Universidad de Concepción
Dirección de Postgrado
Facultad de Ingeniería - Programa de Doctorado en Ciencias de la Computación

MACHINE LEARNING CLASSIFICATION OF SINGLE CELL RNA-SEQ ACROSS DIFFERENT TYPES OF CANCER

Tesis para optar al grado de
DOCTOR EN CIENCIAS DE LA COMPUTACIÓN

POR
MABEL ANGÉLICA VIDAL MIRANDA
CONCEPCIÓN, CHILE
Abril, 2022

Profesor guía: Guillermo Cabrera Vives
Departamento de Ingeniería Informática y Ciencias de la Computación
Facultad de Ingeniería, Universidad de Concepción

Profesora co-guía: Estefanía Nova Lamperti
Departamento de Bioquímica Clínica e Inmunología
Facultad de Farmacia, Universidad de Concepción

ACKNOWLEDGMENTS

I want to thank my advisors Guillermo Cabrera-Vives and Estefanía Nova-Lamperti for the support, motivation and freedom to allow me do research in areas of my interest. Thank to professors in the Computer Science department. I especially want to thank professors Diego Seco and Roberto Asín, for their countless advice and always attending to my queries in a prompt manner regardless of the time. Thank to Johanna, Carolina and Hugo for the support and kindness provided throughout this time.

Thank to WIE UdeC and Niñas Pro members for celebrating each of my achievements and working hard during this period to reduce the gender gap.

This work was funded by the National Agency for Research and Development (ANID) / Scholarship Program / DOCTORADO NACIONAL / 2020 - 21201560, Graduate Office and Faculty at Engineering of the University of Concepción.

Abstract

Human cancers are complex ecosystems composed of different types of cells. The diverse populations of co-existing cells within the same tumor that have genetic, functional, and environmental differences determine the tumor heterogeneity, which is one of the major challenges facing cancer diagnosis and treatment. The aim of this thesis was to apply different machine learning methods to classify single cell RNA-seq (scRNA-seq) samples across nine different types of cancer. We observed that T cells are the most abundant datasets in public repositories due to their important role in immunotherapies. For this reason, we performed an in-silico analysis from scRNA-seq data available in the Gene Expression Omnibus. A first approach was to analyze and characterize genetic T cell signatures from five different types of cancer and apply dimensionality reduction and clustering methods to identify subpopulations from malignant and non-malignant datasets. This analysis revealed that pathways related to immune response, metabolism and viral immunoregulation were observed exclusively in samples of malignant origin. A second approach was to perform two deep learning models to classify cells from nine different types of cancer, where the cells were grouped in the diversity of the cell state, giving us a new perspective in the different classes of tumors present in our dataset. Finally, we observed that working with unsupervised methods, our data help us understand the heterogeneity between tumors. Characterization of cellular diversity was associated with pathways that play a key role in tumor proliferation, progression, and regulation of the microenvironmental immune response.

Tabla de Contenido

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Cancer is one of the leading causes of death world-wide and one of the most complex diseases to treat [173]. The World Health Organization (WHO), indicates that late detection and diagnosis are common in most reported cases, lung cancer being the most common cause of death (1.76 million deaths), followed by colorectal (862 000 deaths), stomach (783 000 deaths), liver (782 000 deaths) and breast cancer (627 000 deaths). Diagnosis and detection of the disease at an early stage is essential to increase the chances of survival [155].

As a way of processing complex datasets, machine learning has been used for biological research [25], to uncover underlying patterns, build models and make predictions with general purpose approaches to learn functional relationships from the data [143, 107]. Applications include data analysis from genomics, proteomics, transcriptomics, medical imaging, among others [7, 121, 215, 92, 195].

Deep learning (DL) methods have proven to significantly outperform classical machine learning methods in different tasks, making it the state-of-the-art solution in many fields [109, 77, 222, 76, 139]. DL models are based on neural networks that learn from the data and are usually composed of numerous parameters subject to modification. In order to properly set these parameters, a large amount of data is required. In the case of cancer analysis, the data may correspond to images or gene expression values. Over the last couple of years, these types of data have shown a remarkable volume growth [63, 133]. In terms of images, there are different technologies and strategies [53]. For example, computer-aided detection (CAD) systems help to recognize patterns in images that might be associated with abnormalities [27, 111]. Likewise, ultrasound imaging is useful in cancer detection, characterizing lesions in different organs and tissues by using

high-frequency sound waves that penetrate tissues in the body [95]. On the other hand, magnetic resonance imaging (MRI) is highly sensitive at detecting tissue abnormalities. Some of its applications include diagnosis, staging, personalized treatment, and treatment monitoring. Thanks to all these advantages, images are the most common type of data analyzed in cancer research [73, 191, 61]. When studying gene expressions, the data correspond to hundreds of gigabytes or terabytes of DNA base pairs depending on the sequencing technology employed [209, 183]. Genomic data commonly comes from next generation sequencing (NGS) and DNA microarrays. One of the advantages of NGS technologies is obtaining the gene expression level of different types of cancer. Thus, contributing to the understanding of genetic differences and heterogeneity in cancer cells and tissues [123, 71]. Single-cell RNA-seq (scRNA-seq) allows to obtain a full genetic description of single cells in comparison with massive sequencing. However, scRNA-seq contains more noise than the analysis of massive sequencing [198], due to a greater amplification of the genetic material and a smaller number of samples. Despite that, methods aimed at reducing dimensionality and identifying subpopulations, as well as clustering methods from machine learning, have improved the analysis to get reliable single cell data [100].

However, using DL for cancer research is a big challenge because the input data in a neural network can vary. Neural network architectures depend on the data available and the question in need of an answer. The state-of-the-art solutions normally only focus on one type of data. As for the learning process, it relies on the data and architecture used. This thesis aims to use machine learning and deep learning to develop classification models to apply in single cell RNA-seq of different types of cancer that contains different labeled and unlabeled cells important in the immune responses. Thus, the right classification and differentiation between the samples, can determine the important genes for cancer prediction and subsequently a treatment.

1.2 Contributions

This section presents the main contributions of the thesis.

- We integrate a large number of cells that come from different types of cancer and with different interests of research, using the next-generation sequencing technology scRNA-seq that helps us to understand the different cell populations and the relationship between genes from nine different types of cancer. No previous work has integrated the amount of data that we used.

- We study some types of cancer in a unique way, such as glioblastoma. Where we applied our unsupervised deep learning frameworks (autoencoder, variational autoencoder and variational deep embedding) for scRNA-seq to infer the cell-type in glioblastoma, but using different datasets. However we observed that even for only one type of cancer it was complex to identify the cell types in an unsupervised way, because they came from different experiments. This work was presented at the EACR Bioinformatics in Cancer 2021 Conference (Virtual).

- We discovered a possible atlas for cell classification by collecting information from the state of the art and performing a semi-supervised machine learning model, it was possible to identify the different cell types. This work was presented at the Single-Cell RNA-seq 2020 Workshop, Earlham Institute, UK.

- We identified common genes across five different types of cancer and compared to non-malignant genes for each T cell subset to identify specific pathways. Exclusive pathways in CD4+ cells, CD8+ cells and Tregs, and common pathways for the tumor-infiltrating T cell subsets were identified. This research was presented at the 2018 Grace Hopper Celebration. Houston, Texas, USA.

  Then, the identified pathways were compared with RNAseq and proteomic data obtained from T cell subsets cultured under malignant environments and we observed that cytokine signaling, especially Th2-type cytokine was the top overrepresented pathway in Tregs from malignant samples.

  We observed that previous data from the Molecular and Translational Immunology Laboratory, have demonstrated an increment in Th2-like Tregs and Teff infiltrated

subsets in melanoma [69], colorectal cancer [69] and oral cancer [52] in comparison with infiltrated subsets from non-malignant tissues (Figure 1.1A). In addition, we identified that Vitamin D signaling promotes these Th subset disbalance in oral cancer. This work was published in [52]. We then compared our RNAseq data obtained from T-cells subsets cultured with malignant and non-malignant environments from oral cancer and proteomic data from CD4-T cells cultured with Vitamin D with the data obtained from the in-silico analysis of this thesis.

In total, 481 genes were obtained from the RNA-seq experiments. For CD4-T cells, 218 common pathways were identified between scRNA-seq and RNA-seq data, whereas for Tregs, 194 common pathways were identified. No CD8 were analyzed for RNA-seq data. For proteomics, only CD4 T cells were analyzed. After analysis, 1,692 proteins were found in the gene list obtained from the sc-RNAseq data, resulting in 561 common pathways.

Signaling by interleukins, nucleotide-binding domain, leucine rich repeat containing receptor (NLR), TRAF6 mediated NF-kappaB activation and Toll Like receptor cascades were the top overrepresented common pathways in CD4-T cells (Figure 1.1B). For Tregs, the data revealed that signaling by interleukins (IL-4, IL-13 and IL-10), transcriptional regulation by TP53, interferon alpha/beta signaling, and Nerve Growth Factor (NGF) stimulated transcription and NTRK were the top overrepresented common pathways (Figure 1.1C). Finally, proteomic data revealed common pathways such as cytokine signaling in immune system, interferon gamma signaling, downstream TCR signaling, MHC class II antigen presentation, and PD-1 signaling (Figure 1.1D). Overall these data support the observation of a preference of regulatory Th2-like cells in cancer, as previously described by us and others, and other interesting pathways such as the recognition of bacteria and virus by NLR and TLR signaling in CD4 and the NGF receptor tyrosine-kinase TrkA signaling in Tregs.

Therefore, we validated and observed common pathways for CD4-T cells and Treg with our experiments, maintaining all the discoveries from the in-silico analysis across different types of cancers (article submitted to Cancers journal).

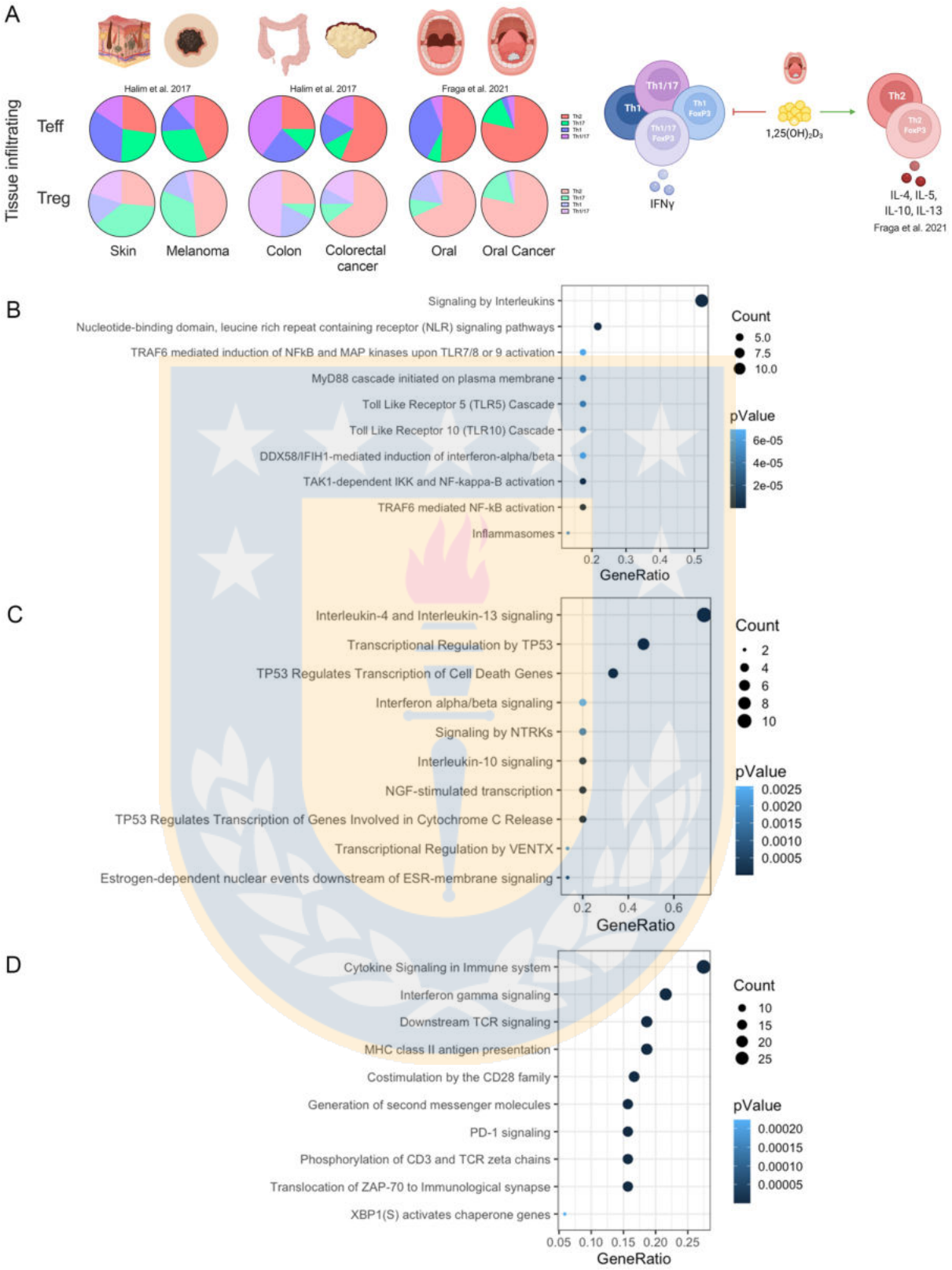


Figure 1.1: Pathways validation.
(**A**) Summary of T-cells in melanoma, colorectal cancer and oral cancer from our lab. Scatter plot of (**B**) Common pathways between CD4-T cells and RNA-seq experiments. (**C**) Common pathways between Treg and RNA-seq experimients. (**D**) Common pathways between CD4-T cells and proteomic data.

