FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Bioinformatic Tools to Decipher Biological Patterns in the Cytoskeleton of Nervous System Cells

Nuno Manuel Ferreira Côrte-Real

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Rui Camacho

Co-Supervisor: Helena Sofia Domingues

Co-Supervisor: Inês Mendes Pinto

August 6, 2020

Bioinformatic Tools to Decipher Biological Patterns in the Cytoskeleton of Nervous System Cells

Nuno Manuel Ferreira Côrte-Real

Mestrado Integrado em Engenharia Informática e Computação

August 6, 2020

Resumo

São vários os fatores que determinam a ocorrência de doenças neurológicas. Para melhor compreender estes fatores, células neurais são estudadas, nomeadamente oligodendrócitos. Falhas de funcionamento no citoesqueleto destas células poderão estar correlacionadas com a ocorrência de doença. Para ganhar uma melhor compreenção destas estruturas e potencialmente identificar padrões que as relacionem com certas doenças, exploramos o estado atual da área da bioinformática com o objectivo de encontrar uma ferramenta apropriada a resolver este problema. Deparamo-nos com uma ferramenta que apresenta grande potencial: Seurat, um pacote de R. Este pacote especializa-se no tratamento de dados obtidos através de sequenciação RNA de uma célula singular. Usamos um grupo de dados de um artigo particularmente inovador e analisamo-lo com o Seurat. Adicionalmente, realizamos uma análise de enriquecimento genético com DAVID, um recurso web que fornece um conjunto de ferramentas de anotação funcional. Construímos ainda uma aplicação web que visa centralizar todas as ferramente previamente mencionadas num recurso único que seja acessível e fácil de usar. Apesar de os nossos resultados não terem conduzido à descoberta de novos padrões no citoesqueleto de oligodendrócitos, este estudo serve como um sumário do estado de arte atual ao brevemente explorar as ferramentas mais predominantes e como um exercício de exploração do Seurat, exibindo a sua flexibilidade e potencial de aplicação em futuras experiências.

Keywords: Data Mining, RNA Sequencing, Oligodendrocyte, Cytoskeleton, Bioinformatics, Clustering, Biological Patterns, Central Nervous System, Neurology, Neurological Disease

Abstract

There are many factors that determine the occurrence of neurological disease. To try to understand these factors, neural cells are studied, namely oligodendrocytes. Malfunctions in the cytoskeleton of these cells may be connected to the occurrence of disease. To gain a better understanding of these structures and potentially identify patterns that relate them with certain diseases, we explore the current state of the art regarding bioinformatic research in order to find the correct tools to tackle this issue. We find one tool that presents great potential: Seurat, a package for R. This package is specialized in treating data obtained from single-cell RNA sequencing. We use a dataset from a breakthrough paper and analyse it with Seurat. Additionally, we perform a gene enrichment analysis with DAVID, a web resource that provides a set of functional annotation tools. We also built a web application that aims to centralize all the previously mentioned tools together in a single user friendly, accessible resource. While our results didn't provide much insight regarding new patterns in the cytoskeleton of oligodendrocytes, this paper serves as a summary of the current state of the art by briefly exploring the more predominante tools in the area and as an exercise of exploration of the Seurat tool, showcasing its flexibility and potential for further research.

Keywords: Data Mining, RNA Sequencing, Oligodendrocyte, Cytoskeleton, Bioinformatics, Clustering, Biological Patterns, Central Nervous System, Neurology, Neurological Disease

Ackowledgements

I would like to thank everyone that in one way or another contributed to make this project happen. Firstly, Rui Camacho, Helena Domingues and Inês Pinto, my supervisors and co-supervisors, whose disponibility, eagerness to help, orientation and general positivity were crucial to this work. Secondly, to all my friends, whose unconditional support and help always guided and kept me in the right path.
Finally, to my family and girlfriend, the embodiment of true faith.

Nuno Côrte-Real

"What we cannot speak about we must pass over in silence."

Ludwig Wittgenstein

Contents

1 Introduction **1**
1.1 Document Structure . . . 2

2 State of the Art **3**
2.1 Introduction . . . 3
2.2 Context . . . 3
2.3 Neuroscience . . . 6
2.3.1 Introduction . . . 6
2.3.2 Marques et al., 2016 . . . 6
2.3.3 Falcão et al., 2018 . . . 7
2.3.4 Other Studies . . . 8
2.4 Data Science . . . 11
2.5 Tools . . . 14
2.5.1 Data Mining Tools . . . 15
2.5.2 Databases and Web Repositories . . . 16
2.6 Summary . . . 17

3 Methodologies and Architecture **19**
3.1 Introduction . . . 19
3.2 Methodologies . . . 19
3.2.1 Seurat Pipeline . . . 20
3.2.2 DAVID . . . 23
3.2.3 Web Application . . . 23
3.3 Architecture . . . 24

4 Case Study **27**
4.1 Introduction . . . 27
4.2 Seurat . . . 27
4.3 Web App . . . 31

5 Results **37**
5.1 Introduction . . . 37
5.2 Overview . . . 37
5.3 Seurat . . . 37
5.4 DAVID . . . 38
5.5 Web App . . . 38

6 Conclusions **41**
6.1 Introduction 41
6.1.1 Conclusions 41
6.2 Future Work 42

References **43**

A Appendix **47**

List of Figures

2.1 Illustration of the myelination process. . . . 4
2.2 Strucutre of the cytoskeleton. . . . 5
2.3 Oligodendrocyte differentiation. . . . 6
2.4 Expression of marker genes for oligodendrocytes. . . . 7
2.5 Example of a gene expression matrix. . . . 8
2.6 Oligodendrocyte populations graphs. . . . 9
2.7 tSNE projection of oligodendrocyte maturation. . . . 10
2.8 Overview of the Seurat dataset integration method. . . . 15

3.1 DAVID workflow and features. . . . 24
3.2 System architecture diagram. . . . 25
3.3 Use cases diagram. . . . 26

4.1 Violin plot showing the relationship between features and RNA counts. . . . 28
4.2 Scatter plot showing the relationship between features and RNA counts. . . . 29
4.3 Top 10 most expressed features. . . . 30
4.4 Top 10 most expressed genes for 4 PCs. . . . 31
4.5 Comparison of the resulting dimensionality reduction between 2 PCs. Cells are colored by their identity class. . . . 32
4.6 Heatmap comparing 500 genes of 1 PC. . . . 33
4.7 Data post Jackstraw application, showing the p-values for 15 PCs. . . . 33
4.8 Elbow plot representing a ranking of PCs based on the percentage of variance. . . . 34
4.9 Visualization of the data post UMAP application. Comparison between 2 PCs. Each cluster is represented with a different color. . . . 34
4.10 Violin plots representing the expression of each of the genes common to all clusters by cluster. . . . 35
4.11 Plot comparing the average expression of each of the genes common to all clusters. 35

5.1 Cluster trees comparison. . . . 39
5.2 Overview of the web application interface. . . . 40
5.3 Overview of the plot display screen. . . . 40

A.1 Cluster tree obtained using Seurat's native methods. . . . 47
A.2 Cluster tree obtained using the exterior package clustree. . . . 48
A.3 Structure of the input configuration JSON file. . . . 49

List of Tables

3.1 Versions of the tools used. 20
3.2 Web application User Story table. 24

Abbreviations

CNS	Central Nervous System
GO	Gene Ontology
MS	Multiple Sclerosis
OL	Oligodendrocyte
MOL	Mature Oligodendrocyte
OPC	Oligodendrocyte Percursor Cell
COP	Differentiation Committed Oligodendrocyte precursor Cell
MFOL	Myelin-Forming Oligodendrocytes
FACS	Fluorescence Activated Cell Sorting
PCA	Principal Component Analysis
PC	Principal Component
CCA	Canonical Correlation Analysis
SNN	Shared Nearest Neighbor
KNN	K Nearest Neighbor
scRNA-seq	Single Cell RNA Sequencing
snRNA-seq	Single Nucleus RNA Sequencing
tSNE	t-Distributed Stochastic Neighbor Embedding
UMAP	Uniform Manifold Approximation and Projection
JSON	JavaScript Object Notation
HTML	HyperText Markup Language

Chapter 1

Introduction

Neurological disorder occurrence and the burden it implies has increased substantially over the past 25 years. This might be due to the correlated increase of life expectancy throughout the years, which results in expanding population numbers and prolonged ageing.

As such, cases of neurological disorders such as Multiple Sclerosis, Parkinson's disease, Alzheimer's disease and Dementia are expected to increase in the next few decades along with the associated need for new therapeutic solutions. One of the possible approaches when facing this issue is a multidisciplinary one, where Neuroscience research comes together with the latest technological developments in Data Mining and Bioinformatics, in order to arrive at new solutions.

A variable of particular interest to take into account when attempting to predict the occurrence of neurological disease is the cytoskeleton of a cell. This structure is a network of subcellular filaments that provide the ability to resist and react to applied external stress. Defaults and mutations in this structure are associated with the incidence of neurological disease. Thus, the possibility to accurately analyse patterns in this structure is extremely valuable for the successful prediction of neurological disease occurrence.

To address the aforementioned problems, the International Iberian Nanotechnology Laboratory focuses in applying bioinformatics tools in order to decipher biological patterns in cellular structures of interest, such as oligodendrocytes and neurons. With that being said, the aim of this project is to expand on the already conducted research, by utilizing the gathered available data, analyze it and process it with Data Mining and Bioinformatics tools and through these generate new knowledge.

Are the proposed goals to be achieved and valuable, ground-breaking data will be generated, which will prove itself of great use in future Neuroscience research. A much needed deeper understanding in how Data Mining and Bioinformatics tools can be applied to this area of knowledge will also be potentially achieved.

Thesis hypothesis

This project bases itself on the premise that through the application of Data Mining tools on the available data bases, it is possible to generate new useful knowledge regarding neurological disease occurrence and prediction.

We will assess if Data Mining tools are useful to help medical practitioners in the prediction of diseases caused by malfunction of neural cells.

In order to test this, we intend to use Data Mining tools, information from repositories and previously conducted research.

1.1 Document Structure

In this report, we first contextualize this topic, detailing the bridge between the two areas of knowledge this study is concerned with: Neurology and Bioinformatics (chapter 2, sections 2.1 and 2.2).

We then explore the state of the art of the two main scientific areas this study is concerned with. Regarding Neurology, we briefly explore some of the studies form the last few years that provided some of the more breakthrough conclusions related to this topic (chapter 2, section 2.3). Regarding Bioinformatics, we compile an explore some of the more relevant tools in recent years, having a few of them been used in studies related to this topic (chapter 2, sections 2.4 and 2.5). Through the exploration of the state of the art, we aim at deepening the contextual basis.

Following this analysis we explain the methodology used for this study and the rationale behind it (chapter 3).

We then present a case study where we take a dataset and analyse it through the aforementioned methodology (chapter 4).

The next part lists the obtained results and discusses them (chapter 5).

We end this document with the reached conclusions and suggestions for future improvements (chapter 6).

Chapter 2

State of the Art

2.1 Introduction

In this chapter we contextualize in further detail the topic at hand. Additionally, to deepen comprehension, we address the current state of academic development of the two distinct sciences in which this paper is inserted in: Data Science and Neuroscience. This is achieved by analysing conducted research that presents itself as relevant to this topic, or that contributed to any kind of technological, theoretical or practical advances within specific areas of knowledge from which data can be extracted and used in the context of this paper.

2.2 Context

In the study of neurological diseases, there is a cell that is particularly relevant: oligodendrocytes (OL). These cells, present in the central nervous system (CNS), originate from the spinal cord and emigrate to various regions of the brain. Their role is to produce a myelin sheath around axons, in a process called myelination. An illustration of this process and of the strucuture of OLs can be seen in Figure 2.1.

This sheath improves the velocity of transmission of an electronic impulse throughout the axon, and therefore is essential to the development of motor capabilities [Pfeiffer et al., 1993].

Malfunction in oligodendrocytes is associated with occurrence of neurological disease [Falcão et al., 2018]. One of the causes for malfunction in these cells might have its origins in the cytoskeleton. The cytoskeleton of a cell is a structure consisting of microtubules, found in the cytoplasm (the liquid surrounding the cell's organelles) and has various functions, namely providing a form to the cell, organize the cell's organelles and is involved in various celular processes. An illustration of the strucutre of the cytoskeleton of a generic cell can be seen in Figure 2.2.

Since their inception up until their emigration to a specific region of the brain, OLs go through a process of maturation called differentiation. They go through various states in an uniform and

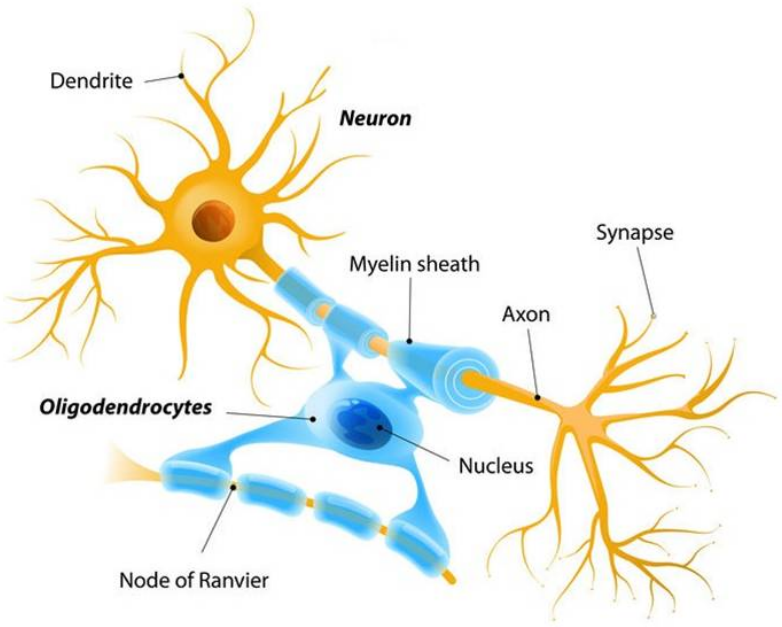


Figure 2.1: Illustration of the myelination process and the structure of an oligodendrocyte, as seen in [Aitamurto, 2015].

continuous manner until reaching a mature state. Upon reaching their mature state, OLs specialized in certain tasks and finally emigrate to a region of the brain where those specialized cells are needed [Marques et al., 2016].

The various states an OL goes through, in order, are listed above:

1. VLMC: vascular leptomeningeal cell
2. OPC: oligodendrocyte percursor cell
3. COP: differentiation-committed oligodendrocyte percursor cell
4. NFOL1/NFOL2: newly formed oligodendrocyte
5. MFOL1/MFOL2: myelin-forming oligodendrocyte
6. MOL1...MOL6: mature oligodendrocyte

An illustration of the OL differentiation process can be seen in Figure 2.3.

This process of specialization is achieved through the selective expression of genes. Through the analysis of the expression of certain specific genes, called marker genes, we can infer in what step of the differentiation process an OL is currently at, and through the analysis of the cellular strucutres and biological processes associated with those genes we can infer what functions the OL has [Marques et al., 2016]. An example of the variation of certain marker genes and their associated OL differentiation state can be seen in Figure 2.4.

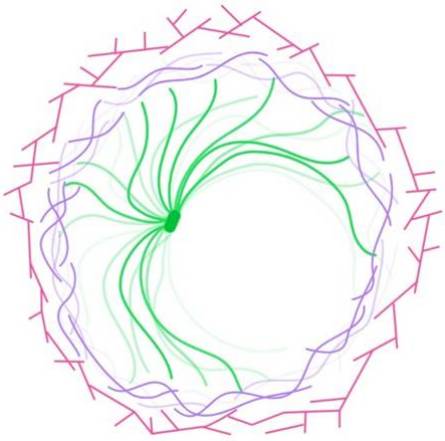


Figure 2.2: Structure of the cytoskeleton of a generic cell, as seen in [Mokobi, 2020].

The study of cells is done through analysis of the genetic information. This information is stored in DNA molecules. These molecules are replicated into RNA molecules, through a process called transcription. Each RNA molecule is constituted by genes. Each gene codifies a protein, that can be produced through a process called translation. These processes are the basis of cellular functioning [Clancy and Brown, 2008]. The complete set of genetic information of a cell (or a number of cells) is called transcriptome. It is through the analysis of transcriptomes that knowledge is obtained, regarding the inner workings of cells [Wang et al., 2010].

