</th>
<th>Arguments</th>
</tr>
</thead>
</table>
| **find_ncRNAs.pl**   | perl find_ncRNAs.pl path_to_condition path_processing_files path_done_files path_filtered_SAM path_individual_results name_fastqfile number_mismatches number_multiple_hits species | 1. Path to the folder containing the input file  
2. Path to a folder that will contain processing files  
3. Path to a folder that will contain check point files  
4. Path to a folder that will contain the output `Filtered_SAM_*.sam` file  
5. Path to a folder that will include the results for each individual sample for one of the conditions  
6. Name of the FASTQ file to be used as input  
7. Number of mismatched to run Bowtie  
8. Number of genomic hits to allow for running Bowtie  
9. Species to be analyzed |
| **run_miRDeep2.pl**  | perl run_miRDeep2.pl ID path_to_C1_filtered path_to_C2_filtered path_input_miRDeep2 path_to_results miRDeep2_or_not species | 1. ID of the session (IsomiR Window context)  
2. Path to a folder including the `Filtered_SAM_*.sam` files resulted from every input FASTQ for the first condition  
3. Path to a folder including the `Filtered_SAM_*.sam` files resulted from every input FASTQ for the second condition  
4. Path to a folder that will include the input for running miRDeep2 (produced by the pipeline)  
5. Path to a folder that will include the output results  
6. A flag to run or not miRDeep2 ("yes" or "no")  
7. Species to be analyzed |
| **find_isomiRs.pl**  | perl find_isomiRs.pl path_to_condition name_of_SAM_file species_genome | 1. Path to the folder of the condition in analysis  
2. Name of the `Filtered_SAM_*.sam` to be analyzed  
3. Species to be analyzed |
| **deseq.pl + Rdeseq.R** | perl deseq.pl ID path_to_Deseq_C1_isomiRs path_to_Deseq_C2_isomiRs path_to_Deseq_C1_ncRNAs path_to_Deseq_C2_ncRNAs path_to_results path_done_file, paired_unpaired | 1. ID of the session (IsomiR Window context)  
2. Path to a folder containing all the `Filtered_SAM_*.sam_not_ambiguous.txt` files for the first condition  
3. Path to a folder containing all the `Filtered_SAM_*.sam_not_ambiguous.txt` files for the first condition  
4. Path to a folder containing all the `HTSeq_counts_*.txt` files for the first condition  
5. Path to a folder containing all the `Filtered_SAM_*.sam_not_ambiguous.txt` files for the second condition  
6. Path to a folder containing all the `HTSeq_counts_*.txt` files for the second condition  
7. Path to a folder that will contain the results  
8. Path to a folder that will contain check point files  
9. A flag indicating the experimental design of the data ("Paired_samples"/"Unpaired_samples") |
| **functional.pl + R_topGO.R** | perl functional.pl ID path_input_functional path_folder_inputs path_to_output species TargetScan miRanda | 1. ID of the session (IsomiR Window context)  
2. Path to the input FASTA file  
3. Path to a folder that will include the inputs for the target predictors (produced by the pipeline)  
4. Path to a folder that will contain the output  
5. Species to be analyzed  
6. A flag indicating to use ("1") or not ("0") TargetScan  
7. A flag indicating to use ("1") or not ("0") miRanda |
<table>
<thead>
<tr>
<th>Script Name</th>
<th>Command</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>charts.pl</td>
<td>perl charts.pl ID path_to_Deseq_C1_isomiRs path_to_Deseq_C2_isomiRs path_to_Deseq_C1_isomiRs_amb path_to_Deseq_C2_isomiRs_amb path_to_ncRNAs_C1 path_to_files_charts_C1 path_to_files_charts_C2</td>
<td>1. ID of the session (IsomiR Window context) 2. Path to a folder containing all the <em>Filtered_SAM_</em> files 3. Path to a folder containing all the <em>Filtered_SAM_</em> files 4. Path to a folder containing all the <em>Filtered_SAM_</em> files 5. Path to a folder containing all the <em>Filtered_SAM_</em> files 6. Path to a folder containing all the <em>Counts_ncRNAs_</em> files 7. Path to a folder containing all the <em>Counts_ncRNAs_</em> files 8. Path to a folder that will contain the results 9. Path to a folder that will contain the results</td>
<td></td>
</tr>
<tr>
<td>add_novel_miRNAs.pl</td>
<td>perl add_novel_miRNAs.pl ID path_folder_table_mirdeep</td>
<td>1. ID of the session (IsomiR Window context) 2. Path to the folder of the table produced by miRDeep2</td>
<td></td>
</tr>
</tbody>
</table>