- By applying an unsupervised model to different scRNA-seq datasets, it is not possible to integrate all data if sequencing methodologies are not standardized. Therefore, it must perform a separate analysis and then join the data.

- The biological results and methodology implemented were also useful in another research. Such as to study mesenchymal stem cells in a type-2 Diabetes mouse model. This work was published in [156]. Finally, other contribution was in the identification of SARS-CoV-2 infections and reinfection, these studies were published in [2, 11].

1.3 Structure of the thesis

The thesis is organized as follows:

In Chapter 2: We expose the research carried out in the thesis, that is, the hypothesis, general and specific goals, as well as the methodology, available resources, and databases used.

In Chapter 3: We talk about the theoretical framework for machine learning and deep learning needed to understand the remaining chapters.

In Chapter 4: We introduce analysis of tumor-infiltrating T cell across different types of cancer supported by machine learning tools. Moreover, we show the results of how we identified only T cells between others type of cells using machine learning models for clustering.

In Chapter 5: We present machine learning and deep learning models to classify different types of cells across nine type of cancer and how the cells are associated to the tumor development.

In Chapter 6: We discuss the final conclusions and future work.

Chapter 2

Research conducted

2.1 Hypothesis

Machine learning and deep learning models based on gene expression profiles from different types of cancer allow the characterization of the tumor heterogeneity by classifying the different types of cells within the tumor and by identifying key genes in the malignant environment.

2.2 Goals

2.2.1 General Goal

To develop machine learning and deep learning models based on gene expression from single cell RNA-sequencing repositories from nine different types of cancer to classify the different types of cells within the tumor and key genes supporting the malignant environment.

2.2.2 Specific Goals

SG1 Develop a computational framework for a high dimensional gene expression data integration from single cell RNA-sequencing repositories obtained from nine different types of cancer.

SG2 Develop a classification model for cell types (T, B, natural killers, macrophages, mast, monocytes, dendritic, neutrophils, endothelial, fibroblast, myocytes, CAFs, myeloid, myofibroblast and epithelial) based on machine learning methods.

SG3 Identify relevant genes in malignant samples and evaluate them with the gene ontology database[1] to study the pathways associated to the development of cancer.

2.3 Methodology

To achieve the specific goals of this thesis, the following tasks were completed:

T1: Review and study the state-of-the-art of machine learning and deep learning applications in genomic data.

T2: Search datasets of scRNA-seq available in public repositories, that were performed with similar experimental design from different types of cancer.

T3: Verify the quality of each sequencing library and perform a digital expression matrix using transcripts per million (TPM) values as gene expression levels for all datasets selected.

T4: Search common genes across all datasets and create a new matrix.

T5: Data exploration of relevant gene markers.

T6: Implement a model for dimensionality reduction and cell type clustering.

T7: Perform a pathway enrichment analysis of relevant genes and visualize how genes are functionally grouped.

2.4 Available resources and databases

The Department of Computer Science of the Universidad de Concepción has the adequate computing resources for the development of this thesis. It was performed in a Linux platform, specifically Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-101-generic x86_-64), mainly using the following programming languages: Python, C and R. The hardware characteristics are:

- CPU cores: 16

[1] http://www.geneontology.org/

- Architecture: x86_64 (64-bit)
- CPU model: Intel(R) Xeon(R) CPU E5620 @ 2.40GHz
- RAM: 64 GB
- Total space in disk used: 1,8 TB.

The main database used to obtain the datasets was the Gene Expression Omnibus (GEO). It is a database for gene expression profiling and RNA methylation profiling managed by the National Center for Biotechnology Information. These high-throughput screening genomics data are derived from microarray or RNA-Seq experimental data (https://www.ncbi.nlm.nih.gov/geo/).

To perform the pathway enrichment analysis we used the Gene Ontology Consortium database (data-version from 2020-05-02) that provides structured, controlled vocabularies and classifications that cover several domains of molecular and cellular biology and are freely available for community use in the annotation of genes, gene products and sequences [36].

Chapter 3

Theoretical framework

3.1 Machine Learning

Machine learning is a subset of artificial intelligence. It is defined as a set of methods that can detect patterns in data and use the uncovered patterns to predict future data, or to perform other kinds of decisions [146]. Usually, machine learning is divided in supervised, unsupervised and reinforcement learning.

Supervised learning: the goal is to learn a mapping from inputs x to outputs y, given a labeled set of input-output pairs $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$, where D is the training set and N is the number of training set. The learner receives a set of labeled examples as training data and makes predictions for all unseen points. Commonly, this scenario is associated with classification, regression, and ranking problems.

Unsupervised learning: the goal is to find interesting patterns in the data $D = \{\boldsymbol{x}_i\}_{i=1}^{N}$. The learner exclusively receives unlabeled training data, and makes predictions for all unseen points. This scenario is associated with learning task such as clustering and dimensionality reduction.

Reinforcement learning: the goal is to learn the optimal behavior in an environment to obtain maximum reward. This optimal behavior is learned through interactions with the environment and observations of how it responds. It differs from supervised learning due to in reinforcement learning the agent decides what to do to perform the given task, in the absence of a training dataset, learning from its experience.

3.1.1 Supervised learning

Classification

In this process the goal is to learn a mapping from inputs x to outputs y, where $y \in \{1, ..., C\}$ with C being the number of classes. When $C = 2$, it is a binary classification, and when $C > 2$, it is a multiclass classification. In the classification setting there are a set of training observations $(x_1, y_1), ..., (x_n, y_n)$ that can be used to build a classifier and this classifier should perform well not only on the training data, but also on the test observations [87].

Some common algorithms used for classification are:

Decision Tree: the simplest description is a divide-and-conquer approach. This algorithm is used to discover features and extract patterns in large datasets for discrimination and predictive modeling. A decision tree is constructed by recursively partitioning the feature space of the training set, finding a set of decision rules that naturally partition the feature space to provide an informative and robust hierarchical classification model [147].

Random Forest: it is a ensemble[1] classifier that produces multiple decision trees, using a randomly selected subset of training samples and variables. The trees are created by drawing a subset of training samples with replacement, i.e., the same objects can be selected several times, while others may not be selected at all. The decision forest chooses the classification, which has the most votes over all the trees in the forest. If the number of instances in a dataset is N, almost $2/3$ of the original size is randomly selected through bootstrapping manner N times. The remaining instances have been used as an out-of-bag set to be evaluated. The set of out-of-bag are those observations that are not used to build the sub-trees, those have been used for evaluating the error prediction. In the last two decades this classifier has performed excellent classification results and speed of processing [13].

Gradient Boosting: it is an ensemble learning method, where each predictor tries to improve on its predecessor by reducing the errors. Gradient boosting has three main

[1] An ensemble method is an approach that combines many simple building block models in order to obtain a single and potentially powerful model.

components: a loss function, weak learners and an additive model. The role of the loss function is to estimate how good the model is at making predictions with the given data. A weak learner is one that classifies the data but does not perform well on its own, perhaps no better than random guess, therefore, it presents a high error rate. The additive model is an iterative and sequential approach of adding the trees (weak learners) one step at a time. Thus, each iteration should reduce the value of the loss function [146].

Naive Bayes: it is a probabilistic machine learning model used for classification tasks based on the Bayes Theorem. It assumes that the presence of one feature in a class is independent of the other feature present in the same class. The Bayes theorem is defined as:

$$p(C_j|x) = \frac{p(C_j)p(x|C_j)}{p(x)}, \tag{3.1}$$

where, C_j represent the possible outcomes or classes and x is the feature vector; $p(C_j|x)$ is the posterior probability of class C_j given the predictor x; $p(C_j)$ is the prior probability of class C_j; $p(x|C_j)$ is the likelihood which is the probability of x given we know it is from class C_j, and $p(x)$ is prior probability of x predictor. Then, a Bayesian classifier generates a label L such as:

$$L = arg\, max_{j\in\{1,2,\ldots,j\}} p(C_j) \prod_{i=1}^{n} p(x_i|C_j). \tag{3.2}$$

In the learning process the known structure, class and conditional probabilities are calculated using the training data, and then the values of these probabilities are used to classify new observations [200].

k-Nearest Neighbor (KNN): it is a non-parametric[2] classifier that looks at a positive integer of K points in the training set that are nearest to the test input x, counts how many members of each class are in this set, and returns that empirical fraction

[2]Algorithms that do not make particular assumptions about the kind of mapping function are known as non-parametric algorithms.

as the estimate [146]. KNN is defined as:

$$p(y = c|x, \mathcal{D}, K) = \frac{1}{K} \sum_{i \in N_k(x,\mathcal{D})} \mathbb{I}(y_i = c), \tag{3.3}$$

where $N_k(x, \mathcal{D})$ are the indices of the K nearest points to x in $\mathcal{D}$ (training data) and $\mathbb{I}(e)$ is the indicator function defined as follows:

$$\mathbb{I} = \begin{cases} 1 \text{ if } e \text{ is true} \\ 0 \text{ if } e \text{ is false.} \end{cases} \tag{3.4}$$

Support Vector Machine (SVM): it is a discriminative (conditional) classification model that learns linear or nonlinear decision boundaries in the attribute space to separate the classes. The objective of the algorithm is to find a hyperplane in an N-dimensional space, where N is the number of features, that distinctly classifies the data points. When separating the two classes of data points, there are many possible hyperplanes that could be chosen, therefore, it is key to find a plane that has the maximum margin, i.e. the maximum distance between data points of both classes [202].

Logistic Regression: it is a probabilistic discriminative model for classification, which directly estimates the odds of a data instance $\boldsymbol{x}$ using its attribute values. The idea is to use a linear predictor, $z = \boldsymbol{w}^T\boldsymbol{x} + b$, to represent the odds of $\boldsymbol{x}$ as follows:

$$\frac{P(y = 1|\boldsymbol{x})}{P(y = 0|\boldsymbol{x})} = e^z = e^{\boldsymbol{w}^T\boldsymbol{x}+b}, \tag{3.5}$$

where, $\boldsymbol{w}$ and b are the parameters of the model and $\boldsymbol{a}^T$ denotes the transpose of a vector $\boldsymbol{a}$. If $\boldsymbol{w}^T\boldsymbol{x} + b > 0$, then $\boldsymbol{x}$ belongs to class 1 since this probability is greater than the probability of belonging to class 0; otherwise, $\boldsymbol{x}$ belongs to class 0. Logistic regression uses a sigmoid function and works best on binary classification problems, although it can be used on multiclass classification problems through the one versus all method [202].

Most common metrics that have been used widely while evaluating a classification

model are:

Accuracy: it measures how often the classifier correctly predicts. It is defined as the ratio of the number of correct predictions and the total number of predictions.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.6}$$

True positive rate (TPR) or recall or sensitivity: it is the probability that an actual positive will test positive.

$$TPR = \frac{TP}{TP + FN} \tag{3.7}$$

True negative rate (TNR) or specificity: it is the probability that an actual negative will test negative.

$$TNR = \frac{TN}{TN + FP} \tag{3.8}$$

False positive rate (FPR): it is the proportion of all negatives that still yield positive test outcomes.

$$FPR = \frac{FP}{FP + TN} \tag{3.9}$$

False negative rate (FNR:) it is the proportion of positives which yield negative test outcomes with the test.

$$FNR = \frac{FP}{FP + TN} \tag{3.10}$$

Precision: it is the ratio of True Positives to all the positives predicted by the model.

$$Precision = \frac{TP}{TP + FP} \tag{3.11}$$

F-measure or F1-score: it is a single metric that combines both Precision and Recall. The higher the F1 score, the better is the performance of the model. The range for F1-score is [0,1].

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3.12}$$

Area under the curve (AUC): is the measure of the ability of a classifier to distinguish between classes. And a ROC curve (Receiver Operating Characteristic curve) is a graph showing the performance of a classification model. It is a way to visualize the tradeoff between the True Positive Rate (TPR) and False Positive Rate(FPR) using different decision thresholds (the threshold for deciding whether a prediction is labeled "true" or "false") for the predictive model.

$$AUC = \frac{1 + TPR - FPR}{2}, \tag{3.13}$$

where TP is true positive (correctly classified), FN is false negative, FP is false positive and TN is true negative.

Regression

Regression is the process of finding the correlations between dependent and independent variables. In this scenario, the response variable is continuous. The task of the regression algorithms is to find the mapping function to map the input variable X to the continuous output variable y.

Some types of regression algorithms are:

Simple linear regression: it is a very straightforward approach for predicting a quantitative response y on the basis of a single predictor variable X. It assumes that there is approximately a linear relationship between X and y. It is defined as $y \approx \beta_0 + \beta_1 X$, where β_0 and β_1 are two unknown constants that represent the intercept and slope terms in the linear model.

Multiple linear regression: this approach depends on more than one variable. Using more independent variables can improve the accuracy of the model, as long as the variables are relevant to the problem. The multiple linear regression model takes the form: $y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p + \epsilon$, where X_j represents the jth predictor and β_j quantifies the association between that variable and the response.

Polynomial regression: can be considered a special case of multilinear regression, in which the data distribution is more complex than a linear one, i.e., the dependent

variable X and the independent variable y are modelled as the nth degree polynomial in X. It is defined as: $y = \beta_0 + \beta_1 X_1 + \beta_2 X_2^2 + \beta_3 X_3^3 + ... + \beta_d X_i^d + \epsilon_i$, where ϵ_i is the error term and d is the degree of the polynomial function.