A transcriptome can be obtained through a process called Single Cell RNA-Sequencing (scRNA-seq) [Wang et al., 2010]. This processes generates a gene expression matrix containing the information of the cell. This matrix is constituted by columns, representing cells and rows, representing genes. Each cell of the matrix represents the expression of a certain gene, in a certain cell. An example of a gene expression matrix can be seen in Figure 2.5

In study of genes, one type of analysis is particularly important: Gene Ontology (GO) analysis, also referred to as gene enrichment analysis. This process consists of identifying genes that might present expression values higher than usual by comparing them with known expression values that are considered normal for those genes. Subsequently, a report is done identifying the cellular structures and biological processes associated with the enriched genes of a dataset. Researchers can then analyse these reports in an explorative manner in order to draw conclusions that might be relevant for their studies. This analysis is done resorting to specialized GO tools, and is a crucial step when attempting to gain a deeper understanding of a cell's working [Harris et al., 2008].

As such, through the analysis of genes related to the cytokeleton of OLs, we hope to identify certain patterns that might be related with the occurrence of specific neurological diseases.

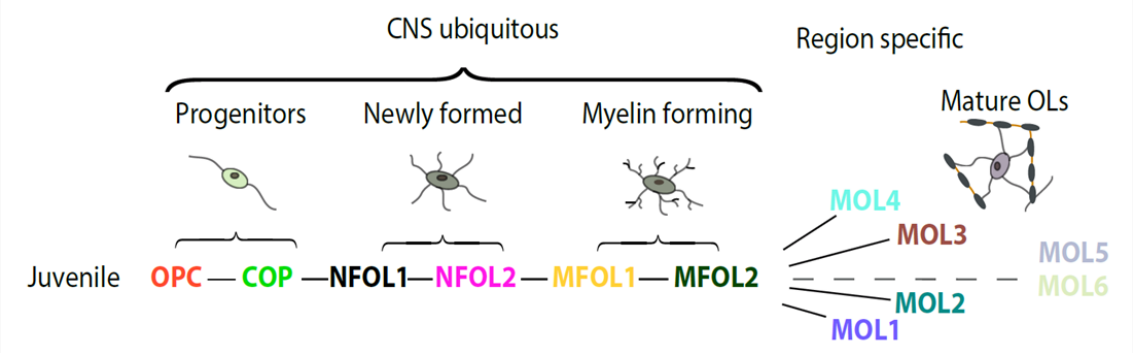


Figure 2.3: The various states an oligodendrocyte goes through in the differentiation process [Marques et al., 2016].

2.3 Neuroscience

2.3.1 Introduction

This section addresses the current developments in Neuroscience research which contribute to the current, consolidated state of knowledge regarding the topic of this paper. This is achieved through the listing of some of the more relevant conclusions reached regarding related studies that are relevant to this topic.

The collected knowledge is then utilized and extrapolated to the context of the cytoskeleton of the aforementioned cell types. Two studies in particular have provided the basis for this report.

2.3.2 Marques et al., 2016

In the first study, OL cells from ten distinct regions of the anterior-posterior and dorsal-ventral axis of the mouse juvenile and adult CNS, including grey matter (spinal cord/dorsal horn, substantia nigra and ventral tegmental area (SN-VTA), amygdala, hypothalamic nuclei, zona incerta, hippocampus/dentate gyrus and CA1, and somatosensory cortex), white matter (corpus callosum) or both (striatum), and also adult CNS (somatosensory cortex, corpus callosum and dentate gyrus) were isolated and treated, and a gene expression matrix was obtained through scRNA-seq. This matrix consisted of 5072 cells and about 23,500 genes. The dataset was then analysed, and through clustering 13 groups were identified. Additionally, a cluster tree, which is a dendogram illustrating the descendecy relationship between cluster groups was built. By analysing the dataset, namely through GO analysis, and the aforementioned cluster tree, each group was identified as a OL population in a specific state of the differentiation process. This breakthrough study provided a deep comprehension of the various states OLs go through during the differentiation process [Marques et al., 2016].

The gene expression matrix used in this study was used as the basis of the analysis conducted in this report. This dataset was taken from the GSE75330, the repository hosting the datasets used in this study, in the Gene Expression Omnibus (GEO) website, a public functional genomics data repository related to the National Center for Biotechnology Information (NCBI).

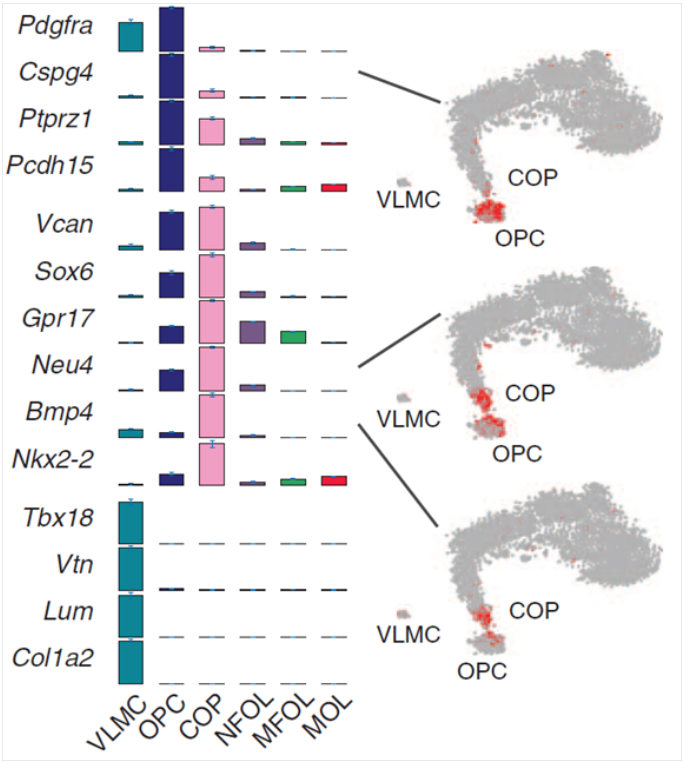


Figure 2.4: Expression of marker genes for oligodendrocytes. [Marques et al., 2016].

Additionally, the tool used in this study for GO analysis, DAVID, was used in this report.

An illustration of the summarization of the results of this study can be seen in Figures 2.6 and 2.7.

2.3.3 Falcão et al., 2018

The second study focused on the roles of OLs in the context of disease, namely Multiple Sclerosis (MS). Mice were induced with experimental autoimmune encephalo-myelitis (EAE), which mimics several aspects of MS. Through a similar process as seen in [Marques et al., 2016], OLs were isolated from these mice and a gene expression matrix was obtained through scRNA-seq. Analysis to this dataset showed that in the context of EAE, OLs went through a re-transcription process; more specifically, genes associated with immunoprotection and innate and adaptive immunity were given priority of expression. This immunoprotective and adaptive response was correlated with malfunctions in the myelination process, thus causing some of the symptons seen in EAE and MS like loss of motor capabilities [Falcão et al., 2018]. This study provided breakthrough knowledge on the role of the myelination process in the context of neurological disease. Clustering was done with GeneFocus, a personalized pipeline based on the R language, and the Seurat package, which specialized in the treatmen of data obtained thorugh scRNA-seq. This language and package were thus used in the analysis conducted and described in this report.

Cell ID	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11
Xkr4	0	0	0	0	0	0	0	0	0	0	0
Rp1	0	0	0	0	0	0	0	0	0	0	0
Sox17	0	0	0	0	0	2	0	0	0	0	0
Mrpl15	0	0	1	0	0	0	0	10	0	0	0
Lypla1	0	0	0	0	0	0	0	0	0	0	0
Tcea1	2	0	0	0	0	0	3	0	1	0	0
Rgs20	0	0	0	0	0	0	0	0	0	0	0
Atp6v1h	2	0	0	1	1	1	0	0	0	0	0
Oprk1	0	0	0	0	0	0	0	0	0	0	0
Npbwr1	0	0	0	0	0	0	0	0	0	0	0
Rb1cc1	0	0	2	0	0	0	0	3	0	0	0
Fam150a	0	0	0	0	0	0	0	0	0	0	0
St18	0	0	0	0	0	0	0	0	0	0	0
Pcmtd1	0	0	0	0	0	1	0	0	1	0	0
Sntg1	0	0	0	0	0	0	0	0	0	0	0
Rrs1	0	0	0	0	0	0	0	0	1	0	0
Adhfe1	0	0	0	0	0	0	0	0	0	0	0
2610203C	0	0	0	0	0	0	0	0	0	0	0
3110035E1	8	6	8	5	1	0	3	9	1	5	4
Mybl1	0	0	0	0	0	0	0	0	0	0	0
Vcpip1	0	0	0	0	0	0	2	0	0	0	0
1700034P1	0	0	0	0	0	0	0	0	0	0	0
Sgk3	0	0	0	0	0	0	0	0	0	1	0
Mcmdc2	0	0	0	0	0	0	0	0	0	0	0
Snhg6	0	0	0	0	0	0	2	2	0	0	0
Snord87	0	0	0	0	0	0	0	0	0	0	0
Ppp1r42	0	0	0	0	0	0	0	0	0	0	0
Cops5	4	0	0	1	0	4	14	0	0	1	0
Cspp1	0	0	0	0	0	0	3	0	0	0	0
Arfgef1	0	1	0	0	0	0	4	0	0	0	0
Cpa6	0	0	0	0	0	0	0	0	0	0	0
Prex2	0	0	0	0	0	0	0	0	0	0	0

Figure 2.5: Example of a gene expression matrix. The columns represent cells and the rows represent genes. Each matrix cell represents the expression of a certain gene, in a certain cell.

2.3.4 Other Studies

An also recently conducted analysis showed that the heterogeneity of OLs in mice is wider than previously stipulated. This analysis showed that progenitor OL cells with different temporal and spatial origins in the central nervous system converge into similar OPC transcriptionals states. This process of convergence is correlated with electrophysiological responses and leads to the differentiation of OLs into six mature cells states, which implies that the differentiation process may not be cell intrinsic but rather be induced by various forms of stimulus derived from the local cellular environment [Marques et al., 2018].

Another relevant study focused on the gene expression profile of human microglia and its comparison with mice models, and how these two differentiate from each other regarding aging pro-

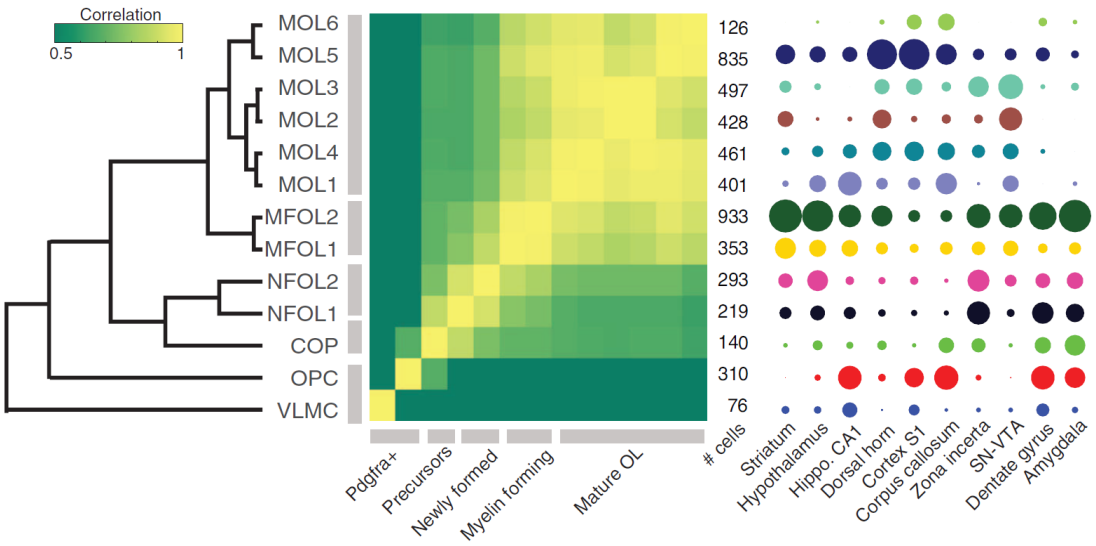


Figure 2.6: Dendogram, heat map and dot plot showcasing, respectively, the different oligodendrocyte populations, their progression through the oligodendrocyte maturation process and the zones from where they were extracted, as seen in [Marques et al., 2016].

cesses. This is relevant since human microglia are primarily implicated in host defense and in the modulation of immune responses. This study concluded that there are critical differences between human and mouse microglia, especially in the aging process, which highlight the necessity to independently study human microglia instead of basing knowledge from mouse models [Galatro et al., 2017].

Lastly, two more recent studies provided extremely useful insights about the inner workings of the process of oligodendrocyte differentiation.

In the first, the mechanical plasticity of oligodendrocytes during differentiation was analysed. The authors reported how cytoskeleton-based mechanosensors and mechanotransducers partake in the differentiation process, and how certain bilayer-associated proteins (MBP, PLP and CNP) are essential in stabilizing and maintaining the myelin structures built by differentiated oligodendrocytes [Domingues et al., 2018].

In the second, the authors studied the role of the regulatory protein Jmy in the differentiation process. This protein is upregulated during myelination and is required for the assembly of actin filaments and protusion formation during differentiation, allowing oligodendrocyttes to acquire an arborized morphology. This mechanisms are closely tied to interactions in the cytoskeleton of the cells question.

For this study, the authors designed a tool called OligoMacro. This tool is a semi-automated, open source macro-toolset for ImageJ (a Java image processing program [Rasband,]) that aimed at analyzing OL morphology during differentiation, in a spatiotemporal scale.

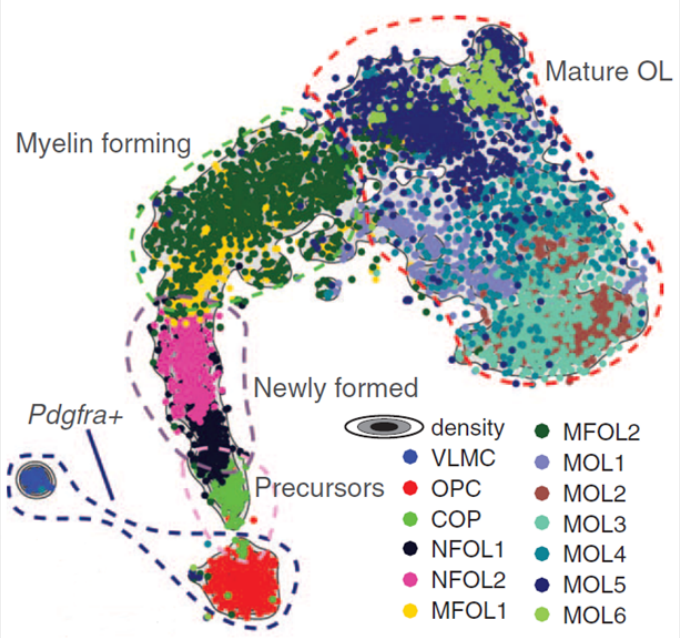


Figure 2.7: t-Distributed stochastic neighbor embedding projection showing the trajectory from OPCs to mature oligodendrocytes [Marques et al., 2016].

This tool, the first of its kind, is an example of a recent breakthrough in Bioinformatics that showcases great potential for future research in this topic [Azevedo et al., 2018].

Two very recent studies provide important insight into the role of OLs in neurological disease.

The first study focused on the heterogeneity of these cells in the context of Multiple Sclerosis.

A novel technique, single-nucleus RNA sequencing, was employed. This technique is similar to scRNA-seq but focuses only on the nucleus of cells. This approach was made resorting to tissue from white matter areas of postmortem human brain from patients with MS (tissue from unaffected areas was used as control).

Through the results obtained, the authors reached some important conclusions that defy the current default approaches for interpreting regenerative therapies in Multiple Sclerosis. Namely, the assumption that enhancing the differentiation of OPCs to OLs that express myelinating genes and proteins directly leads to enhanced remyelination in progressive MS isn't always correct.

The authors observed that MS doesn't happen due to a failure of OPCs differentiation into OLs expressing myelin genes and that the observed loss of a specific population of OLs in MS might play a significant role.

Their results also suggest that subsets of mature OLs contribute to remyelination [Jäkel et al., 2019].

The second study aimed at analysing the role of OLs and the myelination process on age-related deficits in memory.

The authors observed, through studied conducted on mice, that the inhibition of myelination

in OPCs impaired spatial memory in young mice, while the enhance of myelination in OPCs or promotion of OL differentiation recovers spatial memory decline during aging. Such recovering is possible due to the fact that OPCs populate the aged CNS and maintain their potential to differentiate and myelinate.

These observations are aligned with the current understanding that memory deficits are correlated with reduced myelination activity and corresponding decrease in white matter volume in ageing brains.

The authors end up by concluding that rejuvenating myelination can rescue synaptic loss in the hippocampus and improve memory function in aging mice, that spatial memory function requires dynamic myelination in mature adult brains and that diminished myelination in aging tissues can be partly responsible for declines in memory [Wang et al., 2020].

Yet another study that presents itself as relevant to this topic is an older one that focused on the diversity of brain cell types, transcriptomes and the mechanisms responsible for the maintenance of adult differentiated cell types in mice.

The authors resorted to data mining techniques to identify a total of 47 distinct cell subclasses, comprising all known major cell types in the cortex, along with various marker genes, allowing for a correlation of known cell types with morphological characteristics and their location.

One of the more interesting data mining techniques used in this study is the specifically developed BackSPIN, a divisive biclustering method based on sorting point into neighbourhoods, which was essential to filter data noise and identify cell subclasses [Zeisel et al., 2015].

2.4 Data Science

This section addresses the current state of development of Data Science technologies, as well as relevant conducted research that resorted to Data Science methods in order to produce new knowledge within Neuroscience.

The Data Mining process can usually be divided into 4 principal stages: Pre-Processing (cleaning of irrelevant data, selection and transformation of the to be analyzed data), Data Mining (application of methods to the data), Result Validation and Knowledge Presentation.

Regarding the Data Mining stage, there is a wide variety of approaches that can be used. Some relevant ones include:

- Classification: tries to find a model that allows to group elements into a group of data. There are various algorithms that can be used for this process, such as C4.5 (builds decision trees where each node is the attribute that better fits a data group), Support Vector Machine (non-probabilistic, linear, binary classifier), Random Forest (based on the combination of various decision trees) and Naive Bayes (uses the Bayes theorem and probability to classify data). The validation of this process can be done resorting to different models, like Cross

Validation (when the main objective is prevision) and Bootstrap (re-utilization of data used in training)

- Regression: analyzes elements according to the relations between each other and tries to predict the value of a variable or at what group a certain element belong to.

- Association: tries to estimate the probability of occurrence of a certain element according to any existing similarities.

- Clustering: a concept of particular interest for this project, this technique consists in the grouping of various elements into distinct groups, according the the similarities they possess between them. Some of the more prominent clustering algorithms are k-Means (uses the average mean of all elements of a cluster to group them), Farthest First (uses centroids i.e. points that are the furthest apart from each other), Expectation-Maximization (calculates the probability of each element belonging to each one of the clusters) and Density Based Spatial Clustering of Application with Noise (based on the density of the clusters, can identify and filter noise), among others [Miguel and Natividade, 2017].

Recent technologies have allowed the birth of single-cell RNA sequencing (scRNA-seq), a process to analyse in a detailed manner the transcriptional activities of a cell. It is an extremely useful method that allows comparisons between strains of cells and the analysis of the progress of a disease, for example. More specifically, it allows the analysis of the transcriptome of a single-cell, which in turn provides new possibilities of in-depth comparison between cells.

More specifically, in a recent study, a single-cell latent variable model (scLVM) was constructed in order to better understand hidden variables in single-cell RNA-sequencing studies. Such variables, like cell cycle, lend to an increase in heterogeneity in gene expression and lead to confusion in the interpretation of results. This created approach can thus be used to counter the negative effects of these variables [Buettner et al., 2015].

One of the main challenges in single-cell transcriptome analysis is the grouping of cells that belong to the same cell types, based on gene expression patterns. Due to the amount of noise and the sheer dimension of the data in question, as well as the stochastic nature of the biochemical processes of a cell, this clustering process is often difficult and costly.

A recently developed clustering algorithm, called Shared Nearest Neighbour (SNN-Cliq) tackles this problem. This algorithm automatically determines the number of clusters in a data set, while being able to identify clusters of different densities and shapes and avoiding the disregard of data points in regions of low graphical density.

Regarding its efficiency, SNN-Cliq outperforms the other available methods, such as K-Means and Density-Based Spatial Clustering of Applications with Noise, without sacrificing ease of use [Xu and Su, 2015].

Such developments offer new possibilities for future research, data analysis and interpretation of results.

The appearance of new scRNA-seq methodologies raises questions regarding the way in which the resulting data of these processes can be analysed. Normalization of scRNA-seq data must properly account for differences in the amount of RNA transcribed within a cell and in sequencing depth. Additionally, more methods like SNN-Cliq that remove confusing variables and noise from data are needed, in order to better improve the fidelity of results and avoid the compromise of downstream interpretation.

Another problem inherent to scRNA-seq is the integrated or comparative analysis between different data sets consisting of multiple transcriptomic populations.

To address this issue, a strategy that aims to integrate scRNA-seq data sets by identifying shared sources of variation between two data sets was recently developed. This strategy, consisting in a number of steps within a Seurat workflow, was tested with an experiment, where it successfully aligned cell types between human and mouse pancreatic islets, identifying a shared population of beta cells responding to ER protein misfolding stress [Stegle et al., 2015].

As such, we can group the aforementioned methodology with others of its kind that have been recently birthed and have been tackling issues withing Neuroscience, providing answers and the possibility of future research developments.
With scRNA-seq, researchers have for the first time the ability to obtain snapshots of individual cell states with unprecedented resolution, which is a valuable asset for the characterization and study of the cell lineage of oligodendrocytes.

A key point of scRNA-seq studies is what is the biological relevance of the identified cell clusters and what differentiates a cell type from a cell state [van Bruggen et al., 2017].

Comparison between different clustering algorithms is an approach that is increasingly used to verify the robustness of the clusters.
Through the methods, studies and developments in Data Mining and Bioinformatics tools mentioned above, researchers have been increasing the amount of data collected, regarding the study of neural cells.

With this gathered data, new databases have been created, which can be utilized for further research.
For example, in a recently conducted study that aimed to answer the question of how many genes are alternatively spliced in the mouse cortex, a high quality database was generated, containing information regarding the transcriptome of neurons, astrocytes, oligodendrocyte percursor cells, newly formed oligodendrocytes, myelinating oligodendrocytes, microglia, endothelial cells and pericytes from mice.

This database was built using scRNA-seq and an algorithm to detect alternative splicing events in each of the eight analysed cell types.

Through this study, important conclusions were reached. A large number of cell type-enriched genes that had not yet been described previously as cell type-specific was discovered.

It was also discovered numerous cell type-specific or enriched transcription factors, including well known factors and a large number of factors that were not recognized previously to be cell-type specific.

Additionally, the enrichment of the inhibition of matrix metalloproteases and intrinsic prothrombin activation pathways in oligodendrocyte percursor cells (OPCs) was detected, suggesting that OPCs have unique properties in their interactions with the extracellular matrix compared with other cell types of the brain [Zhang et al., 2014].
Another breakthrough tool that has been developed recently and used in the bioinformatic processes of some of the above mentioned papers is Seurat [Lab, 2020]. Released in 2015 and regularly updated, Seurat is an R package designed for quality control, analysis and exploration of single-cell RNA-seq data [Butler et al., 2018].

This tool's pipeline provides various operations for processing single-cell RNA-seq data, such as cell filtering, normalization, feature selection, data scaling, linear dimensional reduction (such as Principal Component Analysis), dimensionality determination, clustering, non-linear dimensional reduction (t-SNE and Uniform Manifold Approximation and Projection for Dimension Reduction - UMAP) as well as various methods for finding differentially expressed features and cluster biomarkers and assigning cell types to clusters [Butler et al., 2018].

With this tool and the operations it provides, it is possible to tackle a problem that was previously very predominant in the area of genome analysis: the lack of existing methods that enable integrate or comparative analysis of different scRNA-seq datasets consisting of multiple transcriptomic subpopulations, either to compare heterogeneous tissues across different conditions, or to integrate measurements produced by different technologies.

A recent paper on this tool explores in detail this issue, creating a successful solution and proving its efficacy through various different analysis. It serves as a quality demonstration on the limits of this tool and its potential for future research applications [Butler et al., 2018].

Still regarding this tool, another study explores how it is possible to use it for comprehensive integration of single-cell data. A method for integrate and compare single cell measurements through the anchoring of various different datasets was developed. More specifically, by identifying cell pairwise correspondences between single cells across datasets, it is possible to transform said datasets into a shared space, even in the presence of extensive technical and/or biological differences. This technique enables the transcriptome-wide prediction of spatial expression patterns, and the harmonization of scRNA-seq derived cell-type labels with in situ gene expression datasets [Stuart et al., 2019]. An illustration of this technique can be seen in Figure 2.8.

2.5 Tools

The evolution of the above analysed areas in the last decade wouldn't be possible without the associated advent of new tools that allowed the pushing of knowledge boundaries. In this section

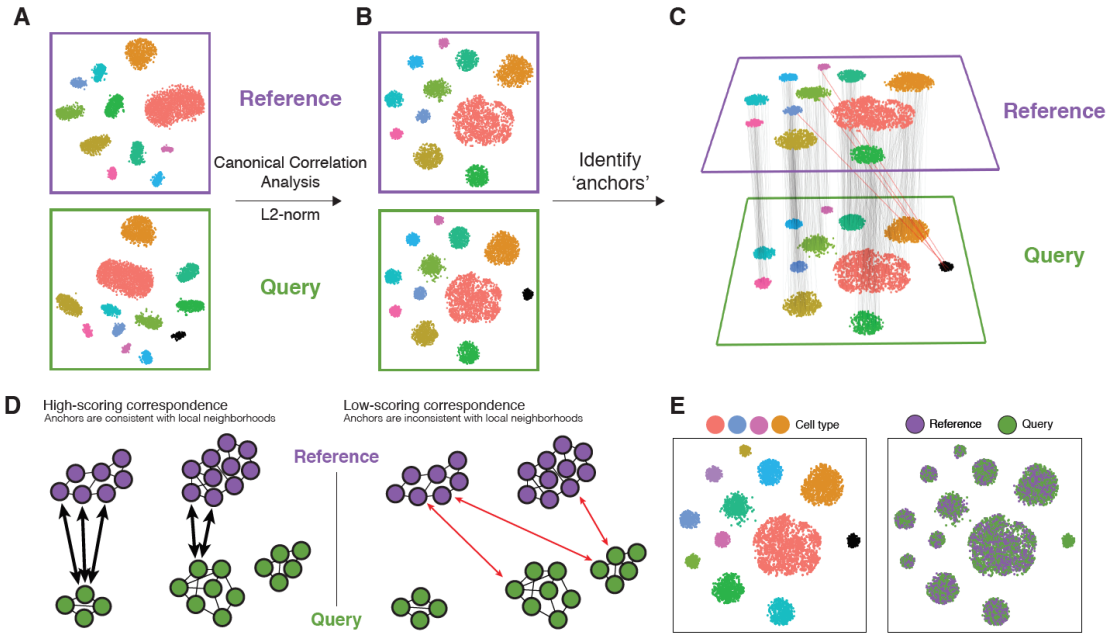


Figure 2.8: Overview of the Seurat dataset integration method [Stuart et al., 2019]. **(A)** Two different datasets from the same single cell experiment, with a unique population (in black). **(B)** Canonical correlation analysis and L2-normalization are applied. **(C)** Pairs of mutual nearest neighbours are identified in the space shred between the two datasets. **(D)** A consistency score is applied to each pair. **(E)** Scores are used to create correction vectors for each query cell [Stuart et al., 2019].

we study the more prominent and widely used tools in these areas. We divided this section in two parts, Data Mining tools and Databases.

2.5.1 Data Mining Tools

The more prominent tools used for Data Mining are the following:

- **Rapidminer**: a data science software platform that provides an integrated environment for data preparation, machine learning, deep learning, text mining, and predictive analytics. One of the more complete available solutions currently available, with graphical analysis, availability of extensions, among other features. Its advanced graphical interface allows users to perform data mining operations without any prior programming knowledge [Jović et al., 2014].
- **WEKA (Waikato Environment for Knowledge Analysis)**: a open source collection of visualization tools and algorithms for data analysis and predictive modeling, together with graphical user interfaces for easy access to these functions. Since it is written in Java, it has great integration potential and versatility [Jović et al., 2014].
- **R**: an open source tool and programming language, mostly optimized for matrix based calculations and statistics. It presents plenty of versatility through the available extensions.

It is not, however, a user-friendly tool, offering just a simple graphical interface and a command line for input, requiring the user to learn the R language in order to use it. Because of this, the language's full potential is difficult to master and the learning curve is steep [Jović et al., 2014]. There are various packages available that focus on bioinformatics and related data. Of particular interest is the Seurat package [Lab, 2020], specialized in single-cell RNA-seq data treatment and analysis, one of the most relevant and innovative tools recently developed in the area [Jović et al., 2014].

Other relevant and widely used tools are:

- **Orange**: a tool based on Python that can be used as a plug-in, through scripting or through its graphical user interface. Among other features, it offers data operations, classification, analysis and unsupervised learning. One of its downsides is the lack of integration with WEKA [Jović et al., 2014].
- **KNIME (Kontaz Information Miner)**: a tool based on the Eclipse IDE that provides a graphical interface similar and workflow similar to Rapidminer. It is highly extendable and can be integrated with WEKA and R, among other possibilities [Jović et al., 2014].
- **scikit-learn (Python)**: a Python package that provides various Data Mining algorithms. Since it is a community based tool, it is constantly improving and gaining new features. One of its advantages is its function-based methods and performance. However, this tool requires the user to be profficient in the Python language in order to use it [Jović et al., 2014].
- **Keras (Python)**: a high-level API for neural networks, written in Python. It allows the development and evaluation of deep learning models [Jović et al., 2014].

Along the aforementioned tools, some recent breakthroughs allowed the creation of some specialized tools, such as:

- **OligoMacro**: an ImageJ macro-toolset aimed at isolating oligodendrocytes from wide-field images, tracking isolated cells, characterizing processes morphology along time, outputting numerical data and plotting them [Azevedo et al., 2018].

2.5.2 Databases and Web Repositories

Conducted research from various studies provided various databases, web repositories and assorted tools containing relevant information related to this topic. Some of these resources represent major efforts in archiving gene data collected through the years, or novel ways to analyse existing data. Others serve as valuable tools that aid in the process of gen enrichment analysis i.e. the biological characterization of a set of genes or proteins, by searching for functional categories, classes, attributes and any other types of data that are over-represented in the set, and may have an association with disease phenotypes.

Bellow are some relevant examples:

- **Gene Ontology**: one of the main available resources for enrichment analysis on gene sets. A platform that allows the consultation regarding genes and their functions, attributes, products, cataloguing, etc. It provides tools for easy access, search and consulting of the information of a gene, as well as a personalized annotation system [Harris et al., 2008].
- **DAVID (Database for Annotation, Visualization and Integrated Discovery)**: a group of tools, including a custom algorithm to measure similarity between genes. The aim of this platform is to condense a list of genes or associated biological terms into organized classes of related genes or biology, called biological modules [Huang et al., 2007]. It agglomerates species-specific gene/protein identifiers from a variety of public genimic resources including NCBI, PIR and Uniprot/SwissProt [Marques et al., 2016].
- **KEGG (Kyoto Encyclopedia of Genes and Genomes)**: collects data regarding genomes, diseases, chemical components and biological pathways at a molecular-level. Deals specially with data generated by genome sequencing and other high-throughput experimental technologies [Khatri et al., 2004].
- **GenBank**: a publicly available database of nucleotide sequences from various species, updated every two months [Khatri et al., 2004].
- **ENSEMBL**: a group of various resources that come together to characterize the human genome. These resources include comparative genomics, genetic trees, various information regarding nucleotide sequences, among others [Khatri et al., 2004].
- **ArrayExpress**: an online repository of functional genomics data [Khatri et al., 2004].

2.6 Summary

As we can see through the aforementioned referenced studies and collected data, a multidisciplinary approach between the areas of Neuroscience, Data Mining and Bioinformatics is encouraged, in order to fully take advantage of the latest technological developments and enhance future contributions to academic research.

In fact, advancements in the two areas complement each other. Breakthroughs in Data Mining allow for different approaches and more in-depth analysis to Neuroscience data, while these processes present themselves as opportunities to test the practical limits and reach of Data Mining and Bioinformatics tools, and further augment them.

Chapter 3

Methodologies and Architecture

3.1 Introduction

This chapter addresses all the tools and methods employed and attempts to explain their inner workings, reason of use and the way they were applied, as well as the structure of the created solution.