Support vector regression (SVR): similar to the support vector machine (SVM) classification algorithm, it is an extension of the margin used to the regression setting, where a hyperplane with maximum margin such that the maximum number of data points are within the margin. SVR instead seeks for coefficients that minimize a different type of loss, where only residuals larger in absolute value than some positive constant contribute to the loss function.

Decision tree regression: a decision tree is built by partitioning the data into subsets containing instances with similar values. The standard deviation is used to calculate the homogeneity of a numerical sample. If the numerical sample is completely homogeneous, its standard deviation is zero. To avoid overfitting, the coefficient of deviation is used, which decides when to stop branching and the average of each branch is assigned to the related leaf node.

Random forest regression: it is an ensemble approach that takes into account the predictions of several decision trees. First, the algorithm selects K random points, then identifies the number of decision tree regressors to be created. The average of each branch is assigned to the leaf node in each decision tree. Finally, to predict output for a variable, the average of all the predictions of all decision trees are taken into consideration.

3.1.2 Unsupervised learning

Clustering

The goal of clustering is to divide data points into homogeneous groups such that the data points in the same group are as similar as possible and data points in different groups are as dissimilar as possible. Clustering algorithms are used to group data points based on certain similarities, most common algorithms are [31]:

- Density-based: clustering separates data objects based on their regions of density,

connectivity, and boundary. The clusters connect dense components, which can grow in any direction that density leads to.

- Hierarchical: this method works grouping data into a tree of clusters, starting by treating every data points as a separate cluster. Then, it repeatedly identify the two clusters which can be closest together, and merge the two maximum comparable clusters.

- Partitioning: this method classifies the information in multiple groups based on the characteristics and similarity of the data. Partitioning clustering requires a fixed number of clusters to be specified a priori. It uses an iterative process to optimize the cluster centers, as well as the number of clusters.

- Grid-based: the data space is divided into a finite number of cells that form a grid-like structure. The performance of this method depends on the size of the grid, being insufficient for highly irregular data distributions.

- Model-based: clustering assumes that the data are generated by a mixture of underlying probability distributions. This method can automatically determine the number of clusters based on standard statistics.

- Evolutionary: this clustering approaches use genetic algorithms, such as particle swarm optimization, and other evolutionary approach [4]. These are stochastic methods and use an iterative process, starting with a random population of solutions, which is a valid partition of data with a fitness value.

Dimensionality reduction

The curse of dimensionality refers to an exponential increase in the size of data needed to populate the parameter space. As a solution to this problem, dimensionality reduction is defined as the transformation of high-dimensional data into a meaningful representation of reduced dimensionality. There are different advantages when using dimensionality reduction, for example, many data mining algorithms work better if the dimensionality or the number of attributes in the data is lower, as by doing this, it is possible to eliminate irrelevant features and reduce noise. Another advantage is that a reduction of

dimensionality can lead to a more understandable model because the model involves fewer attributes. Other advantages are related to the time and memory required by the data mining algorithm which reduced and to the data being more easily visualized [202].

Some techniques for dimensionality reduction are: feature selection methods, matrix factorization, manifold learning and autoencoder methods. Matrix factorization is a method that reduce a matrix into constituent parts that make it easier to calculate more complex matrix operations [202]. The parts can be ranked and a subset of those parts can be selected to represent the dataset. Manifold learning is used to create a low-dimensional projection of high-dimensional data, often for visualization purposes, but preserving the salient structure or relationship in the data [82]. Finally, autoencoders are deep neural networks that perform dimensionality reduction, where a network model is used that seeks to compress the data to a bottleneck layer with fewer dimensions than the original input data (more details in next chapter).

Traditionally, dimensionality reduction was performed using linear techniques such as Principal Components Analysis (PCA) [130], factor analysis [193], and classical scaling [208]. PCA is a linear algebra technique for continuous attributes that find new attributes (principal components) that are a combinations of the original attributes; are orthogonal to each other, and capture the maximum amount of variation in the data. In mathematical terms, PCA is a method that projects a dataset to a new coordinate system by determining the eigenvectors and eigenvalues of the covariance matrix. It involves the calculation of a covariance matrix of a dataset to minimize the redundancy and maximize the variance. The covariance matrix is used to measure how much the dimensions vary from the mean with respect to each other. With the covariance matrix, the eigenvectors and eigenvalues are calculated and the eigenvalues are sorted in descending order. Thus, the components are in order of significance. The eigenvector with the highest eigenvalue is the most dominant principal component of the dataset. Therefore, principal components are calculated by multiplying each row of the eigenvectors with the sorted eigenvalues [211].

Generally, dimensionality reduction helps in data compression by reducing features, reducing storage, removing redundant features and noise, and tackling the curse of dimensionality. However, it may lead to some amount of information loss and sometimes

accuracy can be compromised.

3.1.3 Reinforcement learning (RL)

This is similar to sequential decision problems, except that the reward for each state are not known ahead of time, and the agent may not start out with a transition model. In this sense, RL does not necessarily know what to expect as the outcome of each action it executes. RL techniques can be classified in two main groups: active and passive.

Passive reinforcement learning

In passive RL, the agent's policy π is fixed: in state s, it always executes the action $\pi(s)$. Its goal is simply to learn how good the policy is, learning the utility function $U^{\pi}(s)$. Since the choice for each state are predetermined passive RL is not particularly useful for letting an agent learn how it should behave in an environment, but it is useful for us to learn as one step on the way to active RL.

This is operating under a stochastic environment, where a particular action executed in a particular state does not always lead to the same next state. To learn the utilities of these states under a fixed policy: (1) execute the policy a bunch of times, (2) at the end of every run, calculate the utility for each state in the sequence, and (3) update the average utility for each of the states we observed with our new data points.

Active reinforcement learning

An active agent must decide what actions to take. We want to learn utilities in order to figure out which actions are the best ones to choose. Our choices of actions are not predetermined , making an active learner more powerful. Therefore, the agent attempts to find an optimal policy by exploring different actions in the world. An important aspect of active RL is the friction between maximizing the reward for a specific state and the potential of learning new information. This is commonly known as the exploitation-exploration trade off.

3.2 Deep Learning

Neural networks are DL models that have gained attention during the last couple of years [182]. These models were proposed in 1943 and are inspired by the structure of neurons and how the brain learns [137, 65]. In 1957 Frank Rosenblatt [175] defined the perceptron as a system that illustrates the properties of intelligent systems in general. A basic perceptron consists of one or more inputs, a processor, and a single output. It contains inputs (usually given as a vector) and weights identified as $\boldsymbol{w} = (w_1, ..., w_n)$. The input $\boldsymbol{x} = (x_1, ..., x_n)$ is linearly transformed by multiplying each element x_i by its corresponding weight w_i and adding a constant factor b called the bias. This linear transformation is called the pre-activation z and it is defined as:

$$z = \sum_{i=1}^{n} w_i x_i + b = \boldsymbol{w}^T \boldsymbol{x} + b \tag{3.14}$$

For the perceptron to work as a non-linear model an activation function is applied to z, denoted by σ, producing an activation denoted by $\boldsymbol{a} = \sigma(z)$. The original idea for this activation was to emit an impulse or not depending on the value of z. A step function can be used for this purpose:

$$\sigma(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0. \end{cases} \tag{3.15}$$

when z is greater than or equal to 0 the output is 1, otherwise is 0. For a single neuron the mathematical model is defined as:

$$a = \sigma(\boldsymbol{w}^T \boldsymbol{x}). \tag{3.16}$$

Figure 3.1a shows the mathematical model of the perceptron, that receives n inputs and outputs the activation a. A nonlinear activation function turns the perceptron into a non-linear model. The learning goal is to fit the weights $\boldsymbol{w}$ and bias b according to some objective function to be optimized.

Just as with biological neurons, neural networks are modeled as layers of neurons connected to each other, where the connections are defined in terms of the weights. Multilayer

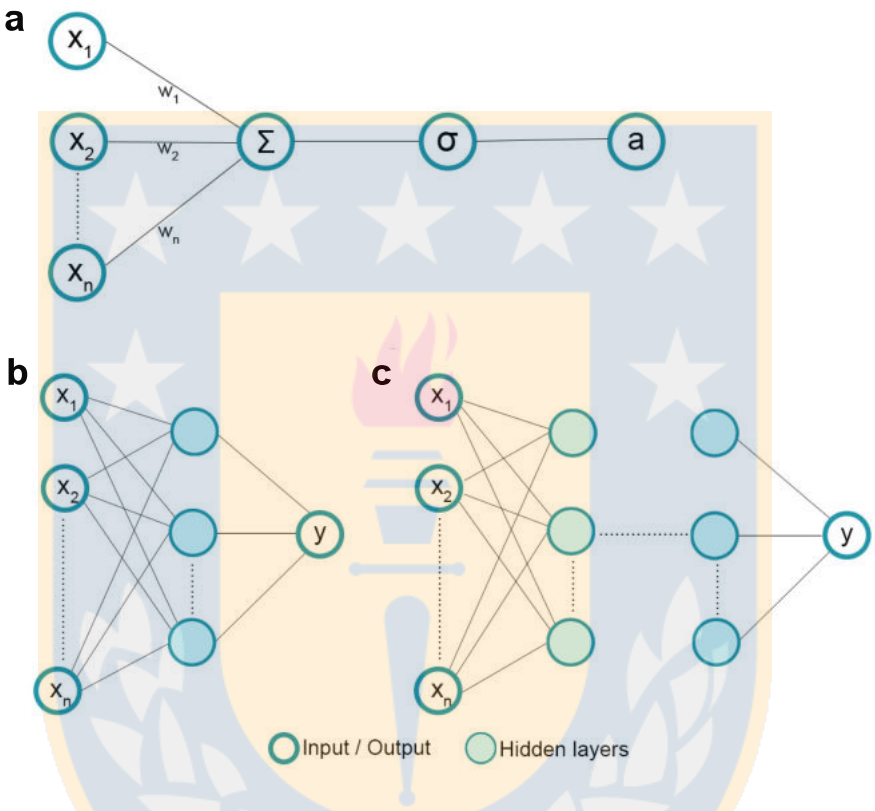


Figure 3.1: Neural networks model.
a. Perceptron model and its components: inputs ($x_1 ... x_n$), weights ($w_{11} ... w_{nm}$), activation function (σ) and output (a). **b.** Simple neural network, with one hidden layer. **c.** Deep neural network with multiple hidden layers.

perceptrons (MLPs), also referred to as feedforward neural networks, are artificial neural networks in which the connections between units do not form a cycle [59]. In a basic neural network architecture, every input unit is connected to every output unit. However, specialized networks have fewer connections, to reduce the number of parameters and the computational cost. Figure 3.1b shows a simple neural network with n inputs, a single hidden layer and an output layer that predicts y. Figure 3.1c represents a deep neural network with a similar structure that includes more hidden layers.

In 1960, the Back Propagation Model was proposed in the context of control theory [93], and later in 1961 the model was used with principles of dynamic programming [20, 45]. However, in 1980, the artificial neural network increased its popularity when Fukushima [54] proposed a multilayered artificial neural network for pattern recognition tasks.

Feedforward neural networks propagate the information from the input through each hidden unit in each layer to produce an output, a process known as forward propagation. The network parameters are usually adjusted by minimizing a loss function using gradient methods such as stochastic gradient descent [40, 6]. Backpropagation, is used to efficiently compute the gradients of the loss function [24, 109, 178, 58, 59].

A comparison between this model and the biological neurons can be made. The inputs of each artificial neuron are associated to the dendrites and the weights, to synapses. The activation of this function will depend on the stimulus received.

3.2.1 Activation functions

Each neuron computes a linear transformation of its input. The output of the neuron is the result of applying an *activation function* to such linear transformation. Examples of activation functions are shown in Table 3.1 and Figure 3.2.

Linear or identity function: it is a simple identity function $f(x) = x$, which linearly transforms the input into the output. Its range and domain are equal to $[-\infty; +\infty]$. Linear functions generate non-binary values. If only linear activation functions are used through the network, the neural network model is equivalent to a simple linear transformation of the input.