3.2 Methodologies

The methodologies applied in this paper were largely inspired by the two previously mentioned papers this report was based on. Additionally, some of the tools mentioned in section 2.5 were utilized. More specifically, Python, R and its interface RStudio and various packages such as Seurat and Clustree and the Flask framework along with the Flask What The Forms package.

Among these tools, one appeared to be particularly promising: Seurat [Lab, 2020]. This is due to its previous use in the papers mentioned above.

Through RStudio, the default scRNA-seq data processing pipeline provided by Seurat was applied to the dataset produced in [Marques et al., 2016] with the aim of reproducing the results in this paper so that further analysis regarding genes relevant to the cytoskeleton of oligodendrocytes could be conducted. More precisely, we aimed at replicating the same number of clusters and the cluster tree seen in [Marques et al., 2016]. By defining a group of genes that results in a similar cluster tree, we can conduct further analysis on the dataset while guaranteeing continuity between this study and the [Marques et al., 2016] study.

Additionally, a web application was developed so that users with no previous knowledge of these tools can use them and visualize results. This tool's value resides in the centralization of two resources: the Seurat pipeline, allowing for data processing of datasets originated by scRNA-seq and gene enrichment/GO analysis, the exploratory analysis of gene sets. This app applies the default

Seurat pipeline to a gene expression matrix according to the configurations provided by the user, and allows the user to originate GO analysis reports through the DAVID tool.

3.2.1 Seurat Pipeline

The versions of the software used in this project can be seen in Table 3.1.

Table 3.1: Versions of the tools used.

Software	Version
Python	3.8.2
R	3.6.3
RStudio	1.2.5042
Seurat	3.1.5
clustree	0.4.3
rpy2	3.3.3
RapidMiner Studio	9.6.0
DAVID	6.7
Flask	1.1.2
Firefox Browser	77.0.1

Seurat takes a gene expression matrix that can be in various formats, including 10xGenomics matrices. A default pipeline was used. This serves as a generic way of treating scRNA-seq data allowing for a general visualization and of the data and possible more specific treatment. A gene expression matrix was used, taken from GSE75330. This gene expression matrix was obtained through cells isolated from ten distinct regions of the anterior-posterior and dorsal-ventral axis of the mouse juvenile and adult CNS, including grey matter (spinal cord/dorsal horn, substantia nigra and ventral tegmental area (SN-VTA), amygdala, hypothalamic nuclei, zona incerta, hippocampus/dentate gyrus and CA1, and somatosensory cortex), white matter (corpus callosum) or both (striatum), and also adult CNS (somatosensory cortex, corpus callosum and dentate gyrus). After being isolated and treated, the cells were FACS sorted, analysed through a microscope and subject to quality control tests. Clustering of the cells was made with BackSpinV2. After further processing and filtering, 5072 cells grouped in 13 clusters were obtained [Marques et al., 2016]. The different populations of OLs found in the dataset can be seen in Figure 2.6.

Seurat revolves around the *SeuratObject* class. This object is created from a gene expression matrix. Every time a certain method is applied to the data, information is generated and stored in the object. This allows flexible application of successive operations and for easily saving the object produced in a session for later use.

Firstly, the data is read and the *SeuratObject* is created. Then a normalization operation is applied. The default method normalizes the feature expression measurements for each cell by the total expression, multiplies this by a scale factor (10,000 by default), and log-transforms the result.

Afterwards, a process of feature selection is performed. This aims to calculate a subset of genes (referred to as "features") that exhibit high cell-to-cell variation in the dataset. Seurat's default feature selection method is described in [Stuart et al., 2019] At this point, the top most highly variable genes can be selected and saved and plots for the most expressed genes can be computed.

After feature selection, operations for scaling the data are applied to remove variation in the dataset. More specifically, a linear transformation method is applied, more specifically Principal Component Analysis (PCA). PCA is a dimensionality-reduction method that is often used to reduce the dimensionality of large data sets, by transforming a large set of variables into a smaller one that still contains most of the information in the large set. Reducing the number of variables of a data set comes at the expense of accuracy; it enhances, however, simplicity of the dataset, making it easier to explore, visualize and conduct further analysis [Jaadi, 2020].

Seurat's default PCA method shifts the expression of each gene, making the mean expression across cells equal to 0 and scales the expression of each gene, making the variance across cells equal to 1. Scaling can be performed on all features or only on previously selected features. The results are stored in the *SeuratObject*.

The next step is the application of a linear dimensional reduction method. This is achieved through Seurat's principal component analysis function. Each PC essentially representing a 'metafeature' that combines information across a correlated feature set, and thus represent a compression of the dataset. The results are stored in the *SeuratObject*. To visualize the resulting principal components, various options are available: dot plots to compare two or multiple components and heat maps to visualize a singular or multiple components.

This step is usually followed by the determination of the dimensionality of the dataset. Seurat offers three ways to achieve this. The first focuses on supervision, determining sources of heterogeneity through PC analysis. The second, more time-consuming, is based on a statistical test on a random null model. The third, a heuristic method. Depending on the needs of the user or the nature of his dataset, the most appropriate method can be applied. Additionally, Seurat uses a custom algorithm based on the JackStraw method. It randomly permutes a subset of the data, re-applies PCA, computes a distribution of feature scores, and repeats this procedure. The results are stored in the *SeuratObject*. A method to visualize the results of this operation is provided. Alternatively, a method to view a ranking of the PCs based on the percentage of variance is also available.

Thus we reach a crucial phase: clustering the cells of the dataset. Seurat has a custom approach to clustering, based on graphs. Starting from a group of preciously specified number of features, cells are embedded in a graph (i.e. KNN) and edges are drawn between cells with a similar feature expression pattern. The weight of the edges is calculated through the overlap between the

surroundings of each cell. The graph is then divided into small groups.

Modularity optimization techniques such as the Louvain algorithm are then applied to iteratively group cells together. In this step, it is possible to adapt the resolution of the operation in order to calculate a decreased or increased number of clusters. Clustering information, including number of clusters and resolution of each application of the algorithm is saved in the *SeuratObject* and can be consulted anytime.

After the clustering process, non-linear dimensional reduction is applied. This process maps the high-dimensional space into a low-dimensional embedding in order to facilitate visualization of the dataset.

Seurat offers two main methods: tSNE and UMAP, among others. Both these methods aim to achieve similar results, tSNE being the older, more standard method.

The role of tSNE is to help the visualization of high-dimensional data by projecting it into a low-dimensional space [Kurita, 2018], mainly through utilizing the local relationships between points to create a low-dimensional mapping.

The more recent UMAP has, however, a few advantages that make it a great new way to map data. Not only it has a superior performance, it also does a better job at preserving the global structure of the dataset and the relations between clusters while having no restrictions on the dimensions of the dataset. Though being mostly an improvement, UMAP also has its limits, namely being unable to separate two nested clusters in a scenario where a dense, smaller cluster is inside a larger, sparser cluster [Andy Coenen,].

Users can thus change their preferred algorithm and tune the method by specifying certain input variables like dimensionality, subset of genes, number of neighboring points to use, etc. The results are stored in the *SeuratObject* and can be visualized through the available plotting methods.

For further analysis, Seurat allows users to find gene markers that define clusters through expression. It allows for the identification of positive or negative markers in a single cluster, in all clusters, or in different groups of clusters or cells, compared against each other. With this method an user can, for example, identify the genes that lead to differentiate two clusters, or identify genes that are common to all clusters [Butler et al., 2018]. It is also possible to visualize markers through violin, dot, ridge and scatter plots, among other formats. Regarding this paper, the gene marker methods were not used.

After treating the data in Seurat and replicating the results in [Marques et al., 2016], the data was introduced in DAVID for a GO analysis, in order to identify genes specific to the cytoskeleton and associated characteristics that could be of interest to the topic of this paper.

Through DAVID, a functional annotation chart, clustering sheet and table with detailed information about certain genes of interest was obtained.

3.2.2 DAVID

Due to its use in [Marques et al., 2016] for gene enrichment analysis, providing an Application Programming Interface (API) for Web integration and being on of the most complete GO analysis tools available, this tool was chosen in order to apply GO analysis to the dataset in this report.

DAVID (Database for Annotation, Visualization and Integrated Discovery) consists of an integrated biological knowledgebase and analytic tools aimed at systematically extracting biological meaning from large gene/protein lists.

With DAVID, an user can identify the genes in a dataset that are enriched, i.e. present abnormal, usually higher than normal expression values. By identifying these genes and the associated cellular components and processes, the user can withdraw knowledge and conclusions regarding the dataset.

This detection is made through a background comparison with known expression for the genes in question that are considered normal [Huang et al., 2007].

To use this tool, the user uploads a gene expression matrix (the genes can be under any official format), chooses what type of report is to be generated and analysis the provided report in an explorative manner.

The available reports are:

- **DAVID gene functional classification:** provides the distinct ability for investigators to explore and view functionally related genes together, as a unit, to concentrate on the larger biological network rather than at the level of an individual gene
- **DAVID functional annotation chart:** provides typical gene–term enrichment (overrepresented) analysis
- **DAVID functional annotation clustering:** uses a similar fuzzy clustering concept as functional classification by measuring relationships among the annotation terms on the basis of the degree of their co-association with genes within the user's list to cluster somewhat heterogeneous, yet highly similar annotation into functional annotation groups
- **DAVID functional annotation table:** a query engine for the DAVID knowledgebase, without statistical calculations; for a given gene list, the tool can quickly query corresponding annotation for each gene and present them in a table format; thus, users are able to explore annotation in a gene-by-gene manner [Huang et al., 2007]

An illustration of the DAVID workflow and available features can be seen in Figure 3.1.

3.2.3 Web Application

A small web application was developed. With this tool, a user can upload a configuration file in the JSON format that specifies which steps of the previously explore Seurat pipeline are to be applied to the data, saves the produced charts and additional information and displays them in a user-friendly manner. DAVID analysis reports can also be generated and saved locally.

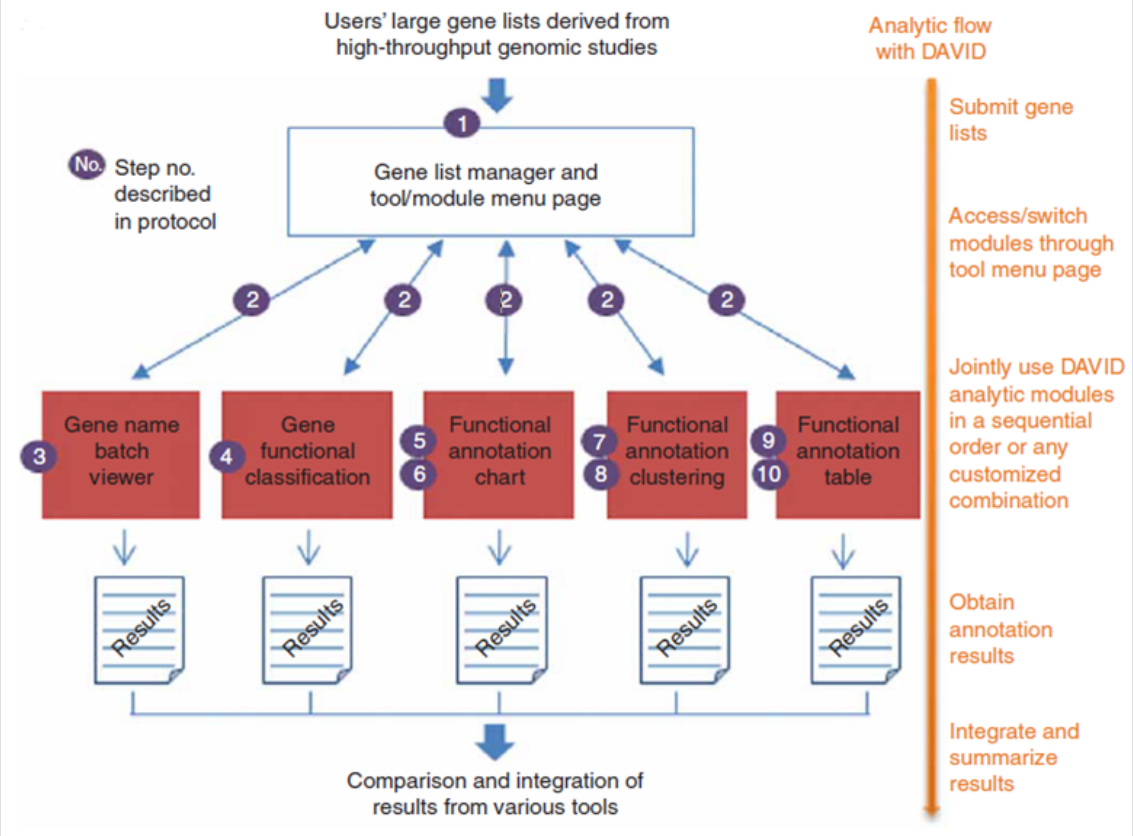


Figure 3.1: DAVID workflow and features [Huang et al., 2007].

3.3 Architecture

The web application was made with Python. The bridge between Python and R was accomplished through rpy2, a pyhton package running embbeded R that serves and an interface between the two languages, providing access to R methods through Python objects. The Seurat R script was converted and adapted into a Python script that accepts a JSON file, where the user defines what steps of the Seurat workflow are to be applied to the data. Afterwards, a small website was creating using Flask, a micro web framework written in Python that provides a easy way to bridge Python code and HTML. Form validation and rendering was made resorting to the tools provided by by the WTForms library, through the intermediary package Flask WTF. DAVID integration is achieved through the former's Python Web API. An illustration of the architecture of the web app can be seen in Figure 3.2.

The actions the user can take are illustrated in Table 3.2 and Figure 3.3.

Table 3.2: Web application User Story table.

As...	I want...	So that...
a User	to upload a JSON configuration file	I can define the data to be analysed, according to my preferences
a User	select the type of analysis to be done	I can search for a specific kind of information regarding the data I have
a User	view the results of the conducted analysis	I can identify and obtain useful information regarding the data I have

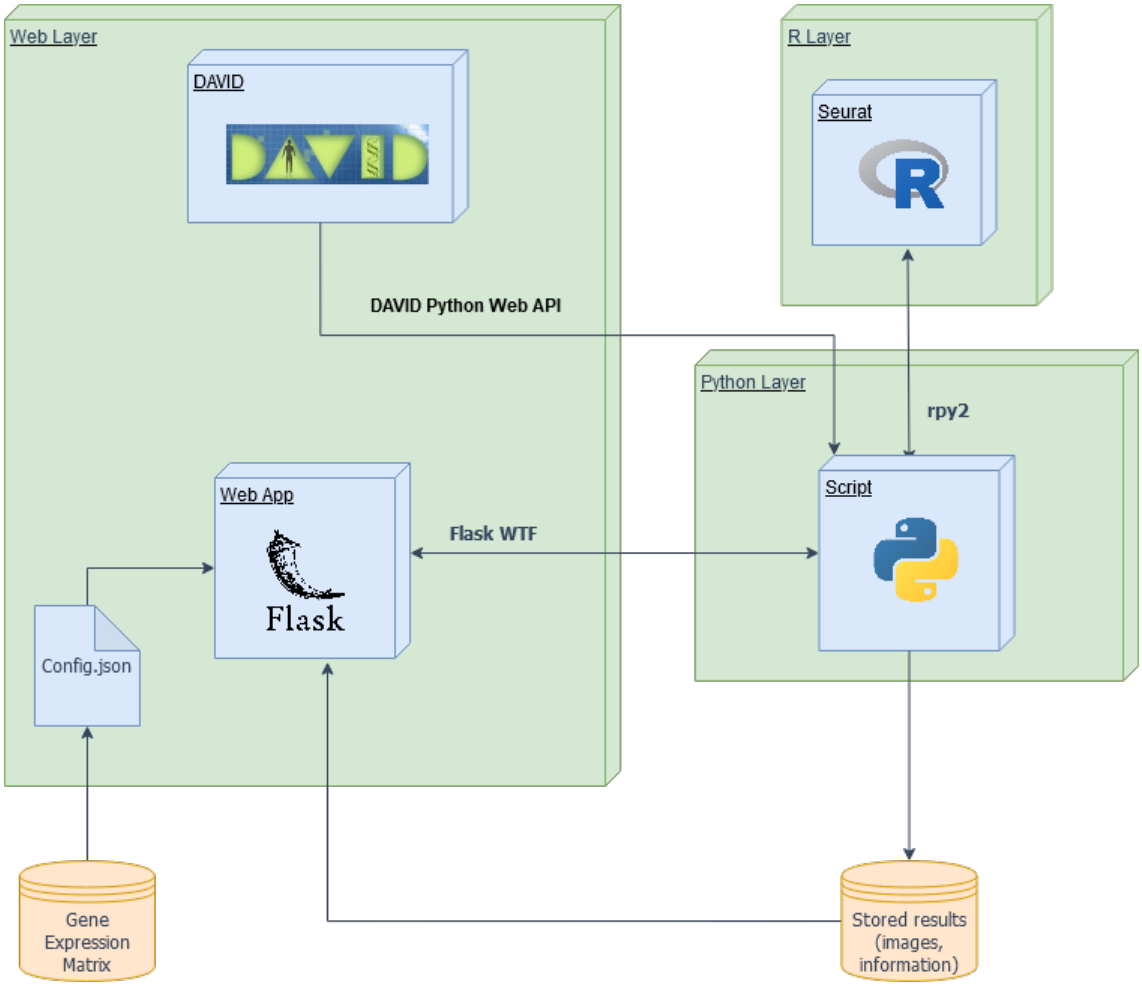


Figure 3.2: System architecture diagram.