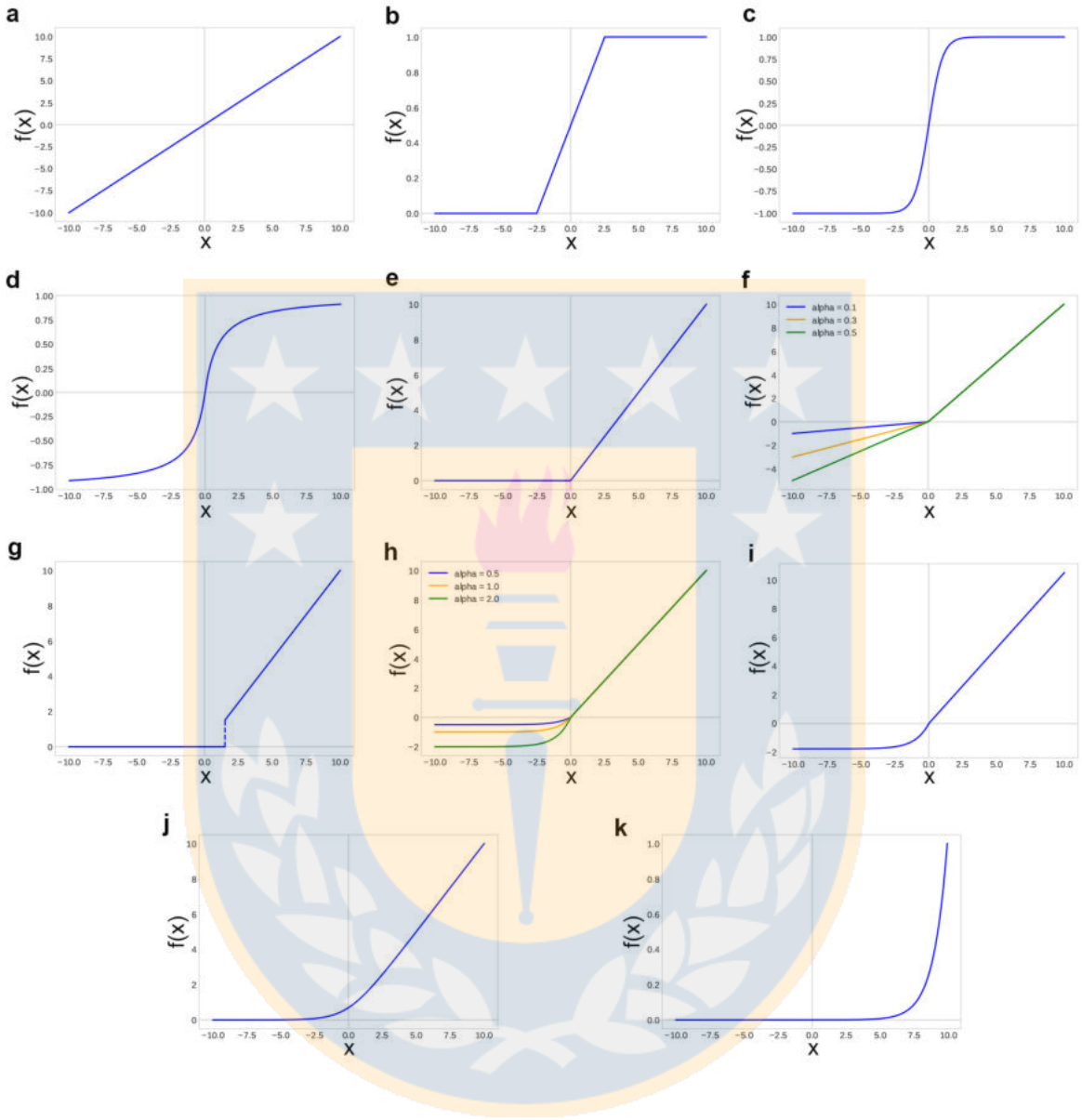


Figure 3.2: Activation functions.
Shape of **a.** Linear, **b.** Hard sigmoid, **c.** Hiperbolic tangent, **d.** SoftSign, **e.** Rectified linear unit, **f.** Leaky ReLU, **g.** Thresholded ReLU, **h.** ELU, **i.** SELU, **j.** Softplus, and **k.** Softmax activation functions.

Table 3.1: Activation function summary.

Function	Equation
Linear	$f(x) = x$
Hard sigmoid	$f(x) = max\left(0, min\left(1, \frac{(x+1)}{2}\right)\right)$
Hiperbolic tangent	$tanh(x) = \frac{e^x - e^{-x}}{e^a + e^{-x}}$
SoftSign	$f(x) = \frac{x}{1+\|x\|}$
Rectified linear unit	$f(x) = max(0, x)$
Leaky ReLU	$f(x) = \alpha x + x = \begin{cases} x, & \text{if } x > 0 \\ \alpha x, & \text{if } x \leq 0 \end{cases}$
ELU	$f(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha(e^x - 1) & \text{if } x \leq 0 \end{cases}$
SELU	$f(x) = \lambda \begin{cases} x, & \text{if } x > 0 \\ \alpha(e^x - \alpha) & \text{if } x \leq 0 \end{cases}$
Softplus	$f(x) = log(1 + e^x)$
Softmax	$f(x) = \frac{e^{a_i}}{\sum_j e^{a_j}}$

Logistic sigmoid function: this function returns values ranging from 0 to 1. It is a non-linear activation function that gives an 'S' shaped curve (hence the name sigmoid). The function is differentiable i.e., the slope of the sigmoid curve can be calculated at any two points [70]. One of its advantages is that it produces the ϕ parameter of a Bernoulli distribution, generating a value from 0 to 1 although its derivatives are closer to 0 for small or large inputs.

Hard sigmoid: it is a linear approximation of the sigmoid function. The range in this case is equal to 0 for an input between $(-\infty, -2.5)$, increasing linearly from 0 to 1 for an input between $[-2.5, 2.5]$ and being equal to 1 for an input between $(2.5, +\infty)$. The advantage of this function over the logistic sigmoid function is that hard sigmoid is faster to compute, as there is no need to calculate exponential functions. However, the derivative of this function is not continuous and is 0 for input values below -2.5 or higher than 2.5 [37].

Hyperbolic tangent: this is a sigmoid function ranging from -1 to 1. Generally, it is used for categorizing two classes. The *tanh* function is also sigmoidal ('S'-shaped), but the range is between (-1 to 1). This is an advantage compared to the sigmoid function because of its steeper derivative, which makes it more efficient in a wider

range for faster learning and grading. The negative inputs are mapped strongly negative and the zero inputs are mapped near zero in the *tanh* graph. It is a differentiable and monotonic function, but it is not derivative [15].

SoftSign: this is a sigmoid function with range (-1, 1) that converges polynomially as opposed to the *tanh* function, which converges exponentially.

Rectified linear unit (ReLU/ELU): this function behaves linearly for positive values in the domain preserving many properties of linear models. It truncates negative values to zero which is why its range is $(0, +\infty)$. ReLUs accelerate the convergence of stochastic gradient descent and do not activate all the neurons at the same time. Returning zero for negative values causes the function to be non-differentiable at the origin [149, 129].

Leaky ReLU: this function is a variant of the ReLU, the difference is that it introduces a small negative slope to the normal ReLU, that helps updating the weights during the backpropagation process. By introducing an α parameter, the gradients do not turn into zero during training. The range for this function is $(-\infty, +\infty)$.

Thresholded ReLU: another variant of ReLU, that is activated only if the input is above some threshold specified by the user. In this sense, the output is 0 for $x < \theta$, and equal to x if $x \geq \theta$ [103].

Exponential Linear Unit (ELU): this is another variant of the ReLU, generally used to accelerate the training of neural networks. It outputs a linear value for non-negative inputs but uses a monotonically increasing function for negative values. The goal of ELUs is to push the mean unit activation closer to 0. This reduces the computational complexity and improves the learning speed thanks to a more robust representation and a better generalization when compared to regular ReLUs and Leaky ReLUs. At the same time, ELUs solve the vanishing gradient problem by using the identity for positive values and an α parameter that controls the saturation point for negative inputs [34].

Scaled Exponential Linear Unit (SELU): it is another variant of ReLUs. This activation function scales positive and negative values. Its activation converges toward

Table 3.2: Loss functions summary

Function	Equation
Mean Absolute Error or L1	$S = \sum_{i=1}^{n} \lvert y_i - \hat{y}_i \rvert$
Mean Squared Error or L2	$S = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$
Cross entropy	$S = -\frac{1}{n} \sum_{i=1}^{n} [y_i log(\hat{y}_i) + (1 - y_i) log(1 - \hat{y}_i)]$
Hinge loss	$Hinge\, loss = \sum_i max(0, 1 - y_i * h_\theta(x_i))$
Kullback-Leibler Divergence	$D_{KL}(p\|\|q) = \mathbb{E}_{x\sim p}\left[\log \frac{p(x)}{q(x)}\right] = \mathbb{E}_{x\sim p}\left[\log p(x) - \log q(x)\right]$

zero mean and unit variance when propagated through multiple layers during network training, thus learning robust features. Another advantage is that SELUs are not affected by vanishing and exploding gradients [101].

Softplus: since it represents a smooth approximation of the ReLU, it is also known as SmoothReLU. Softplus produces outputs between (0, $+\infty$) [47]. An advantage of Softplus is that it improves the model performance with fewer epochs needed to converge during training.

Softmax: commonly used in the final layer of a network designed for multiclass classification problems. It represents the probability distribution of a categorical variable over n different classes by assigning decimal probabilities to each class, which add to 1 [59].

3.2.2 Loss function

When training a neural network, a loss function is minimized. This loss measures the quality of a set of parameters for solving the task the neural network was designed for. There are different loss functions according to the type of problem that must be addressed. Below we describe a set of commonly used loss functions and Table 3.2 shows their corresponding equations [88].

Mean Absolute Error or L1 loss function: used to minimize the absolute differences between the target value (y_i) and the estimated values ($\hat{y}_i$).

Mean squared error or L2 loss function: used to minimize the squared differences between the target value (y_i) and the estimated values ($\hat{y}_i$).

Cross entropy loss: it measures the performance of a probabilistic classification model and increases as the predicted probability diverges from the ground truth label.

Hinge loss: used for binary classification. The hinge function is a convex surrogate loss function for the 0-1 loss and it is the basis of the widely-used support vector machine model.

Kullback-Leibler Divergence (KL Divergence): is a measure of how one probability distribution differs from a baseline distribution. A KL divergence loss of 0 suggests that the distributions are identical. It calculates how much information would be lost if the predicted probability distribution is used to approximate the desired target probability distribution. The KL divergence can be used for multi-class classification, in which case it is functionally equivalent to multi-class cross-entropy. Table 3.2 shows the equation for two separate probability distributions $p(x)$ and $q(x)$ over the same random variable x.

3.2.3 Gradient descent

Gradient descent (GD) is used to minimize an objective function $J(\boldsymbol{\theta})$ parameterized by the parameters of the neural network $\boldsymbol{\theta}$. For functions with multiple inputs, GD uses the gradient of J (vector that contains all the partial derivatives of J with respect to $\boldsymbol{\theta}$ denoted as $\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$) to update the parameters iteratively in the direction of the steepest descent. This is achieved by following the negative of the gradient as shown in Figure 3.3 [19].

Gradient descent present variants that differ in how much data is used to compute the gradient of the objective function. Due to the limited amount of data, in practice GD can be performed in different ways as described below:

Batch gradient descent: it computes the gradient of the cost function to the parameters $\boldsymbol{\theta}$ for the entire training dataset. The size of the steps is known as the learning rate η and it defines how much the parameters opposed to the gradient at iteration t should be moved in order to advance towards the minimum. Parameters are updated by

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^{t} - \eta \cdot \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})|_{\boldsymbol{\theta}^{t}} \tag{3.17}$$

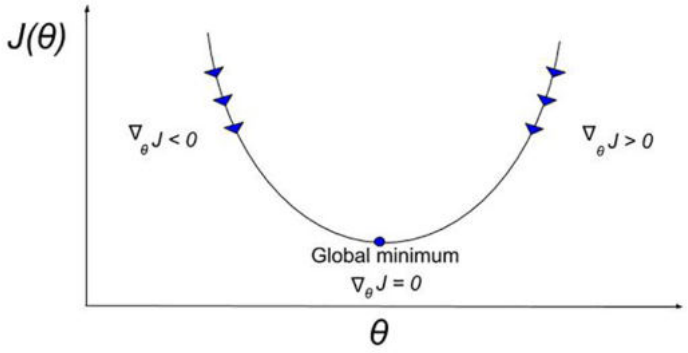


Figure 3.3: Gradient descent.
Example of how gradient descent uses the derivatives to reach the global minimum.

Batch gradient descent is computationally efficient since it produces a stable error gradient and convergence. It will converge to the global minimum if the loss function is convex and may converge to a local minimum if the loss function is non-convex.

However, it might be impractical for large datasets given that batch gradient descent needs to calculate the gradients of the loss function for all the data. Similarly, it has been shown that batch gradient descent has a slow convergence when compared to the other methods described below [177].

Stochastic gradient descent (SGD): this variant performs a parameter update for each training example x_n and label y_n, and learning is performed for every example:

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^{t} - \eta \cdot \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}^{t}; \boldsymbol{x}_n; y_n) \tag{3.18}$$

Generally, it is used to learn online and faster given that SGD performs one update at a time and with a high variance that causes the objective function to fluctuate [19]. In spite of ameliorating the model due to its frequent updates, SGD increases the run time which renders it computationally expensive. A disadvantage is that this algorithm may converge to a local minimum and present a high variance, without reaching the global optimal result.

Mini-batch gradient descent: this is a combination of two types of gradient descent: batch and stochastic. Mini-batch gradient descent is useful technique that splits the

training dataset into small batches and implement GD on each batch one after the other. It performs an update for every mini-batch of n training examples, defined as:

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^{t} - \eta \cdot \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}^{t}; \{\boldsymbol{x}_n, y_n\}_{n \in \mathcal{M}_m}), \tag{3.19}$$

where $\mathcal{M}_m$ corresponds to the set of indexes of the m mini-batch. This allows a variance reduction in the parameter updates. Usually the batch size is a power of 2 achieving better run time than common batch sizes. Mini-batch gradient descent also helps avoiding local minima while converging at a higher speed than batch gradient descent since it uses less examples. All of this results in a more stable convergence while getting closer to the global minimum [177, 117].