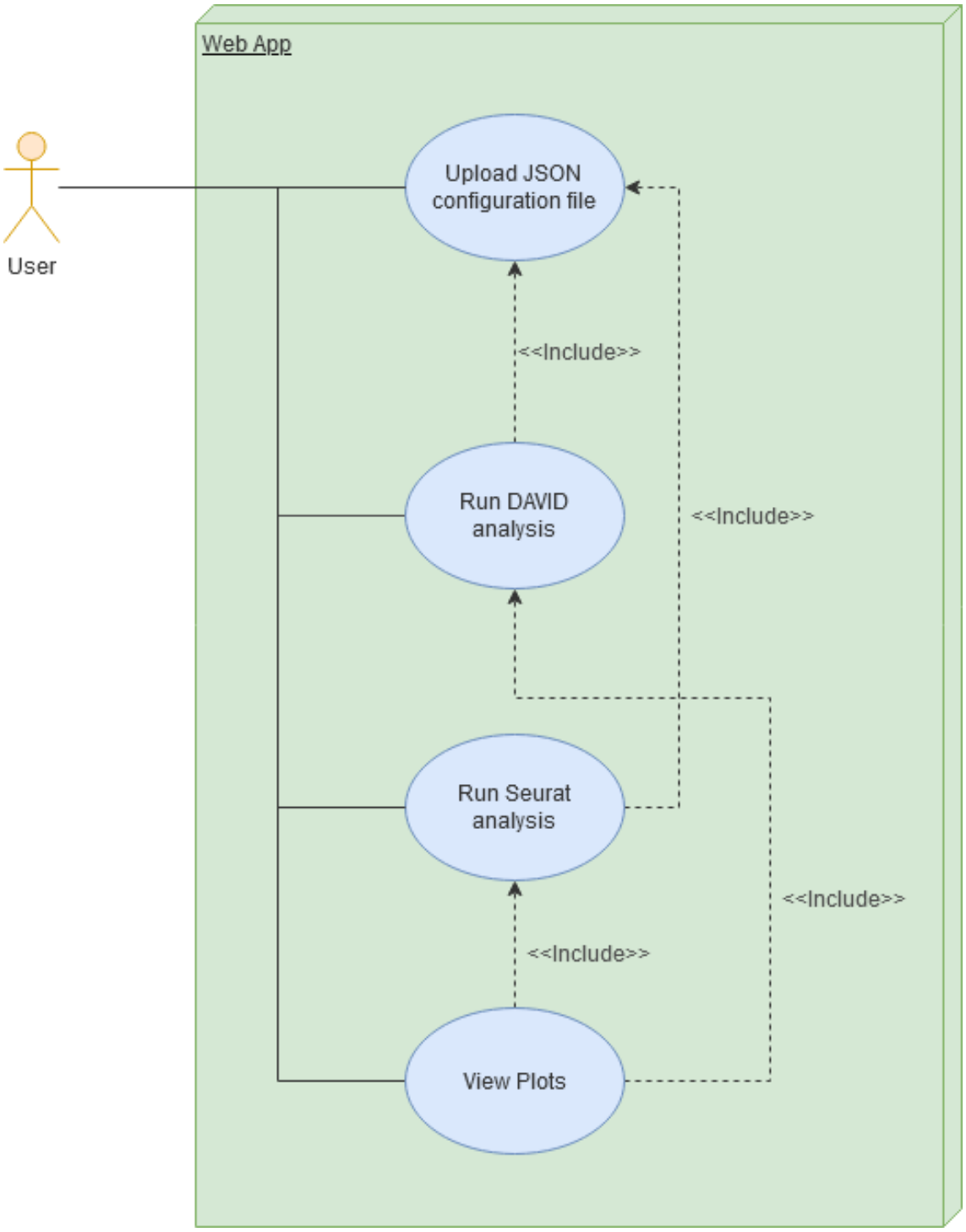


Figure 3.3: Use cases diagram.

Chapter 4

Case Study

4.1 Introduction

This chapter details how, through the aforementioned methods, our results were achieved.

4.2 Seurat

The aim of this analysis was to replicate the results in this paper using methods from the Seurat library, testing its limits, and from those results conduct further analysis, in the search for new knowledge regarding the cytoskeleton of neural cells and related genes.

As previously mentioned, the Seurat pipeline was used on data taken from [Marques et al., 2016]. This dataset consists of a gene expression matrix in the TAB format was used (taken from GSE75330). This data consists of 5069 transcriptomes of single oligodendrocyte cells from spinal cord, substantia nigra-ventral tegmental area, striatum, amygdala, hypothalamic nuclei, zona incerta, hippocampus, and somatosensory cortex of male and female mice between post-natal day 21 and 90. Cells were sampled from CNS regions of mice of various strains.

The *SeuratObject* is created with this data (*CreateSeuratObject* method), with all the parameters set to their respective default values. This object is the core of this tool. With every operation made, metadata is produced and stored in the object. This mechanic facilitates saving the chain of operations applied, by saving the *SeuratObject* and uploading it in order to continue the analysis.

Since the dataset is already pre-processed, no further methods are applied. A representation of the data in this state can be viewed in Figures 4.1 and 4.2.

Data is normalized logarithmically (*NormalizeData* method) with a scale factor equal to 10,000, with all the other parameters set to their respective defaults. Normalizaed data is represented in Figure 4.3.

Scaling is then applied (*ScaleData* method) to all genes of the dataset and other parameters set to their defaults.

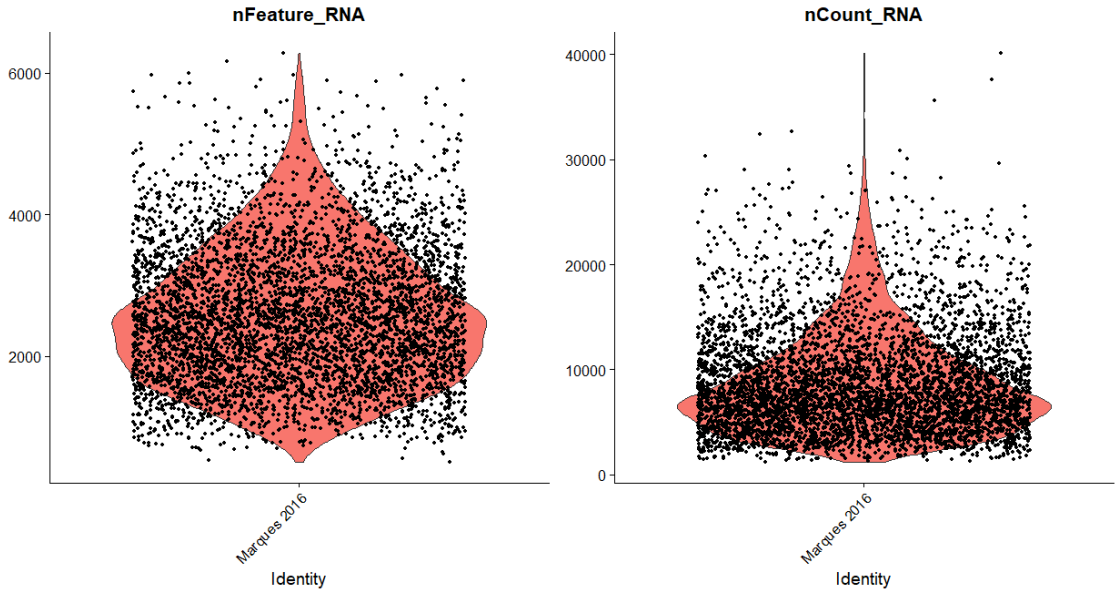


Figure 4.1: Violin plot showing the relationship between features and RNA counts.

Linear dimensional reduction is performed resorting to PCA (*RunPCA* method), with all the parameters set to their defaults. Plots were produced to visualize some of the PCs of the dataset, displayed in Figures 4.4, 4.5 and 4.6.

To determine the dimensionality of the dataset, Seurat's custom version of the Jackstraw algorithm is applied (*JackStraw* and *ScoreJackStraw* methods), with the number of replicate samplings to perform (num.replicate) set to 500, in order to increase accuracy. It's possible to increase performance by decreasing this number. This method assigns a p-value for each gene's relation with eac PC. A PC is considered to have a strong enrichment when the associated p-values have low values. This information is stored in the SeuratObject, and can be visualized in Figures 4.7 and 4.8.

Cell clustering was made with resorting to Seurat's *FindNeighbors* and *FindClusters* methods, with the k number (k.param parameter of the *FindNeighbors* method) set to 66 in order to reach the 13 clusters found in [Marques et al., 2016]. This number was reached through trial and error, and is noticeably high. This is because clustering is applied 5 times with 5 different resolution values (0, 0.1, 0.5, 0.8 and 1 with the k.scale parameter set to 25) in order to gather enough metadata to generate a cluster tree through the Clustree library methods. The information regarding each clustering application is saved in the *SeuratObject*, where a column with each resolution is added to the meta data.

Despite having reached a similar number of clusters as in [Marques et al., 2016], pairing each OL population from this paper to a corresponding cluster from the Seurat results is yet to be done. A suggested method to achieve this is through the search for potential patterns in the levels of expression of relevant marker genes from each population that could also be found in the anonymous clusters.

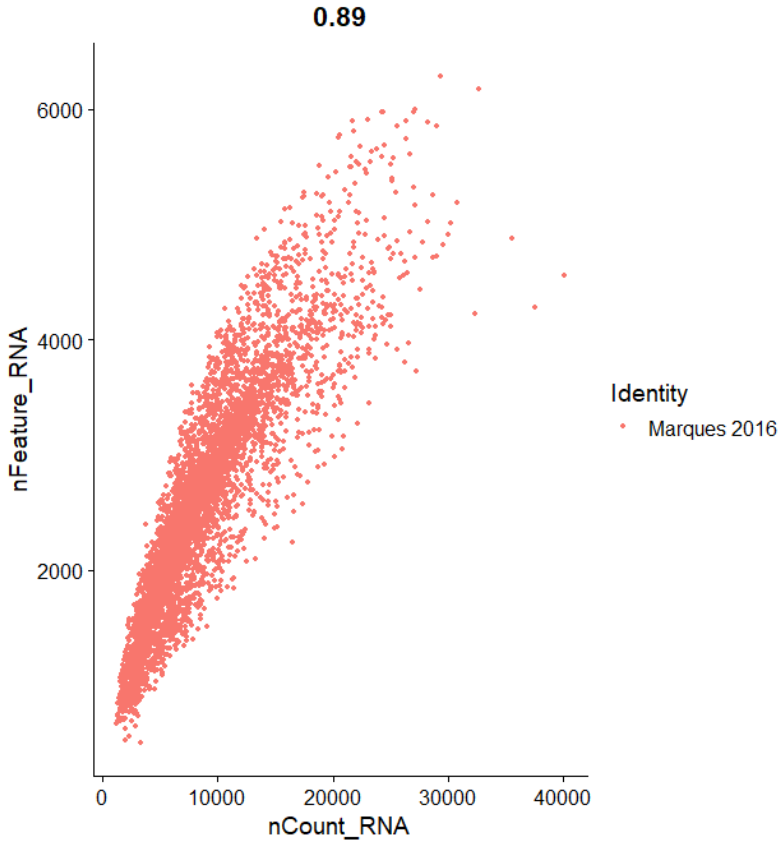


Figure 4.2: Scatter plot showing the relationship between features and RNA counts.

Non-linear dimensional reduction is performed with Seurat's *RunUMAP* method, which employs the UMAP algorithm, with all parameters set for their default values. The result of this operation can be seen in figure 4.9.

To further explore the dataset, the gene markers which were common to all clusters were calculated. To achieve this, first the markers of each cluster were calculated with Seurat's *FindMarkers* method. Afterwards, they were re-organized in a decreasing order, according to their gene expression and stored in tables. The resulting matrices were then intersected and plots computed. The results from this process can be seen in Figures 4.10 and 4.11.

Additionally, two cluster trees using two different methods were computed, for comparing purposes. The aim here was to match the cluster tree after processing the data in Seurat to the one obtained in [Marques et al., 2016].

In order to build a cluster tree, a subset of genes from the dataset was obtained. The process employed to achieved this was the following:

- Calculate the Average Gene Expression of the dataset; this was achieved through Seurat's *AverageExpression* method
- Find the differentially expressed genes of each cluster (i.e. the gene markers that define the cluster); this was achieved through Seurat's *FindMarkers* method

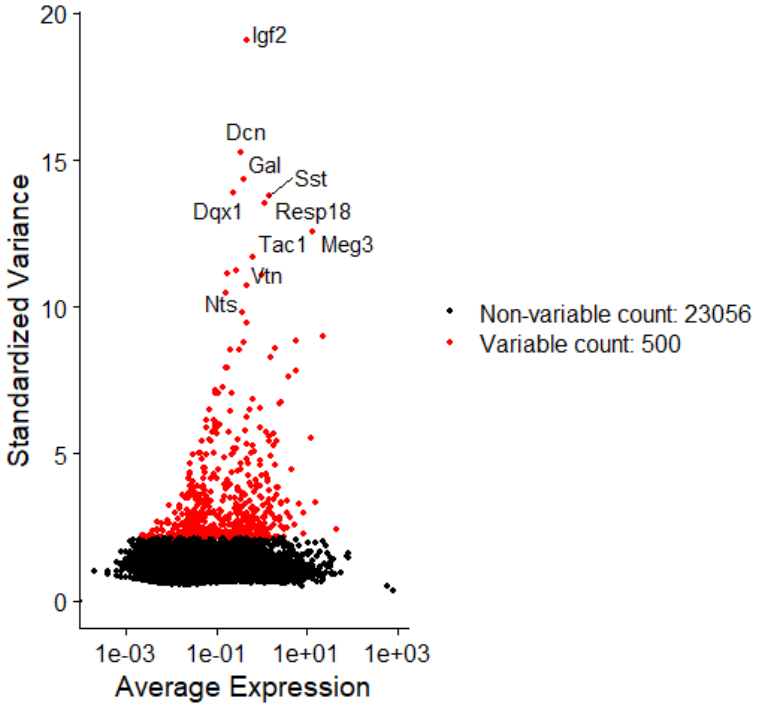


Figure 4.3: Top 10 most expressed features.

- Select the top 100 genes with highest expression, for each cluster

The first cluster tree was computed using Seurat's native BuildClusterTree function, with all parameters set to their default. The result can be seen in Figure A.1.

The second tree was built resorting to the Clustree R package, which offers a more in depth way to visualize cluster trees. The result obtained was the same and can be seen in Figure A.2.

After processing the data with Seurat, DAVID was used for further processing. More specifically, this tool was used to obtain information regarding GO and assorted enriched genes in the clusters. The version 6.7 was used. Starting from the previously saved ordered gene expression tables for each cluster, the top 500 most expressed genes for each clusters were selected and introduced in DAVID's Functional Annotation tool, with the identifier parameter set to "Official Gene Symbol" and the List Type parameter set to "Gene List". This number of genes was chosen since it is within the limits of this tool (it limits the number of genes to 3000 per analysis) while still covering the number of markers available in each cluster (only clusters 4, 8 and 11 fall short of 500 genes, having 416, 257 and 375 markers, respectively).

Additional filtering is applied to include only results regarding the *Mus Musculus* species.

To analyse the results computed by DAVID, special attention is given to the Functional Annotation Clustering data. This tab provides provides ordered enrichment scores for similar GOs. Through the study of the enrichment data it is possible to better understand the cellular functions associated to the more genes which are more expressed and enriched, for each cluster.