Setting parameters and choosing a proper learning rate can be a challenging task. Therefore, different gradient descent variants have been proposed to improve the learning performance. These optimizers work by modifying the learning rate component, the gradient component or both [59]. Some examples are [177]:

Momentum: this method proposed in 1999 [169] was designed to accelerate learning and primarily aims at solving the poor conditioning of the Hessian matrix and variance in the stochastic gradient. Momentum adds a fraction γ of the updated vector of the past time step $\boldsymbol{v}_{t-1}$ to the current update vector $\boldsymbol{v}_t$, as shown in Equation 3.21:

$$\boldsymbol{v}_t = \gamma \boldsymbol{v}_{t-1} + \eta \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}), \tag{3.20}$$

$$\boldsymbol{\theta} = \boldsymbol{\theta} - \boldsymbol{v}_t. \tag{3.21}$$

Nesterov Accelerated Momentum [150]: the difference with the standard momentum method resides in the moment in which the gradient is evaluated after the current velocity is applied. The idea is to look at a point to which current momentum is pointing and compute gradients from that point. In Equation 3.23 computing $\boldsymbol{\theta} - \boldsymbol{v}_{t-1}$ give us an approximation of the next position of the parameters.

$$\boldsymbol{v}_t = \gamma \boldsymbol{v}_{t-1} + \eta \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta} - \gamma \boldsymbol{v}_{t-1}) \tag{3.22}$$

$$\boldsymbol{\theta} = \boldsymbol{\theta} - \boldsymbol{v}_t \tag{3.23}$$

AdaGrad (Adaptive gradient) [128]: is an algorithm where the learning rate is adjusted separately for each parameter on each step. It modifies individual learning rates keeping the sum of squares of parameter-wise gradients. When gradient change is small, the learning rate is slightly affected by the algorithm and it moves towards the optimum faster. When the gradient is large the learning rate further decreases as defined in Equation 3.24:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \frac{\eta}{\sqrt{G_t + \epsilon}} \cdot \odot \boldsymbol{g}_t, \tag{3.24}$$

where $\boldsymbol{g}_t$ is the gradient at time t, G_t is a diagonal matrix where each diagonal element contains the sum of the squares of the gradients with respect to the network parameters $\boldsymbol{\theta}$ at time t, ϵ is a small constant that avoids division by zero, and $\odot$ is the matrix-vector product. Notice that by performing this operation, the learning rate now is scale differently for each parameter and is inversely proportional to the square root of G_t.

RMSprop (Root Mean Square Propagation) [145]: this method uses an adaptative learning rate that modifies AdaGrad to perform better in a nonconvex setting. It works dividing the learning rate for a weight by a running average of the magnitudes of recent gradients for that weight.

$$v_t = \gamma v_{t-1} + (1 - \gamma) \cdot (\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}))^2, \tag{3.25}$$

$$\boldsymbol{\theta} = \boldsymbol{\theta} - \frac{\eta}{\sqrt{v_t + \epsilon}} \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}), \tag{3.26}$$

where γ is a hyperparameter that weights the contribution of $\boldsymbol{v}_{t-1}$ and the square of the gradient to $\boldsymbol{v}_t$, and ϵ is a small constant that avoids division by zero.

Adam (Adaptive Moment Estimation) [97]: computes adaptive learning rates for each parameter. It is considered to be a combination of momentum and RMSProp that, besides using the decaying average of past squared gradients for parameter-specific learning rates, it also employs a decaying average of past gradients in place of the current gradient. Formally it is shown in Equation 3.27:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon}\hat{m}_t, \tag{3.27}$$

where $\hat{v}_t$ and $\hat{m}_t$ are actually bias-corrected averages to ensure that the values are not biased towards 0.

Other examples of gradient descent optimization algorithms are: AdaDelta [224], AdaMax [97], Nadam [116], AMSGrad [216] which are variations of the algorithms explained above.

3.2.4 Regularization

An important desired feature of deep learning models is their ability to predict with low error both on the data used for training, and on new data unseen by the model. This ability to generalize is often achieved by regularization techniques. Goodfellow et al., [59] defined regularization as "any modification we make to the learning algorithm that is intended to reduce the generalization error, but not its training error". Different strategies are used to reduce the test error and maintain accuracy. As artificial neural network models are usually composed of many parameters, they tend to overfit to the training data, but not predict correctly on unseen data.

Some of the most common strategies for regularization are:

L1 and L2 regularization: these are commonly known as weight decay, and modify the general cost function by adding a "regularization term". These terms force the model to obtain small value for their parameters, hence, diminishing the its complexity and therefore reducing overfitting.

L1 penalizes the absolute value of the weights and tends to drive some weights to zero. It is also known as Lasso regression, and it is defined as in the Equation 3.28:

$$\tilde{J}(\boldsymbol{x}, y) = J(\boldsymbol{x}, y) + \lambda \sum_i |\boldsymbol{\theta}_i|, \tag{3.28}$$

where $J(\boldsymbol{x}, y)$ is the error and λ is the regularization parameter.

L2 regularization penalizes the square value of the weights and tends to drive all the weights to smaller values [151], as defined by the Equation 3.29:

$$\tilde{J}(\boldsymbol{x}, y) = J(\boldsymbol{x}, y) + \lambda \sum_i \boldsymbol{\theta}_i^2, \tag{3.29}$$

where $J(\boldsymbol{x}, y)$ is the error and λ is the regularization parameter.

Dropout: regularization strategy that at each training iteration randomly selects some nodes and removes them from the network including its connections. Dropout can be applied to hidden or input layers. Each iteration has a different set of nodes and finally different set of outputs, producing a more robust model by avoiding co-adaptation. The probability of choosing the number of nodes to be dropped is a hyperparameter of the function. Dropout is equivalent to sampling from an exponential number of networks which helps reducing overfitting [194].

Data augmentation: using as much data as possible to train the network helps the model to generalize better. However, data is limited and this is why synthetic data with similar-to-real-data variations is created for some machine learning problems. Data augmentation is a method that increase the dataset size as a data-space solution for limited data problem. It has been effective for classification problems, especially for object recognition [59]. Moreover, it has been applied to images [165] and speech recognition tasks [86] with satisfactory results.

Early stopping: when training deep neural networks using GD methods, the model fits the data iteratively. During the first learning steps GD improves the performance of the model for data outside the training set, but after a point, it starts overfitting to the training data. Early stopping works by keeping a subset of the training set as the validation set and then interrupting the training procedure when the performance on the validation set worsens. Under this strategy it is possible to get

models with better validation set error and therefore, a better generalization error, as the validation set was not used to fit the parameters [220].

3.2.5 Neural network architectures

Convolutional Neural Network (CNN): This architecture is a neural network for processing data that has a known grid-like topology (e.g. images). This network employs convolutions which are a specialized kind of linear operation. A typical CNN network has three main types of layers: Convolutional Layers, Pooling Layers and Fully-Connected Layers [48]. The convolutional layer is the core building block of this type of network, and does most of the computational heavy lifting. The element responsible for carrying out the convolution operation in the first part of the convolutional layer is called the Kernel/Filter (K). The goal of this operation is to extract the high-level features from the input [1]. Pooling layers are in charge of reducing the spatial size of the convolved feature and extracting dominant features [104]. A pooling function replaces the outputs of the network at a certain location with a summary statistic of the nearby outputs [55]. Max Pooling layers are commonly used. They output the maximum within a rectangular neighborhood [148]. Other pooling functions calculate the average of a rectangular neighborhood (average pooling) [225]. Convolutional and pooling layers are usually combined in a CNN architecture which is then fed to a fully-connected architecture used to learn the prediction task from the high-level features as represented by the output of the convolutional architecture [46]. Figure 3.4a shows a visual representation of these networks and how they work. First square represents the input. The following two columns represent the convolutional and pooling layers (i.e. the convolutional architecture) followed by a hidden fully connected layer before the output (last column with three outputs).

CNNs are a good architecture for processing spatial information for 2D and 3D images, given that digital images store pixel values in a two-dimensional grid of data to which convolutions can be naturally applied [219]. The kernel is a multidimensional array of parameters that is directly learned from the data working as an optimized feature extractor that is applied in each position of the image. Among all the types of cancer studied, this is the most used architecture (77.5%) in the analysis of medical images.

Some variations of this architecture are: Spatially Constrained Convolutional Network (SC-CNN) [192], Multiresolution Convolutional Network (MR-CN) [115], Fully Convolutional Network (FCN) [122], among others.

Recurrent Neural Network (RNN): RNN are a type of neural network meant to analyze sequential data. They operate over sequences of vectors, and their structure includes cyclic connections. They differ from other neural networks in that they possess a circuit involving hidden to hidden recurrences which serves as temporal memory for the networks [142]. Figure 3.4b shows a graph of this network with a recurrent hidden state (curve arrow). A simple recurrent network is modeled by:

$$\boldsymbol{h}_{(t)} = g(b + U\boldsymbol{h}_{t-1} + W\boldsymbol{x}_t), \tag{3.30}$$

$$\boldsymbol{o}_{(t)} = c + V\boldsymbol{h}_{(t)}, \tag{3.31}$$

where t is the time step, W represent the weights connecting the hidden units $\boldsymbol{h}_t$ and the input units $\boldsymbol{x}_t$, and V represent the weights connecting the hidden units $\boldsymbol{h}_t$ with the output units $\boldsymbol{o}_t$; b and c are the offsets of the hidden and output layers; g is the activation function and U represents the weights connecting hidden units at time $t-1$ to hidden units at time t.

Generative Adversarial Networks (GAN): These networks are generative models that aim at simulating data that follow the same distributrion as the real data. GANs make use of two adversarial models: a generative model G that captures the data distribution given a random input variable z, and a discriminative model D that estimates the probability that a sample came from the training data rather

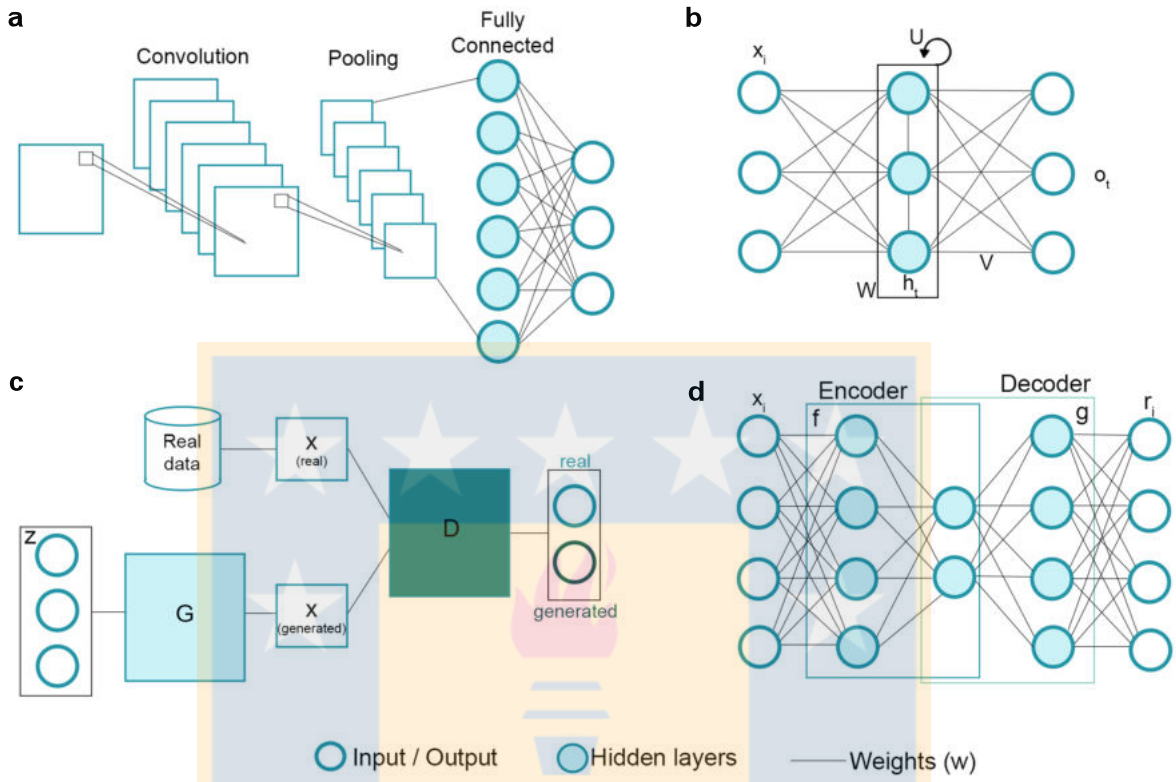


Figure 3.4: Visualization of neural networks architecture.
First and last columns at each sample represent input and output, and colored sky-blue dots represent hidden layers. **(a)** Convolutional neural network (CNN) the convolution is performed on the input with the use of a filter or kernel to then produce a feature map. Pooling layers are added to reduce the number of parameters and computation in the network. The last layers are fully connected layers that have full connections to all the activations in the previous layer including the output. **(b)** Recurrent neural network (RNN) in this network the information cycles through a loop. When it makes a decision, it takes into consideration the current input and also what it has learned from the inputs it received previously. **(c)** Generative Adversarial Network (GAN) is composed of a generator and a discriminator. The input for the generator is a noise source and the discriminator takes as inputs real and generated examples, to finally distinguish between the two sources (real/fake). **(d)** Autoencoder (AE) this network works by compressing the input into a latent-space representation, and then reconstructing the output from this representation.

than G. Thus, D tries to discriminate samples as accurately as possible, while G tries to generate data that the discriminator is not able to correctly distinguish as non-real [60].

To define how a GANs work, it is necessary to consider three distributions:

- $p_z(z)$: distribution of the noise input z,
- $p_g(x)$: the distribution over the generated data, and
- $p_r(x)$: distribution of the real data x.

To ensure the decisions of discriminator D over real data are accurate we maximize $\mathbb{E}_{x\sim p_r(x)}[logD(x)]$. At the same time, given a fake sample $G(z)$, for $z \sim p_z(z)$, the discriminator is expected to output a probability, $D(G(z))$, close to zero by maximizing $\mathbb{E}_{z\sim p_z(z)}[log(1 - D(G(z)))]$. As for the generator, it is trained to increase the chances of D producing a high probability for a fake example, thus to minimize $\mathbb{E}_{z\sim p_z(z)}[log(1 - D(G(z)))]$. Therefore, the function to be maximized by the discriminator and minimized by the generator is:

$$\begin{aligned} V(D,G) &= \mathbb{E}_{x\sim p_r(x)}[logD(x)] + \mathbb{E}_{z\sim p_z(z)}[log(1 - D(G(z)))] \\ &= \mathbb{E}_{x\sim p_r(x)}[logD(x)] + \mathbb{E}_{x\sim p_g(x)}[log(1 - D(x)]. \end{aligned} \tag{3.32}$$

A visualization of how these networks work is depicted in Figure 3.4c where the architecture is composed of two networks: a generator and a discriminator (hidden layers). The generator generates data, and the discriminator uses sample and real data to classify data in generated or real.

AutoEncoder (AE): This neural network is trained to encode its input into a latent embedded space and reconstruct from this embedding a representation as close as possible to the input. The architecture may be viewed as an encoding function and a decoder that produces a reconstruction [14]. For input data $\boldsymbol{x}_i$ these networks learn an encoder function f and decoder function g such that the output $\boldsymbol{r}_i = g(f(x_i))$ is similar to $\boldsymbol{x}_i$. An autoencoder is trained by minimizing a loss function that aims at measuring the difference between $\boldsymbol{x}_i$ and $\boldsymbol{r}_i$, such as $\mathcal{L}(\boldsymbol{x}_i, \boldsymbol{r}_i) = \|\boldsymbol{x}_i - \boldsymbol{r}_i\|^2$.

Usually, the embedding space is smaller than the input, obtaining a compressed representation of the data and learning correlations which facilitate tasks such as classification, visualization, communication and storage of the data [78]. Figure 3.4d shows a general structure of an AE where the dimensionality of the input is the same as the dimensionality of the output, and the decoder creates a reconstruction of the input from the hidden layer.

Some examples of AE architectures are: denoising autoencoder (DAE) [213, 96], and stacked denoising autoencoder (SDAE) [214, 124] or deep autoencoder [78], among others. The first one, is an AE that receives a corrupted data point as input and is trained to recover the original undistorted data as its output; the second one is a DAE with multiple hidden layers; and finally, the third one is composed of two symmetrical networks that have shallow layers constituting the encoder and a second set of layers used as a decoder.

Chapter 4

Analysis of tumor-infiltrating T cell across different types of cancer supported by machine learning tools

Cancer is derived from our own cell, and the immune responses for cell growth present a big challenge. However, genetic engineering of T cells[1] can be used therapeutically to overcome the challenges. The importance of T cells can be due to it can be modified with genes encoding receptors that recognize cancer-specific antigens, also several hundred billion T cells reside in our lymphoid tissues and circulate through the bloodstream to detect and destroy diseased cells.

Single cell profiling (scRNA-seq[2]) has enabled high resolution mapping of cellular heterogeneity, development, and activation states in diverse systems. This approach has been applied to analyze human T cells in diseased tissues and in response to immunotherapies in cancer, for this reason we selected this type of data for our analysis.

CD8+ and CD4+ T cells play a key role in cellular immune responses against cancer by cytotoxic responses and effector lineages differentiation, respectively. These subsets have been found in different types of cancer; however, it is unclear whether tumor-infiltrating T cell subsets exhibit similar transcriptome[3] profiling across different types of cancer in comparison with healthy tissue-resident T cells. Thus, in this section, the aim of this study was to analyze the single cell transcriptome of tumor-infiltrating CD4-T, CD8-T and Tregs (subpopulations of T cells) obtained from different types of cancer to identify specific pathways for each subset in malignancy. An in-silico analysis was performed from scRNA-seq data available in public repositories (Gene Expression Omnibus) including

[1]T cells are part of the immune system and develop from stem cells in the bone marrow. They help protect the body from infection and may help fight cancer. Also called T lymphocyte and thymocyte.

[2]Single-cell sequencing is a next-generation sequencing (NGS) method that examines the genomes or transcriptomes of individual cells, providing a high-resolution view of cell-to-cell variation.

[3]A transcriptome is the full range of messenger RNA, or mRNA, molecules expressed by an organism. The term can also be used to describe the array of mRNA transcripts produced in a particular cell or tissue type.

breast cancer, melanoma, colorectal cancer, lung cancer and head and neck cancer. After dimensionality reduction, clustering and selection of the different subpopulations from malignant and non-malignant datasets, common genes across different types of cancer were identified and compared to non-malignant genes for each T cell subset to identify specific pathways. Our data revealed that tumor-infiltrating T cells exhibit 38 exclusive pathways in CD4+ cells, 72 exclusive pathways in CD8+ cells and 100 exclusive pathways in Tregs. In addition, we also identified 31 common pathways for the three T cell subsets in malignant tissues, including viral infection and metabolism. In summary, our analysis allowed to integrate a large amount of data using machine learning in order to identify common genetic T cell signatures across different types of cancers. Therefore, potential immunomodulatory therapies associated with these pathways could be applied to different types of tumors.

4.1 Related work

The main role of the immune system is to protect the body against infections and abnormal cell growth. One of the main players in the recognition of pathogenic antigens or neoplastic transformation are T-lymphocytes. This subset has been shown to be involved in the regulation of the immune response by operating both cellular and humoral immunity [204]. T-lymphocytes can be divided into two main subpopulations, CD8+ cytotoxic T- and CD4+ helper T cells. CD8-T cells are characterized by inducing cell-mediated lysis during viral infection and malignancy [138, 51]. On the other hand, CD4-T cells play an important role in the adaptive immune system by inducing a regulated effective response to pathogens, associated with a cytokine profile and the modulation of other subsets such as macrophages, B cells and NKs [105, 125]. CD4-T cells can be divided into effector T-helper lineages such as Th1, Th2 and Th17 and regulatory T cells (Tregs). Tregs are a subpopulation of CD4-T cells that maintain self-tolerance and modulate the immune system by controlling pro-inflammatory responses via different suppressive mechanisms [69].

In cancer, the cytotoxic responses from CD8-T cells and effector Th1 and Th17 cells have been considered protective in terms of tumor development [157]. By contrast, the presence of Th2 and Tregs has been associated with bad prognosis [110, 21]. The balance

between these effector and regulatory responses can be affected by cancer cells that promote phenotypic changes [170, 66] and migration [85] of regulatory subsets that inhibit anti-tumor pro-inflammatory responses. Therefore, it is crucial to identify whether Tregs, CD4-T and CD8-T cells surrounding tumors exhibit a common specific genetic signature in comparison with tissue-resident T cell subsets from healthy volunteers across different types of cancer. This information will lead us to the mechanism by which the tumors command Tregs, CD4-T and CD8-T signaling pathways as well as the identification of potential specific responses aimed at predicting the efficacy of clinical therapies for cancer treatment [230, 197].

During the last decades, the former T cell subsets have been identified with the analytical method of flow cytometry by using fluorescent-labelled antibodies against proteins such as CD3, CD8, CD4, CD25, CD127 and FOXP3 [84]. This technique is widely used in clinical samples for the monitoring of these cells in cancer immunotherapies [179, 171]. However, nowadays, novel genetic sequencing techniques have allowed the identification of T cells based on their genetic signature, including the same markers classically used in flow cytometry. It has been shown that the identification of T cells by scRNA-seq can provide not only the identification of these cells, but also new biological pathways related with their function [56, 203].

scRNA-seq allows to obtain a full genetic description of single cells in comparison with massive sequencing. However, scRNA-seq contains more noise than the analysis of massive sequencing [198], due to a greater amplification of the genetic material and a smaller number of samples. Despite that, methods aimed at reducing dimensionality and identifying subpopulations, as well as clustering methods from machine learning, have improved the analysis to get reliable single T cell data [100].

Common machine learning methods have been applied to scRNA-seq data in order to reduce their dimensionality and identify subpopulations. Such methods include Principal Component Analysis (PCA) [10] which aims at reducing the data dimensionality by calculating a transformation of the data into a set of linearly uncorrelated values called principal components. PCA is a simple and very useful tool for examining heterogeneity in scRNA-seq data [166, 118, 176]. Recently, t-distributed stochastic neighbor embedding (t-SNE), has also been applied to dimensionality reduction in scRNA-seq data [206, 136, 57].

t-SNE is a stochastic method for dimensionality reduction originally aimed at visualizing high dimensional data. It is a non-linear dimensionality reduction technique that finds a lower dimensional space in which similar objects are close and dissimilar objects are distant with high probability [72]. Clustering techniques can then be applied to the reduced dimensionality space in order to find groups of similar cells [100].

In this study we have analyzed publicly available scRNA-seq data from CD4-T, CD8-T and Treg cells isolated from melanoma [206, 114], breast [9, 32], lung [199, 68], colorectal [228] and head and neck [168] cancer. We identified common genes between tumor-infiltrating T cell subsets. We compared these genes with the genetic profile of the same non-malignant tissue resident subsets. For malignant-related T cells subsets, results showed that 652 genes in CD4-T, 69 genes in CD8-T cells and 557 genes in Treg, were common between the different tumors, but different from genes from non-malignant samples. In terms of pathway analysis, specific gene discrimination between malignant and non-malignant samples revealed unique immune response pathways in CD4-T, CD8-T and Tregs associated with metabolism and immunoregulation. Altogether the datasets analysis revealed that tumor-infiltrating T cell subsets exhibit similar and unique genetic signatures across different types of cancer in comparison with tissue resident subsets.

4.2 Our Approach

4.2.1 Data collection and pre-processing

Data collection from previous scRNA-seq expression profiles from malignant and non-malignant cells were included in the analysis. In some cases, non-malignant samples were obtained from the adjacent normal tissues. We selected scRNA-seq data from isolated cells from breast (GSE114727 and GSE75688), lung (GSE126030 and GSE99254), colorectal (GSE108989), melanoma (GSE72056 and GSE123139), and head and neck cancer (GSE103322). The datasets were obtained from the Gene Expression Omnibus (GEO) repository and all of them were sequenced on Illumina HiSeq2500/HiSeq4000 or Illumina NextSeq 500 (Homo sapiens) with a similar experimental design. We verified the quality of each sequencing library with FastQC [5], a software package that estimates the number of un-callable and low quality bases. Mapping to the human reference genome (hg38) was

done using STAR [41], a high performance community-standard aligner.

Each dataset was analyzed separately as a digital expression matrix. We used transcripts per million (TPM) values as gene expression levels for all the analysis, calculated as:

$$\frac{10^6 \cdot C_{ij}/\text{length of gene i}}{\sum_i C_{ij}/\text{length of gene i}},$$

where C_{ij} is the count value of gene i in cell j. We removed genes with low expression values, considering as cutoff the upper median TPM values [83].

4.2.2 T cell identification

Datasets GSE126030, GSE99254 and GSE108989 contain only T cells. Datasets GSE114727, GSE75688 and GSE123139 contain different type of cells, although each of them is detailed in separate files available in the Gene Expression Omnibus (GEO) repository, indicating explicitly which of them are T cells.

To identify the different cells for the datasets GSE72056 and GSE103322 we started by using PCA and variance analysis in order to obtain the value of the optimal number of components using the Scikit-learn implementation of PCA [162]. We identified eight components in GSE72056 and nine components in GSE103322.

We used agglomerative clustering to define subsets of cells and for assigning their labels (T, B, macrophages, endothelial, cancer-associated fibroblasts (CAFs) NK, mast and myocytes cells). We used t-SNE [210] for dimensionality reduction in order to visualize the cells in a two-dimensional scatter plot. Following [206] and [168], we used six cluster for GSE72056 and eight for GSE103322 in order to cluster the same number of cells.

To identify the T cell cluster in GSE72056 and GSE103322 we removed all the cells that has no expression for cell markers and those were the ones we used in this analysis.

4.2.3 Analysis of T cells subpopulations

In order to identify our target genes, we classified subpopulation of T cells according to the following gene selection criteria: (1) for CD4-T cells *cd3g*>0, *cd8*=0, *cd4*>0 and *foxp3*=0; (2) for CD8-T cells, *cd3g*>0, *cd8*>0, *cd4*=0, *foxp3*>0; and (3) for Tregs *cd3g*>0, *cd8*=0, *cd4*>0 and *foxp3*>0.

To identify the T cell subpopulations included in our analysis, we generate a multiple list comparator with the name of the genes in each condition, to finally extract which genes are common across malignant samples, non-malignant samples and different between malignant and non-malignant samples.

4.2.4 Pathways and GO categories analysis

A pathway enrichment analysis was performed using the Gene Ontology Consortium database (data-version from 2020-05-02) [8, 36]. This database includes information about biological processes, molecular functions and cellular components. Reactome pathway database [89] was also used to identify pathway enrichment due to its database of human pathways, reactions and processes allowing an orientation and model in biological pathways that include classic intermediary metabolism, signaling, innate and adapted immunity, transcriptional regulation, apoptosis and diseases that are highly expressed in our data. Exclusive pathways were identified using InteractiVenn [74].

To visualize the list of GO terms and find how genes are functionally grouped we use Cytoscape v.3.8.2 with the plugin ClueGO v.2.5.7 [17] with a ($p<0.001$) and kappa statistics to calculate the relationships between the terms based on the similarity of their associated genes.