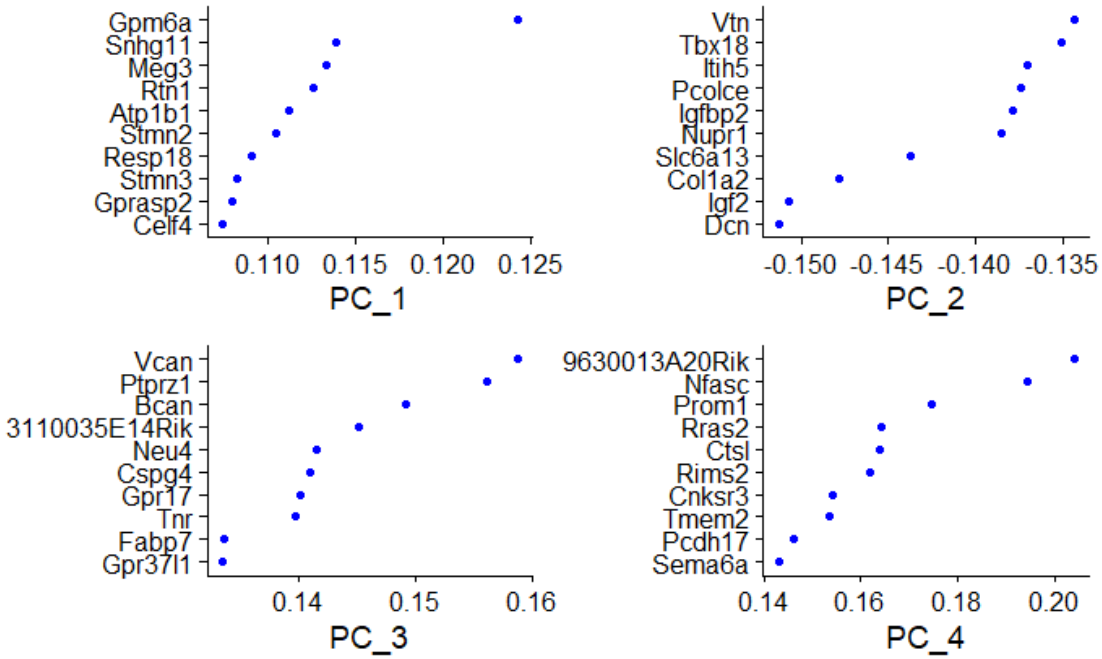


Figure 4.4: Top 10 most expressed genes for 4 PCs.

By searching in the data for certain keywords associated with the cytoskeleton (i.e. microtubule, tubulin, actin, myosin, etc.) it's possible to study specific information regarding this structure.

4.3 Web App

A simple web application was built, with the aim of creating a user friendly and immediate way to use Seurat and visualizing results.

The Seurat code is embedded in Python through rp2, a crossover package for the latter.

Flask was used as a easy way to build a web interface that could communicate with the Python scripts, with form validation being made with the Flask WTForms package.

The application takes as input a gene expression matrix and a JSON file with the user's preferences.

The gene expression matrix can be in any format that can be processed by Seurat (i.e. CSV, TAB, 10xGenomics, etc.)

The JSON file dictates what operations will be applied to the data and/or what information will be displayed. The format of the JSON input file can be seen in Figure A.3. The application uses Seurat to produce images similar to the ones that can be seen in this article.

To increasing this tool's utility, we intended to integrate it with DAVID's available web services. If this goal could be achieved, the process of scRNA-seq data treatment and gene enrichment

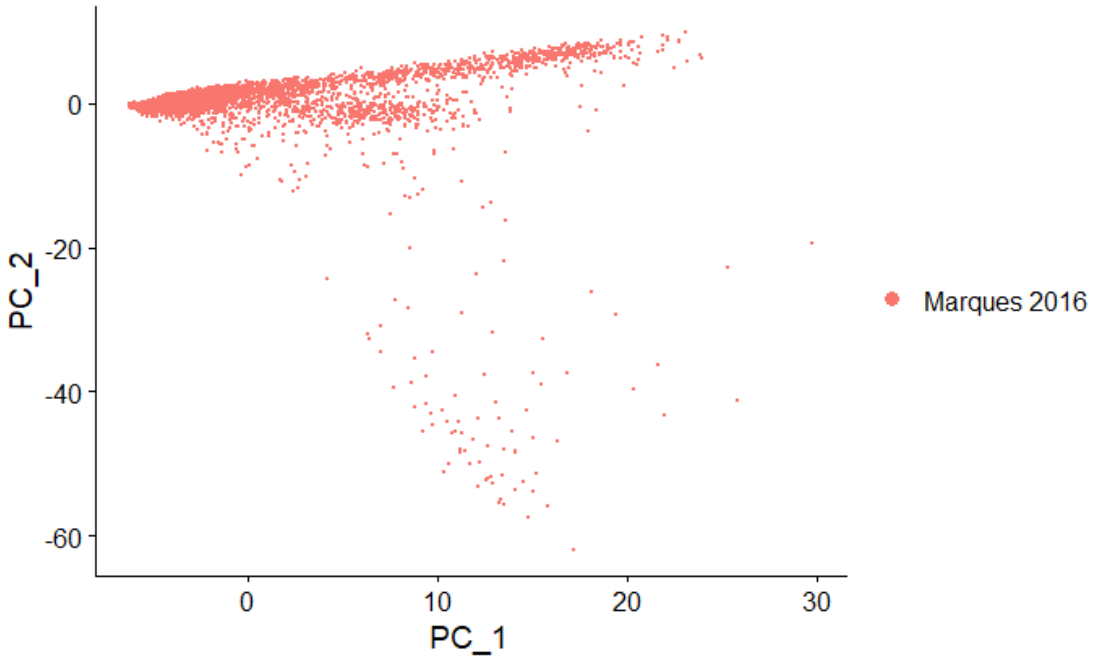


Figure 4.5: Comparison of the resulting dimensionality reduction between 2 PCs. Cells are colored by their identity class.

through GO could be centralized in the same application, a feature that could potentially be useful for researchers that are not familiarized with these tools.

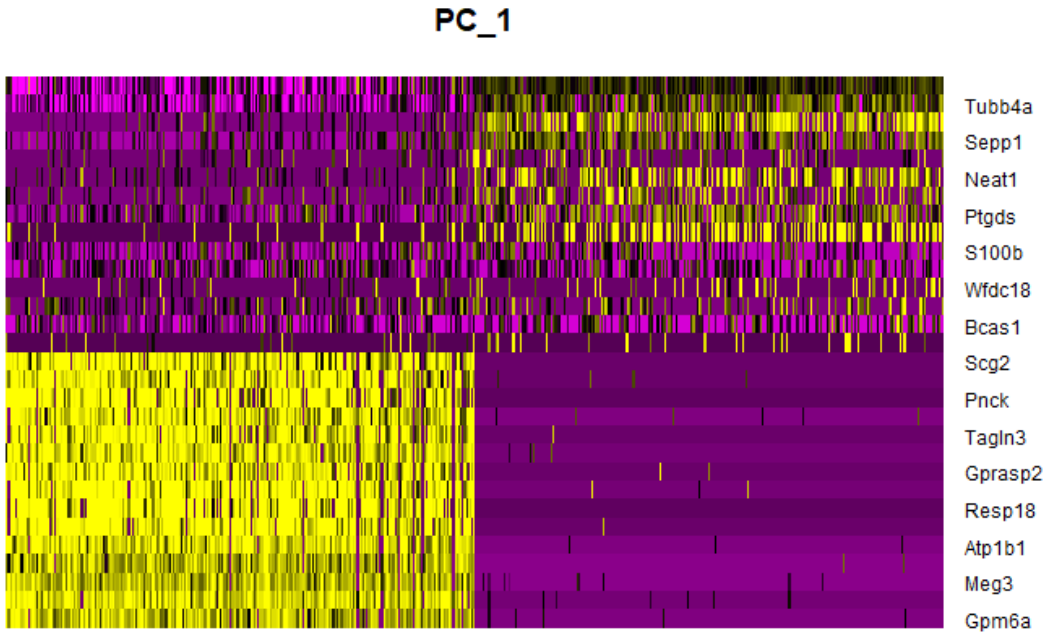


Figure 4.6: Heatmap comparing 500 genes of 1 PC.

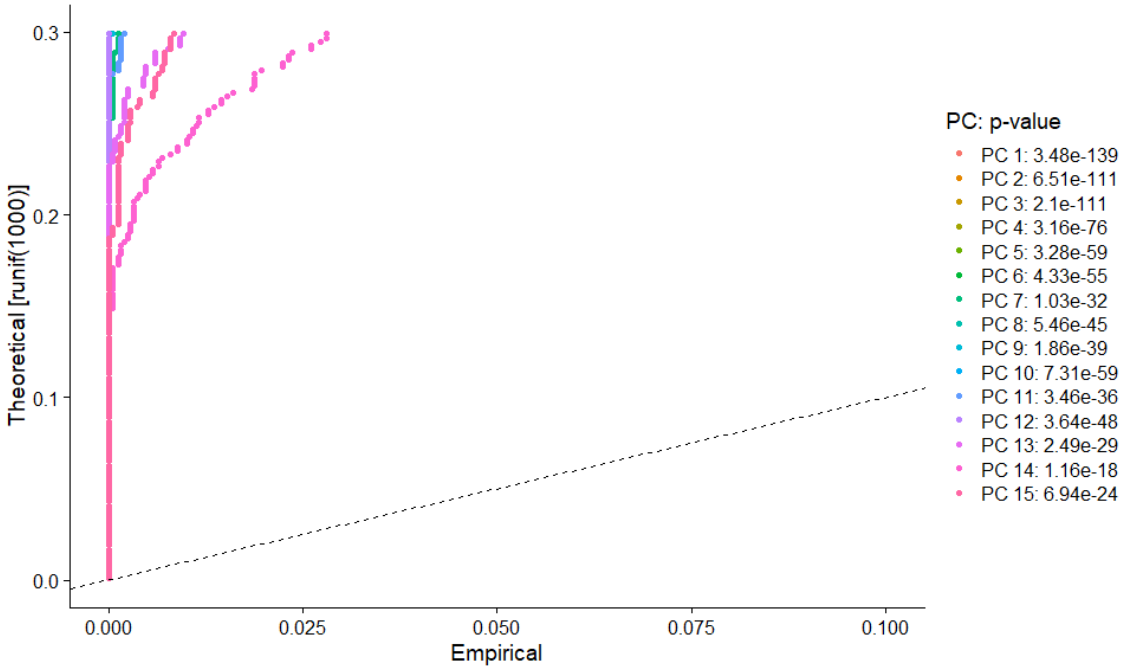


Figure 4.7: Data post Jackstraw application, showing the p-values for 15 PCs.

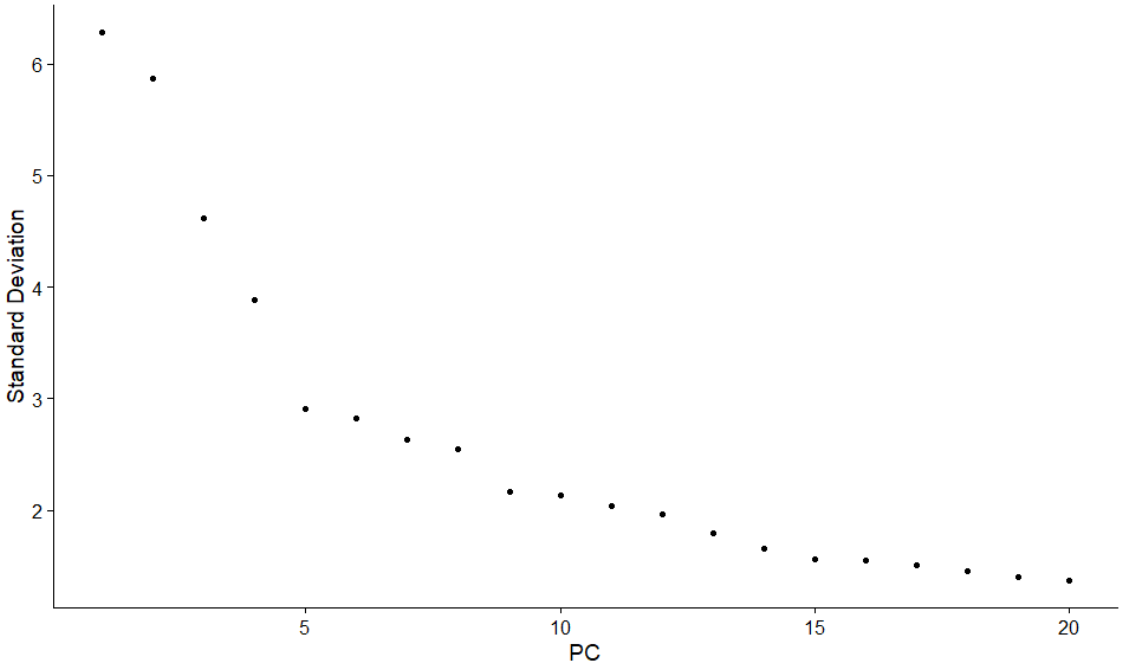


Figure 4.8: Elbow plot representing a ranking of PCs based on the percentage of variance.

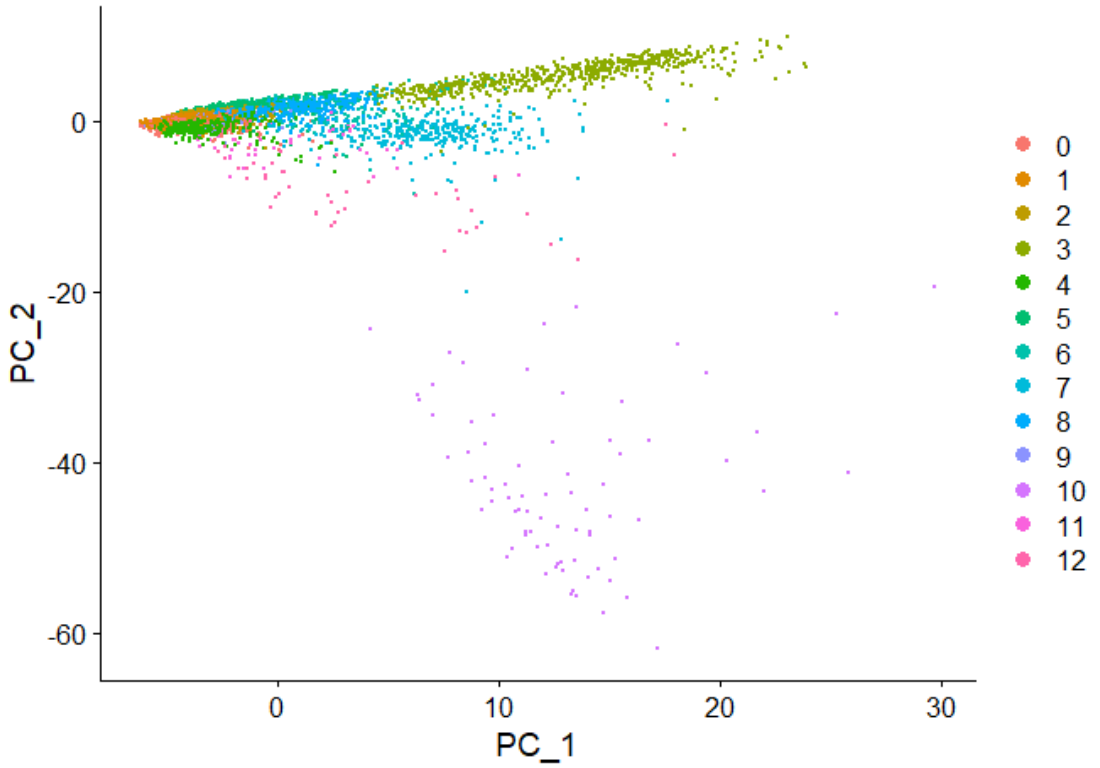


Figure 4.9: Visualization of the data post UMAP application. Comparison between 2 PCs. Each cluster is represented with a different color.

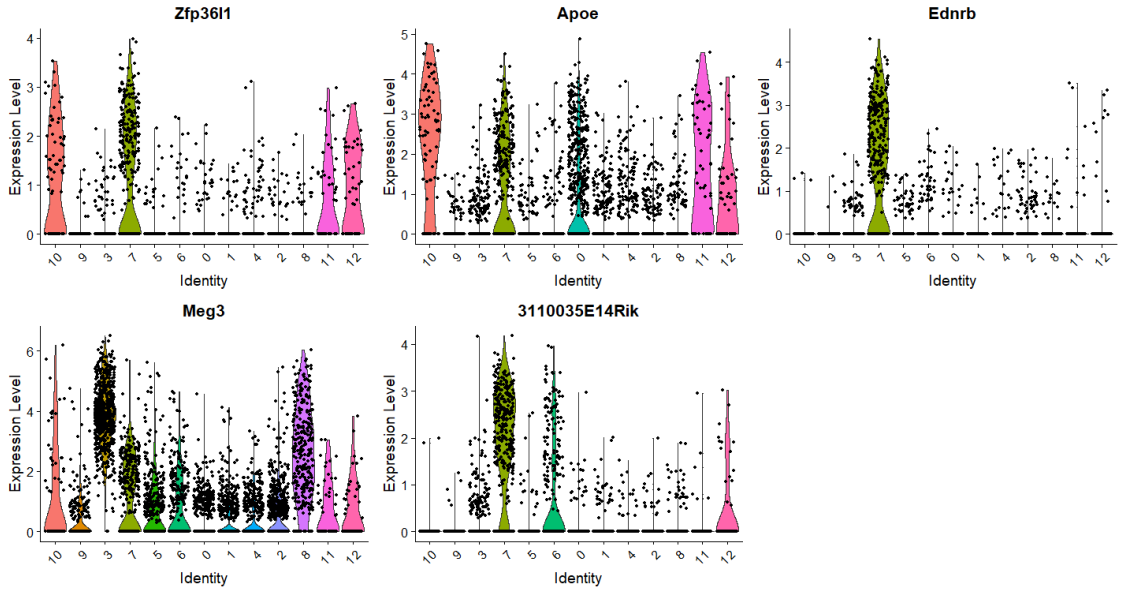


Figure 4.10: Violin plots representing the expression of each of the genes common to all clusters by cluster.

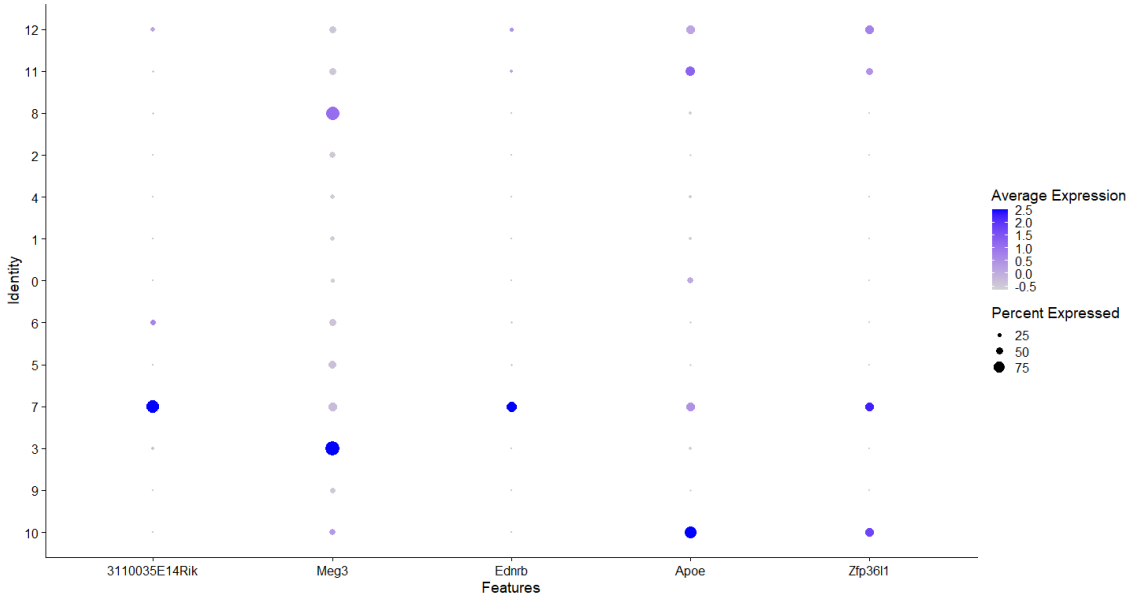


Figure 4.11: Plot comparing the average expression of each of the genes common to all clusters.

Chapter 5

Results

5.1 Introduction

In this chapter the results of this project are listed and discussed.

5.2 Overview

The initial goal of extracting new knowledge regarding the specific genes that differentiate each oligodendrocyte population was not achieved. The creation of an application that centralizes in a single resource scRNA-seq data analysis tools and GO tools was achieved.

5.3 Seurat

The Seurat package for Python was one of the main tools used in this project. The first task was to replicate the results observed in [Marques et al., 2016]. More specifically, we aimed to achieve a similar number of clusters and a similar cluster tree, starting from the same dataset, using the tools offered by Seurat.

Pre-processing, normalization, scaling, dimensional reduction and clustering were conducted successfully. The obtained number of clusters was the same as in [Marques et al., 2016].

However, the cluster tree could not be successfully replicated. While the obtained cluster tree presents similarities to the tree in [Marques et al., 2016], namely an accurate correspondence of the VLMC and OPC populations (matched to clusters 10 and 9, respectively), it isn't a total replica and therefore compromised, to some extent, the rest of the results. This is because the relation between the resulting clusters isn't guaranteed to be the same as seen in [Marques et al., 2016], and therefore the matching of oligodendrocyte populations can't be accurately done. Since we can't identify what state of the differentiation process each clusters corresponds, we can't draw accurate conclusions regarding the genes markers of each clusters.

It is notable, however, how the two approaches used to build the cluster tree (Seurat native methods and the Clustree methods) achieved the exact same results, as seen in Figures A.1 and

A.2. This implies that source of discrepancy in our results might be related to the genes that were selected to build the cluster tree, or in the way they were processed.

A comparison between the obtained tree and the tree originally obtained by [Marques et al., 2016] can be seen in Figure 5.1.

5.4 DAVID

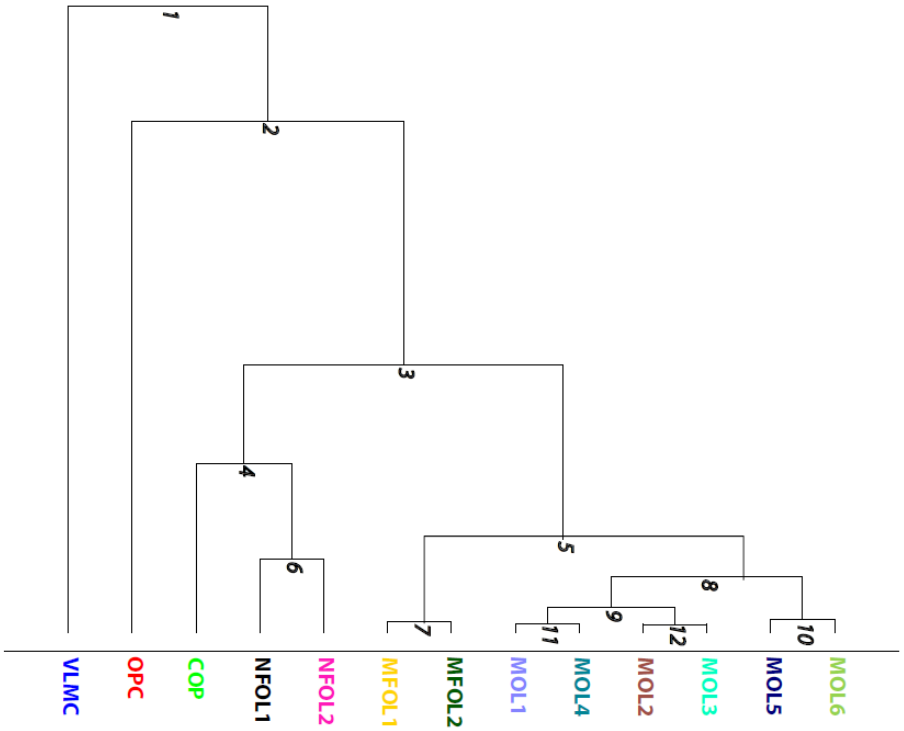
Since the originated cluster tree wasn't a perfect replica of the tree seen in [Marques et al., 2016], no accurate analysis could be done. However, DAVID analysis of the genes used to originate the cluster tree in this report resulted in the identification of some proteins and structures associated with the cytoskeleton.

5.5 Web App

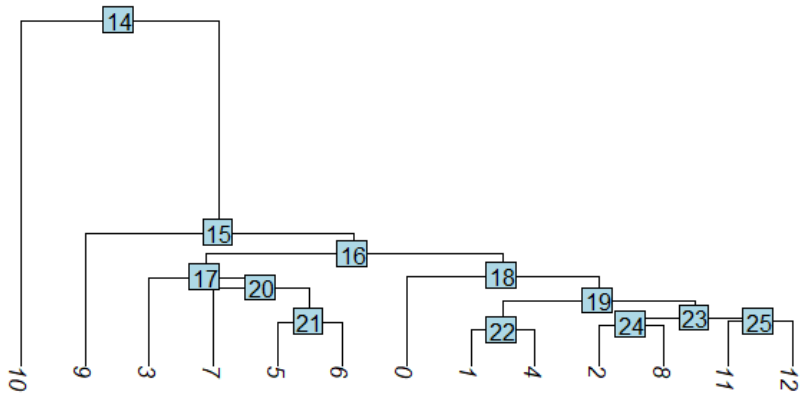
Although the developed web application is still in an embrionary state, it demonstrates how specialized tools such has Seurat and DAVID can be centralized in the form of a single, accessible resource. By connecting these two tools through Python, we preserve some of the flexibility required for future integration in other types of applications.

Additionally, since GO analysis is a very common method of analysis that is usually followed by the processing of RNA sequencing (as seen in [Marques et al., 2016] and [Falcão et al., 2018]) and these methods have been growing increasingly more important in RNA studies, the centralization of the two can potentially facilitate the work of researchers by streamlining the data analysis process as a whole.

An overview of the web app's interface can be seen in Figures 5.2 and 5.3.



(a) Original tree, where for each cluster a oligodendrocyte populations was identified.



(b) Obtained tree without any identified oligodendrocyte populations for the clusters.

Figure 5.1: Comparison between the structure of the obtained cluster tree and the original tree from [Marques et al., 2016].

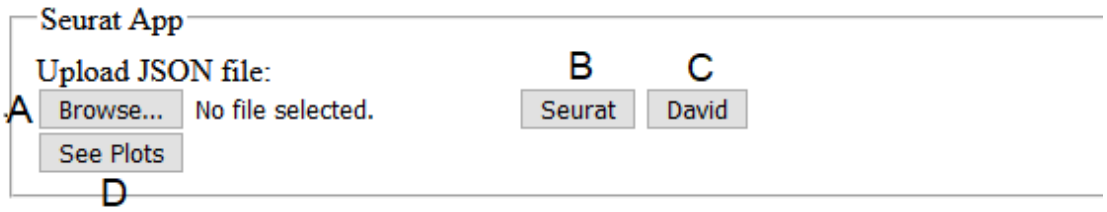


Figure 5.2: Overview of the web application interface. **(A)** Button to select and upload a JSON configuration file. **(B)** Button to run the Seurat analysis pipeline. **(C)** Button to run the DAVID analysis. **(D)** Button to see the images produced by the Seurat analysis.

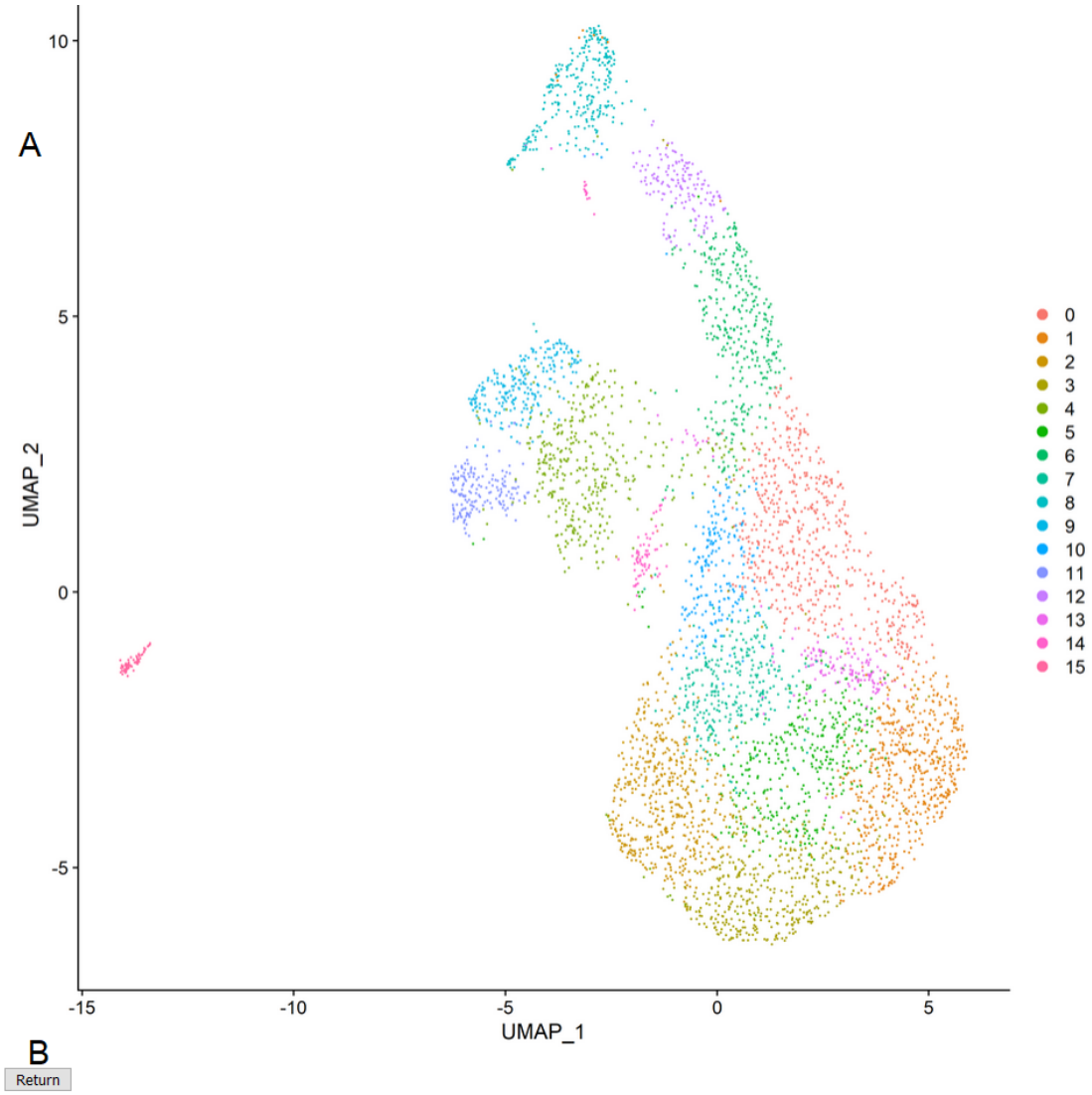


Figure 5.3: Overview of the plot display screen. **(A)** Example of a plot produced by Seurat. **(B)** Return button that takes the user back to the main interface page.

Chapter 6

Conclusions

6.1 Introduction

This chapter ends the study by listing the conclusions drawn from the obtained results and leaves suggestions for future enhancements.

6.1.1 Conclusions

We have provided an overview of the current state of the art regarding the area of bioinformatic by exploring some of the more relevant articles and tools in recent years. Due to its previous usage in groundbreaking research, one tool in particular stood out as being relevant for this topic: Seurat.

Using this tool, the results from a specific paper on the topic were attempted to be replicated. Although this goal hasn't been reached with complete accuracy, Seurat presented itself as a potent, flexible tool, and we believe that with a deeper understanding of the dataset, more theoretical knowledge and further tuning this goal could be properly achieved.

From the results obtained with Seurat, GO was conducted using DAVID, a set of functional annotation tools for gene enrichment, useful for extracting detailed information from lists of genes. More specifically, information regarding the cytoskeleton of OLs was collected. Further studies need to be conducted on this information in order to detect potential patterns that could lead to useful knowledge on the importance of this structure in the neurological cell behavior.