4.3 Results

Figure 4.1 illustrates the bioinformatics pipeline for the analysis and identification of exclusive T cell subsets transcriptomic pathways in malignancy. Briefly, the scRNA-seq datasets from breast (GSE114727 and GSE75688), lung (GSE126030 and GSE99254), colorectal (GSE108989), melanoma (GSE72056 and GSE123139), and head and neck cancer (GSE103322) were obtained from the Gene Expression Omnibus (GEO) repository. Each dataset was analyzed separately as a digital expression matrix and transcripts per million values were used as gene expression levels (Figure 4.1a). Once the datasets were analyzed, we reduced dimensionality and clustered the different cell types in order to identify the T lymphocytes (Figure 4.1b). We then analyzed the transcriptome of CD4+, CD8+ and Tregs from malignant and non-malignant samples (Figure 4.1c) and identified common genes across the different types of cancer for each T cell subset (Figure 4.1d). Common

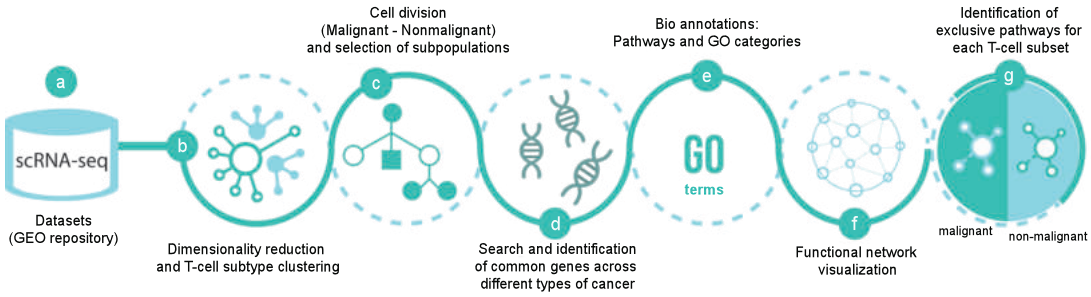


Figure 4.1: scRNA-seq pipeline.
(a) Datasets of scRNA-seq were selected from Gene Expression Omnibus (GEO) database from breast (GSE114727 and GSE75688), lung (GSE126030 and GSE99254), colorectal (GSE108989), melanoma (GSE72056 and GSE123139), and head and neck cancer (GSE103322). **(b)** The data was processed using dimensionality reduction and clustering techniques in order to separate T cells and its subpopulations CD4-T, CD8-T and Treg **(c)** from malignant and non-malignant origin. **(d)** Then, common genes across the different types of cancer from malignant and non-malignant origin were selected and **(e)** pathways and GO categories to profile the gene selection was obtained from Gene Ontology (GO) and Reactome database. **(f)** Visualization using Cytoscape and ClueGO plugin for biological process and **(g)** differentiation of the functions between conditions of the subpopulations of T cells was displayed.

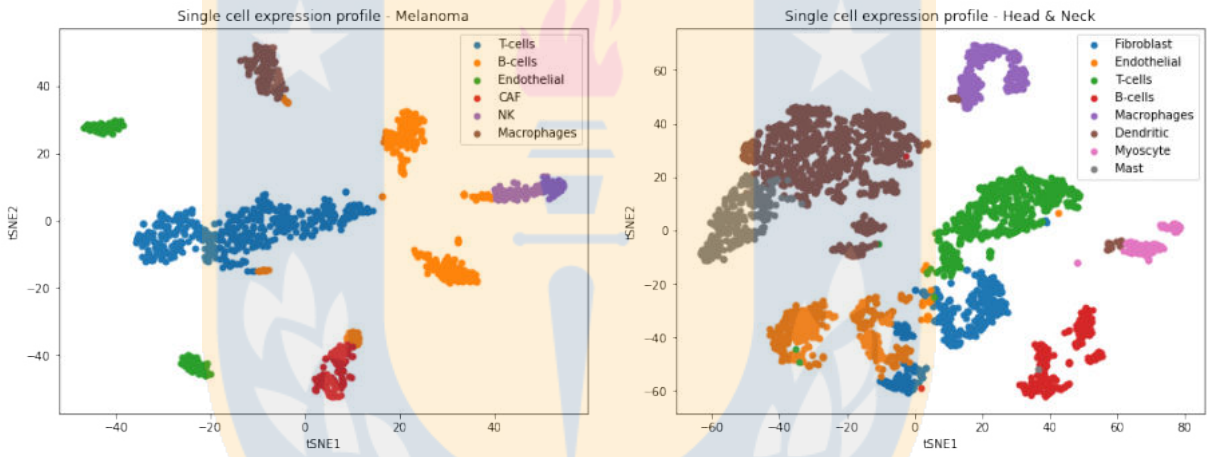
genes from malignant and non-malignant samples for each type of subpopulation were mapped to biological pathways using gene ontology (GO) terms [8, 36], classified as biological process, molecular functions, cellular components and reactome pathways [89] (Figure 4.1e). The data was visualized using different networks (Figure 4.1f) and exclusive pathways for each subpopulation in both conditions were finally identified (Figure 4.1g).

4.3.1 Profiling of tissue-infiltrating T cells from different types of cancer

In order to obtain the transcriptomic profile of tissue infiltrating T cells, we analyzed each dataset individually to get the gene count for each cell type per condition (malignant, non-malignant). We defined as a cutoff the upper median TPM values of gene expression in each scRNA-seq dataset and removed genes with low expression values in all the type of cells identified.

For datasets GSE126030, GSE99254 and GSE108989, the T cells were already labeled

in separate files in the Gene Expression Omnibus repository, therefore these files used for further analysis. Datasets GSE72056 and GSE103322 contain different types of cells embedded in a single matrix file, thus dimensionality reduction and clustering were required to identify T cells. Briefly, we used PCA and variance analysis in order to obtain the value of the optimal number of components using the Scikit-learn implementation of PCA [162]. We identified eight components in melanoma (GSE72056) (Figure 4.2a) and nine components in head and neck cancer (GSE103322) (Figure 4.2b). In order to identify the T cells population from these datasets, we performed a t-SNE dimensionality reduction and a clustering approach. Figure 4.2a and 4.2bb show t-SNE plots identifying the different types of cells, including B cells, macrophages, endothelial, cancer-associated fibroblasts (CAFs), NK cells, mast cells and myocytes. Based on this analysis, only T cells were selected by using the gene markers detailed on Appendix Table S1.



(a) Melanoma cancer (b) Head and neck cancer

Figure 4.2: Cell identification in melanoma and head and neck cancer
Single cells plot of gene expression profiles using dimensionality reduction technique t-SNE for malignant cells for Melanoma and Head and Neck cancer data colored by type of cells.

For the identification of T cell subsets in all datasets, we performed a comparison using a set of classical gene markers including *cd3g*, *cd8*, *cd4* and *foxp3*. Thus, these gene markers helped to filter and confirm the identification of CD4 helper T cells (*cd3g*>0; *cd8*=0; *cd4*>0; *foxp3*=0), CD8 cytotoxic T cells (*cd3g*>0; *cd8*>0; *cd4*=0; *foxp3*>0) and Tregs (*cd3g*>0; *cd8*=0; *cd4*>0, *foxp3*>0) from the original datasets. In Table 4.1

we reported a summary of counting of genes by dataset after removing genes with a low expression values, identifying the type of tissue with its respective Gene Expression Omnibus ID (GEO), and the counting of malignant and non-malignant related genes for CD4-T, CD8-T and Treg cells. We identified a total of 67,917 genes in malignant CD4, 63,038 genes in malignant CD8 and 56,827 genes in malignant Treg from five tissues (breast, lung, colorectal, head and neck and melanoma). On the other hand, we identified 24,436 genes in non-malignant CD4, 28,341 genes in non-malignant CD8 and 20,877 genes in non-malignant Treg for two tissues (lung and colorectal).

Table 4.1: Summary of the datasets used in this work. Each row indicates the number of genes for CD4-T, CD8-T and Treg.

Data ID	**Type of cancer**	**Condition**	**CD4**	**CD8**	**Treg**
GSE114727	Breast	Malignant	12855	12601	8924
GSE75688	Breast	Malignant	8878	2620	6835
GSE126030	Lung	Non-malignant	8418	10573	6620
GSE99254	Lung	Malignant	10571	11034	10349
		Non-malignant	9748	10488	7532
GSE108989	Colorectal	Malignant	7324	7005	7283
		Non-malignant	6270	7280	6725
GSE103322	Head and neck	Malignant	7574	8093	7517
GSE72056	Melanoma	Malignant	10136	10779	8732
GSE123139	Melanoma	Malignant	10579	10906	7187

Once identified the different T cell subsets, we performed a second analysis to determine those common genes for each condition (malignant; non-malignant) across all different type of samples (Figure 4.3). Then, we compared the common genes between the malignant and non-malignant condition for each T cell subset. The Venn diagram in Figure 4.4a, Figure 4.4b and Figure 4.4c represents the total number of genes from scRNA-seq data for the CD4-T, CD8-T and Tregs subsets, respectively. We observed 652 (11.74%) of exclusive genes in malignant-derived CD4-T, 69 (1.10%) of exclusive genes in malignant-derived CD8-T cells and 557 (12%) of exclusive genes in malignant-derived Tregs analyzed. This overall gene identification revealed exclusive and common genes per T cell subset and condition, allowing to use this information for further pathway analysis.

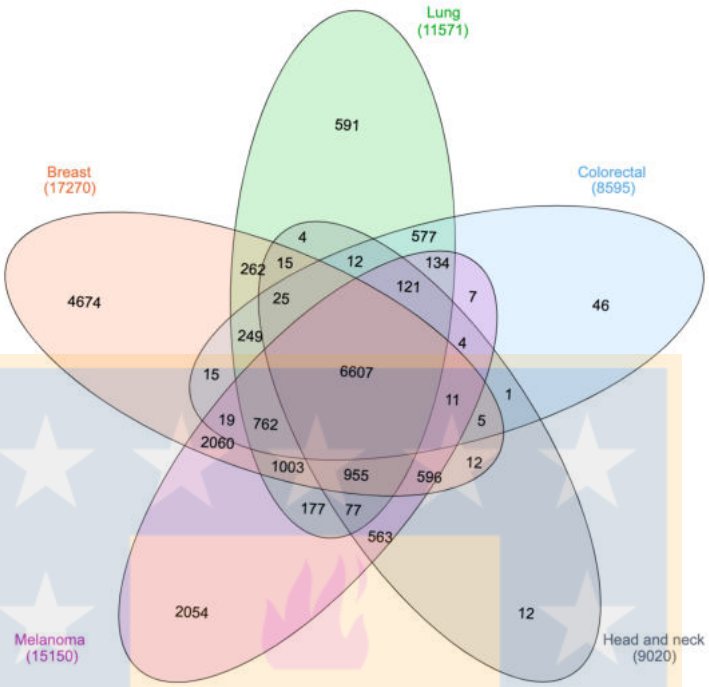
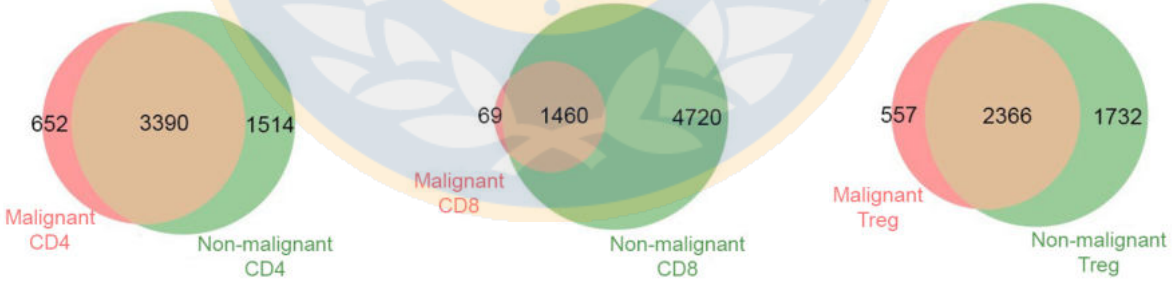


Figure 4.3: Venn diagram of number of genes of T cells across different types of cancer



(a) CD4-T cells (b) CD8-T cells (c) Treg cells

Figure 4.4: Venn diagram of T cell subpopulations representing the exclusive and common genes between malignant and non-malignant origin.

4.3.2 GO annotations and biological pathways in T cells from malignant and non-malignant cancer

GO annotations and biological pathways were analyzed by comparing the data obtained from malignant and non-malignant samples per subset. The Gene Ontology Consortium [8, 36] was used to establish the GO enrichment and the reactome pathway database [89] to analyze pathways and generalize the concept of the reactions that match our datasets, including biological process with a focus in signaling, metabolism, transcriptional regulation, apoptosis and disease. We identified 7,490 GO annotations for biological process, 1,404 GO annotations for molecular functions, 2,035 for GO annotations of cellular component ontology and 12,033 reactome pathways (Table 4.2). These annotations were divided into malignant and non-malignant origin in CD4, CD8 and Tregs. For GO terms 38, 72 and 100 biological functions were associated exclusively to malignant CD4-T cells, CD8-T cells and Tregs, respectively. In general, from malignant samples we observed that some of the main associations found in the GO terms between CD4-T, CD8-T and Treg corresponded to immune response and defense, some of them being transcendental for cancer immunotherapies [160, 158].

Table 4.2: Summary of functional enrichment annotations for malignant and non-malignant CD4-T, CD8-T and Treg.