Additionally, a simple web application was built. This application aimed at providing an accessible, user friendly interface for the use of Seurat. It also serves as a bridge between this tool, the Python language and the web. This application demonstrates how a powerful GO tool, DAVID, can be integrated in a web resource along scRNA-seq data analysis tools, providing substantial value for future research.

Although the objectives that were initially defined weren't completely achieved, this paper serves nonetheless as a summary of the current state of the art and an exercise with one of the more promising tools.

6.2 Future Work

Due to the available amount of time, the web application could not be fully developed, and is left in a rudimentary state.

There is, therefore, plenty of room for improvement left. Numerous aesthetic improvements could be made in the interface and a form could be created to facilitate the input of the user's preferences.

Regarding the DAVID features, they aren't ready to take a specific groups of genes, other report analysis types could be made implemented (only a chart report analysis is available in the current version) and an option to save the generated data needs to be created.

Regarding the Seurat script, it needs further improvements on it's flexibility, stability and overall the web application.

References

[Aitamurto, 2015] Aitamurto, T. (2015). Myelination assay. https://www.electrospinning.co.uk/case-studies/myelination-assay/.

[Andy Coenen,] Andy Coenen, A. P. Understanding umap. https://pair-code.github.io/understanding-umap/.

[Azevedo et al., 2018] Azevedo, M. M., Domingues, H. S., Cordelières, F. P., Sampaio, P., Seixas, A. I., and Relvas, J. B. (2018). Jmy regulates oligodendrocyte differentiation via modulation of actin cytoskeleton dynamics. *GLIA*.

[Buettner et al., 2015] Buettner, F., Natarajan, K. N., Casale, F. P., Proserpio, V., Scialdone, A., Theis, F. J., Teichmann, S. A., Marioni, J. C., and Stegle, O. (2015). Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nature Biotechnology*, 33(2):155–160.

[Butler et al., 2018] Butler, A., Hoffman, P., Smibert, P., Papalexi, E., and Satija, R. (2018). Integrating Single-Cell Transcriptomic Data Across Different Conditions, Technologies, and Species.

[Clancy and Brown, 2008] Clancy, S. and Brown, W. (2008). Translation: DNA to mRNA to Protein | Learn Science at Scitable. *Nature Education*, 1(1):1–7.

[Domingues et al., 2018] Domingues, H. S., Cruz, A., Chan, J. R., Relvas, J. B., Rubinstein, B., and Pinto, I. M. (2018). Mechanical plasticity during oligodendrocyte differentiation and myelination.

[Falcão et al., 2018] Falcão, A. M., van Bruggen, D., Marques, S., Meijer, M., Jäkel, S., Agirre, E., Samudyata, Floriddia, E. M., Vanichkina, D. P., Ffrench-Constant, C., Williams, A., Guerreiro-Cacais, A. O., and Castelo-Branco, G. (2018). Disease-specific oligodendrocyte lineage cells arise in multiple sclerosis. *Nature Medicine*, 24(12):1837–1844.

[Galatro et al., 2017] Galatro, T. F., Holtman, I. R., Lerario, A. M., Vainchtein, I. D., Brouwer, N., Sola, P. R., Veras, M. M., Pereira, T. F., Leite, R. E., Möller, T., Wes, P. D., Sogayar, M. C., Laman, J. D., Den Dunnen, W., Pasqualucci, C. A., Oba-Shinjo, S. M., Boddeke, E. W., Marie, S. K., and Eggen, B. J. (2017). Transcriptomic analysis of purified human cortical microglia reveals age-associated changes. *Nature Neuroscience*, 20(8):1162–1171.

[Harris et al., 2008] Harris, M. A., Deegan, J. I., Ireland, A., Lomax, J., Ashburner, M., Tweedie, S., Carbon, S., Lewis, S., Mungall, C., Day-Richter, J., Eilbeck, K., Blake, J. A., Bult, C., Diehl, A. D., Dolan, M., Drabkin, H., Eppig, J. T., Hill, D. P., Ni, L., Ringwald, M., Balakrishnan, R., Binkley, G., Cherry, J. M., Christie, K. R., Costanzo, M. C., Dong, Q., Engel, S. R., Fisk, D. G., Hirschman, J. E., Hitz, B. C., Hong, E. L., Krieger, C. J., Miyasato, S. R., Nash, R. S., Park,

J., Skrzypek, M. S., Weng, S., Wong, E. D., Zhu, K. K., Botstein, D., Dolinski, K., Livstone, M. S., Oughtred, R., Berardini, T., Li, D., Rhee, S. Y., Apweiler, R., Barrell, D., Camon, E., Dimmer, E., Huntley, R., Mulder, N., Khodiyar, V. K., Lovering, R. C., Povey, S., Chisholm, R., Fey, P., Gaudet, P., Kibbe, W., Kishore, R., Schwarz, E. M., Sternberg, P., Van Auken, K., Giglio, M. G., Hannick, L., Wortman, J., Aslett, M., Berriman, M., Wood, V., Jacob, H., Laulederkind, S., Petri, V., Shimoyama, M., Smith, J., Twigger, S., Jaiswal, P., Seigfried, T., Howe, D., Westerfield, M., Collmer, C., Torto-Alalibo, T., Feltrin, E., Valle, G., Bromberg, S., Burgess, S., and McCarthy, F. (2008). The Gene Ontology project in 2008. *Nucleic Acids Research*, 36(SUPPL. 1):440–444.

[Huang et al., 2007] Huang, D. W., Sherman, B. T., Tan, Q., Collins, J. R., Alvord, W. G., Roayaei, J., Stephens, R., Baseler, M. W., Lane, H. C., and Lempicki, R. A. (2007). The DAVID Gene Functional Classification Tool: A novel biological module-centric algorithm to functionally analyze large gene lists. *Genome Biology*, 8(9).

[Jaadi, 2020] Jaadi, Z. (2020). A step by step explanation of principal component analysis. https://builtin.com/data-science/step-step-explanation-principal-component-analysis.

[Jäkel et al., 2019] Jäkel, S., Agirre, E., Mendanha Falcão, A., van Bruggen, D., Lee, K. W., Knuesel, I., Malhotra, D., Ffrench-Constant, C., Williams, A., and Castelo-Branco, G. (2019). Altered human oligodendrocyte heterogeneity in multiple sclerosis. *Nature*, 566(7745):543–547.

[Jović et al., 2014] Jović, A., Brkić, K., and Bogunović, N. (2014). An overview of free software tools for general data mining. *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2014 - Proceedings*, pages 1112–1117.

[Khatri et al., 2004] Khatri, P., Bhavsar, P., Bawa, G., and Draghici, S. (2004). Onto-Tools: An ensemble of web-accessible ontology-based tools for the functional design and interpretation of high-throughput gene expression experiments. *Nucleic Acids Research*, 32(WEB SERVER ISS.):449–456.

[Kurita, 2018] Kurita, K. (2018). Paper dissected: "visualizing data using t-sne" explained. https://mlexplained.com/2018/09/14/paper-dissected-visualizing-data-using-t-sne-explained/.

[Lab, 2020] Lab, S. (2020). Seurat: R toolkit for cell genomics. https://satijalab.org/seurat/.

[Marques et al., 2018] Marques, S., van Bruggen, D., Vanichkina, D. P., Floriddia, E. M., Munguba, H., Väremo, L., Giacomello, S., Falcão, A. M., Meijer, M., Björklund, Å. K., Hjerling-Leffler, J., Taft, R. J., and Castelo-Branco, G. (2018). Transcriptional Convergence of Oligodendrocyte Lineage Progenitors during Development. *Developmental Cell*, 46(4):504–517.e7.

[Marques et al., 2016] Marques, S., Zeisel, A., Codeluppi, S., van Bruggen, D., Mendanha Falcão, A., Xiao, L., Li, H., Häring, M., Hochgerner, H., Romanov, R. A., Gyllborg, D., Muñoz-Manchado, A. B., La Manno, G., Lönnerberg, P., Floriddia, E. M., Rezayee, F., Ernfors, P., Arenas, E., Hjerling-Leffler, J., Harkany, T., Richardson, W. D., Linnarsson, S., and Castelo-Branco, G. (2016). Oligodendrocyte heterogeneity in the mouse juvenile and adult central nervous system. *Science*, 352(6291):1326–1329.

[Miguel and Natividade, 2017] Miguel, L. and Natividade, B. (2017). Data Mining para análise dos resultados de Gene Expression.

[Mokobi, 2020] Mokobi, F. (2020). Myelination assay. https://microbenotes.com/plant-cell/.

[Pfeiffer et al., 1993] Pfeiffer, S. E., Warrington, A. E., and Bansal, R. (1993). The oligodendrocyte and its many cellular processes. *Trends in Cell Biology*, 3(6):191–197.

[Rasband,] Rasband, W. Imagej homepage. https://imagej.nih.gov/ij/features.html.

[Stegle et al., 2015] Stegle, O., Teichmann, S. A., and Marioni, J. C. (2015). Computational and analytical challenges in single-cell transcriptomics. *Nature Reviews Genetics*, 16(3):133–145.

[Stuart et al., 2019] Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., Mauck, W. M., Hao, Y., Stoeckius, M., Smibert, P., and Satija, R. (2019). Comprehensive Integration of Single-Cell Data. *Cell*, 177(7):1888–1902.e21.

[van Bruggen et al., 2017] van Bruggen, D., Agirre, E., and Castelo-Branco, G. (2017). Single-cell transcriptomic analysis of oligodendrocyte lineage cells. *Current Opinion in Neurobiology*, 47:168–175.

[Wang et al., 2020] Wang, F., Ren, S. Y., Chen, J. F., Liu, K., Li, R. X., Li, Z. F., Hu, B., Niu, J. Q., Xiao, L., Chan, J. R., and Mei, F. (2020). Myelin degeneration and diminished myelin renewal contribute to age-related deficits in memory. *Nature Neuroscience*.

[Wang et al., 2010] Wang, Z., Gerstein, M., and Snyder, M. (2010). Nihms229948. 10(1):57–63.

[Xu and Su, 2015] Xu, C. and Su, Z. (2015). Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics*, 31(12):1974–1980.

[Zeisel et al., 2015] Zeisel, A., Muñoz-Manchado, A. B., Codeluppi, S., Lönnerberg, P., Manno, G. L., Juréus, A., Marques, S., Munguba, H., He, L., Betsholtz, C., Rolny, C., Castelo-Branco, G., Hjerling-Leffler, J., and Linnarsson, S. (2015). Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science*, 347(6226):1138–1142.

[Zhang et al., 2014] Zhang, Y., Chen, K., Sloan, S. A., Bennett, M. L., Scholze, A. R., O'Keeffe, S., Phatnani, H. P., Guarnieri, P., Caneda, C., Ruderisch, N., Deng, S., Liddelow, S. A., Zhang, C., Daneman, R., Maniatis, T., Barres, B. A., and Wu, J. Q. (2014). An RNA-sequencing transcriptome and splicing database of glia, neurons, and vascular cells of the cerebral cortex. *Journal of Neuroscience*, 34(36):11929–11947.

Appendix A

Appendix

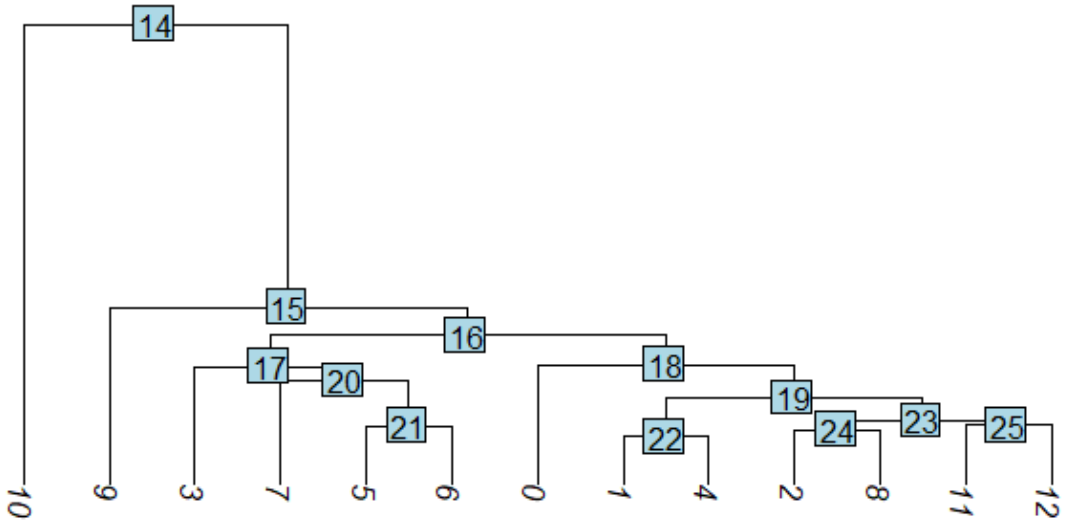


Figure A.1: Cluster tree obtained using Seurat's native methods.

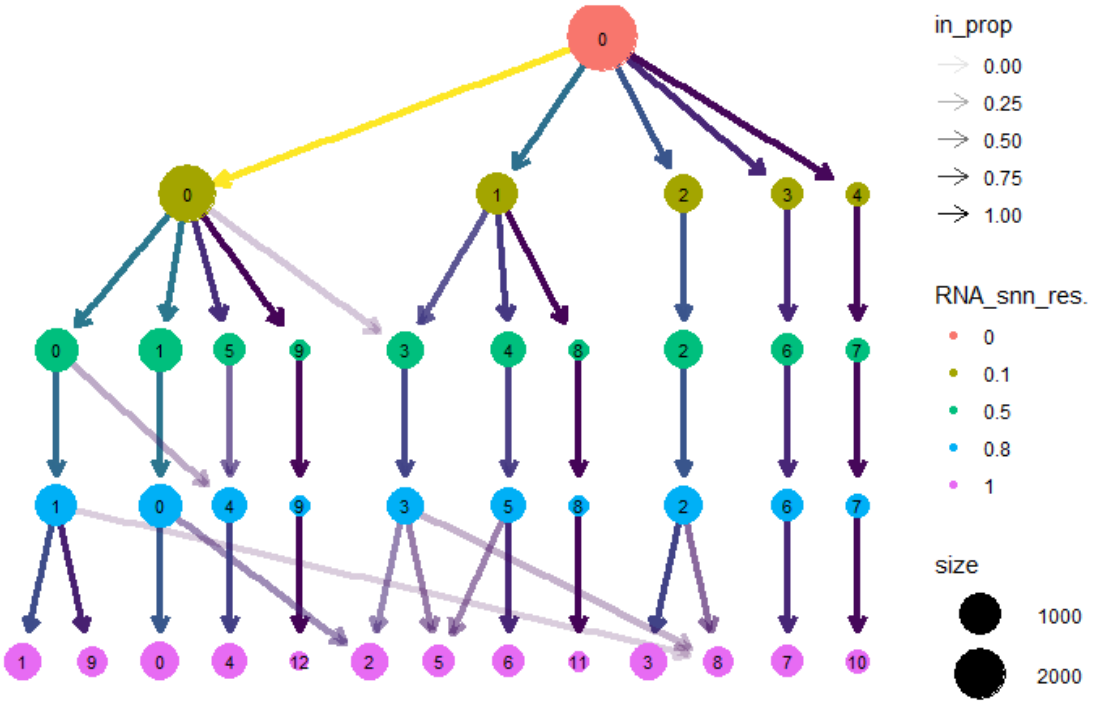


Figure A.2: Cluster tree obtained using the exterior package clustree.

```
{
    "setup": {
        "datapath": "D:/Documents/gene_expression_matrix.tab",
        "savepath": "D:/Documents/results",
        "saverob": "yes"
    },
    "options": {
        "normalization": "yes",
        "featselect": "yes",
        "scaling": "yes",
        "lindimred": "yes",
        "datadim": "yes",
        "clustering": {
            "clustering": "yes",
            "kparam": "25"
        },
        "nonlindimred": {
            "nonlindimred": "yes",
            "type": "umap"
        },
        "clustree": "yes",
        "commongenes": "yes",
        "markers": "yes",
        "averageexp": "yes"
    },
    "david": {
        "chart": "yes",
        "table": "no",
        "termclust": "no",
        "geneclust": "no"
    }
}
```

Figure A.3: Structure of the input configuration JSON file. In **"setup"**, the "datapath" attribute defines the localization of the dataset, the "savepath" attribute defines the folder where any images produced by the analysis will be locally stored and the "saverob" attribute defines if the R object with the session data is to be saved. In **"options"**, each attribute represents a step of the Seurat pipeline, along with the definition of some variable values. In **"david"**, each attribute represents a type of analysis.