Functional enrichment	**Malignant**			**Non-malignant**		
	CD4	**CD8**	**Treg**	**CD4**	**CD8**	**Treg**
Biological Process	1335	950	1388	1282	1265	1270
Molecular Function	270	159	249	240	249	237
Cellular Component	376	271	385	322	339	342
Reactome Pathway	2059	1746	1966	2073	2138	2051

In order to determine interaction between biological pathways associated to malignant or non-malignant samples and visualize them as an interaction network, the biological process terms were analyzed using the Cytoscape software [188] and the ClueGO plugin [17] to visualize the non-redundant biological terms for large clusters of genes as a hierarchical biological network. Using this method, we observed 17 exclusive GO terms in malignant CD4, 17 exclusive GO terms in non-malignant CD4, 12 exclusive GO terms in malignant CD8, 86 exclusive GO terms in non-malignant CD8, 54 exclusive GO terms in malignant

Treg and 25 exclusive GO terms in non-malignant Treg. Figure 4.5 display a visualization of the network for malignant and non-malignant genes for CD4-T, CD8-T and Treg, including clustered pathways for each condition (malignant and non-malignant) and labelling according to the most significant term per group. For the specificity of each pathway, we observed differences between malignant and non-malignant samples. This data is available in the child nodes that together with the p-values information from ClueGO indicated different functional categories inside the networks. Of note, common functions between malignant and non-malignant T-cell subsets were observed in this analysis despite previous gene selection. In summary, our data revealed 93 clusters for malignant CD4, 92 clusters for non-malignant CD4, 43 clusters for malignant CD8, 123 clusters for non-malignant CD8, 83 clusters for malignant Treg and 81 clusters for non-malignant Treg samples (Figure 4.5). One cluster that was differentially observed in Treg cell was regulation of macromolecule metabolic process, which was positive in malignant samples, but negative in non-malignant Treg. Besides clusters, this analysis shows different specialization between the pathways for each condition.

4.3.3 Exclusive biological pathways in T cells from different types of cancer

After defining the pathways in cells from malignant and non-malignant samples we identified exclusive pathways for T-lymphocytes between both conditions (Figure 4.6). The 31 recurrent annotations found in the three T cell subsets from malignant samples were associated with different types of cancer and other diseases of the immune system, being the malignant genes associated with the types of cancer studied and viral diseases such as HIV Infection. However, we also found that pathways associated to localization were common between the three subpopulations from malignant samples, such as positive regulation of protein localization to Cajal body and SRP-dependent cotranslational protein targeting to membrane. Another important function found were viral transcription and viral gene expression pathways that plays an important role in viral transcription and translation [229, 49]. From non-malignant samples we observed 23 recurrent annotations within the T cells highlighting the regulation of translation in response to stress, nucleotide-excision repair, DNA damage recognition and viral budding were the most important and common terms between T cell subsets.

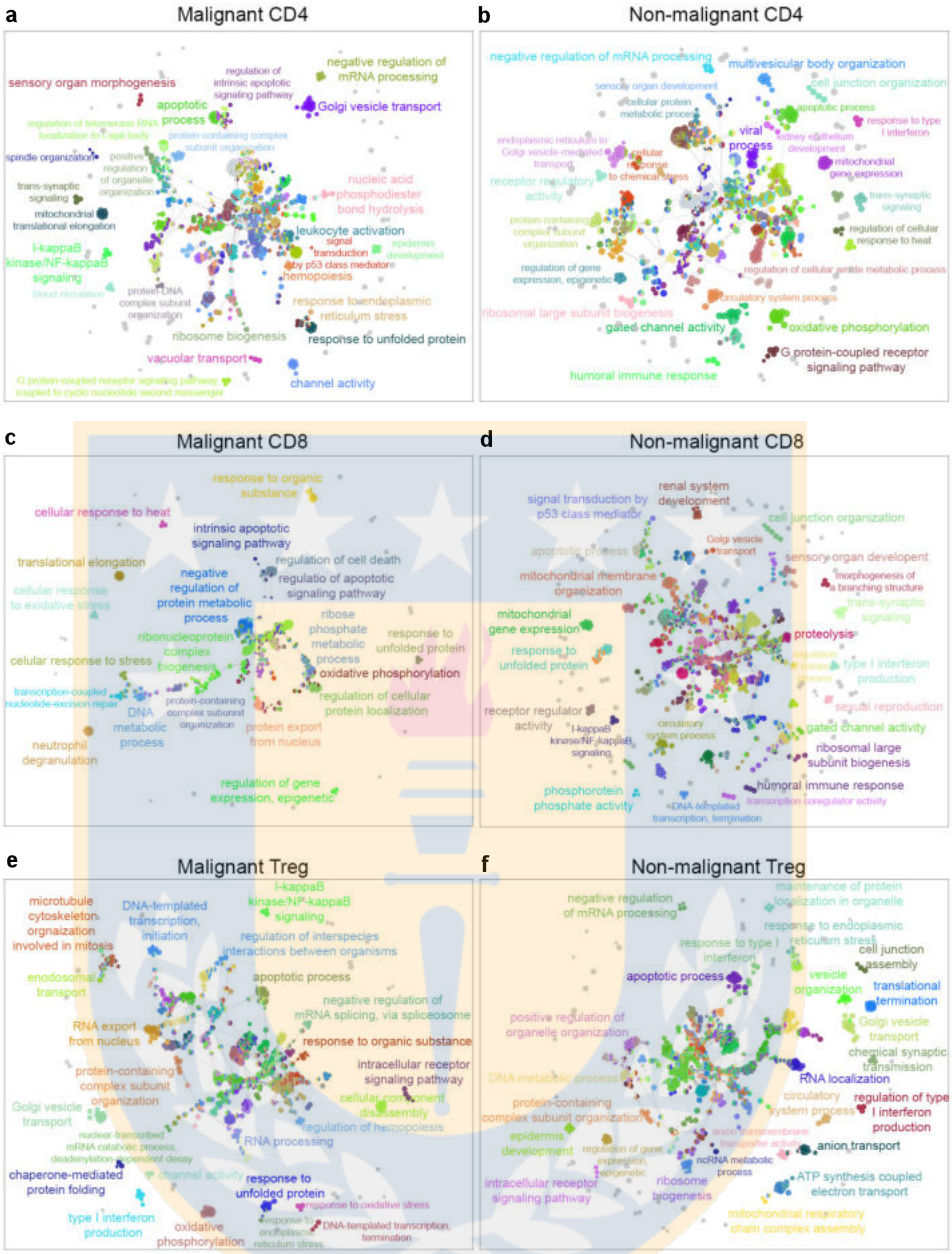


Figure 4.5: Gene ontology of biological processes networks.

The visualization of the networks was performed in Cytoscape using the ClueGO plugin. **(a)** Network visualization of GO terms associated for CD4-T cells with malignant origin and **(b)** for GO terms associated to CD4-T cells from non-malignant origin. **(c)** Network corresponding to GO terms of CD8-T cells from malignant origin and **(d)** represent the GO terms associated to CD8-T cells with non-malignant origin. **(e)** GO terms associated to Treg from malignant origin and **(f)** GO terms associated to Treg from non-malignant origin.

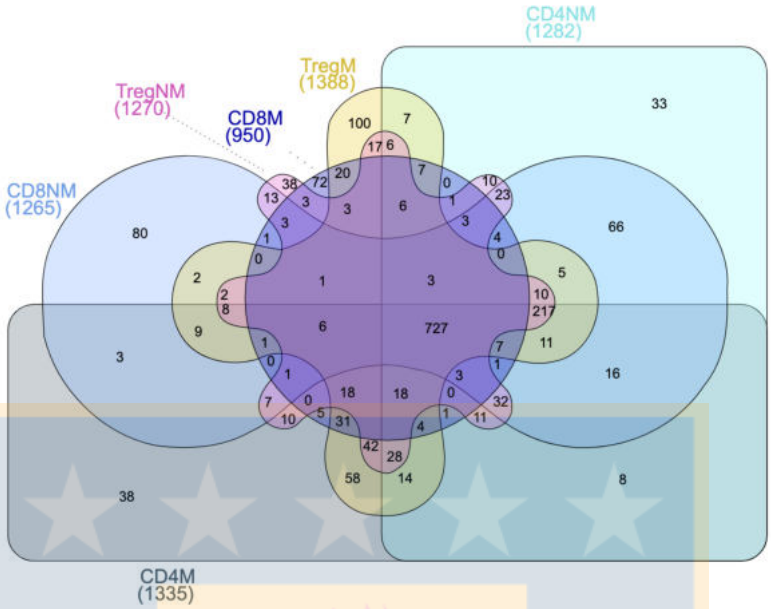


Figure 4.6: Comparison of common biological processes across all the T cell subpopulation

When molecular function annotations were analyzed, we observed that peroxiredoxin activity, threonine-type peptidase activity and NADH dehydrogenase activity were recurrent and relevant annotations for CD4, CD8 and Treg cells from malignant origin. For T cells from non-malignant origin, we found that between the three subpopulation structural constituent of ribosome, thyroid hormone receptor binding and snoRNA binding were common annotations between them. In the case of cellular components annotations, we observed that signal recognition particle, endoplasmic reticulum targeting and proteasome core complex, beta-subunit complex were the three most important annotations in common between malignant CD4, CD8 and Treg cells. For non-malignant cells, we observed that methylosome, U2-type catalytic step 2 spliceosome and eukaryotic translation initiation factor 3 complex (eIF3m) were the most important annotations in common between CD4-T, CD8-T and Tregs.

Exclusive functions of the different T cell subsets between malignant and non-malignant samples were finally analyzed. 38 exclusive biological process terms belong to malignant

CD4 and 33 to non-malignant CD4, 72 exclusive biological process terms belong to malignant CD8 and 80 to non-malignant CD8 and 100 exclusive biological terms belong to malignant Treg and 38 to non-malignant Treg (Figure 4.6).

The main pathways associated exclusively to malignant CD4 were somatic diversification of immune receptors via germline recombination within a single locus, protein targeting to vacuole, lymphocyte activation involved in immune response, positive regulation of DNA-binding transcription factor activity, T cell differentiation involved in the immune response, negative regulation of extrinsic apoptotic signaling pathway via death domain receptor and positive regulation of leukocyte activation among others. On the other hand, the main exclusive pathways for non-malignant CD4 were negative regulation of response to wounding, aminoglycan biosynthetic process, chemical synaptic transmission, cell communication involved in cardiac conduction, neuromuscular synaptic transmission, interleukin-15-mediated signaling pathway, negative regulation of lymphocyte apoptotic process, among others.

Malignant CD8 was characterized mainly by metabolic process such as purine nucleoside triphosphate biosynthetic process, ATP biosynthetic process, response to epidermal growth factor, gluconeogenesis, response to gamma radiation, negative regulation of oxidative stress and positive regulation of signal transduction by p53 class mediator. Non-malignant CD8 pathways were characterized by cell morphogenesis involved in differentiation, mitochondrial RNA metabolic process, DNA modification, phospholipid metabolic process, multicellular organism development, cell recognition, and others.

In the case of malignant Tregs, we observed a detailed specificity in the pathways such as positive regulation of cell differentiation, circadian regulation of gene expression, cellular response to glucocorticoid stimulus, response to drug, negative regulation of mRNA catabolic process, negative regulation of cell population proliferation, positive regulation of cell cycle G1/S phase transition, negative regulation of NIK/NK-kappaB signaling and tricarboxylic acid metabolic process, meanwhile for non-malignant Tregs main pathways were negative regulation of cytokine production, negative regulation of immune effector process, positive regulation of pathway-restricted SMAD protein phosphorylation, positive regulation in response to stress, positive regulation of T cell proliferation and negative regulation of receptor signaling pathway via JAK-STAT, among others.

Across the T cells subpopulations we also observed 59 exclusive positive and negative regulations of different biological process. In Figure 4.7 we highlighted negative biological functions with a left orientation and positive biological functions with a positive orientation. Analyzing the different mechanism of regulation from the cell help us to understand how the regulation is working simultaneously in several pathways either positively or negatively to regulate exclusive process of each T cell subpopulations.

4.3.4 Reactome pathways in T cells from different types of cancer

Reactome pathways were also analyzed between T cell subsets through Reactome Pathway Database [89]. We observed a total of 5,771 annotations for malignant data and 6,262 annotations for non-malignant data. In the three T cell subsets derived from malignant samples the main pathways found were associated to cellular response to stress, translation and ER-Phagosome pathway. The non-malignant derived T cell subsets were also associated to RNA process and class I MHC mediated antigen processing & presentation. In malignant CD4 the most overrepresented pathways are SRP-dependent cotranslational protein targeting to membrane, GTP hydrolysis and joining of the 60S ribosomal subunit and L13a-mediated translational silencing of Ceruloplasmin expression that are part of the metabolism of proteins in the database. In the case of non-malignant CD4 the main three pathways are translation, metabolism of RNA and processing of Capped Intron-Containing Pre-mRNA, thus regulating more processes of metabolism of RNA. For malignant CD8 the most overrepresented pathways are peptide chain elongation, formation of a pool of free 40S subunits and eukaryotic translation elongation that are also part of metabolism of proteins. In non-malignant CD8 same pathways as in non-malignant CD4 are the most overrepresented. In malignant Treg formation of a pool of free 40S subunits, nonsense mediated decay (NMD) independent of the exon junction complex (EJC) and peptide chain elongation are the most important pathways also related to metabolism of protein and RNA. In non-malignant Treg we identified translation, metabolism of RNA and regulation of expression of SLITs and ROBOs, observing in this subpopulation more pathways from the developmental biology location.

Furthermore, for a more exhaustive analysis and subsequent discussion we focus on the

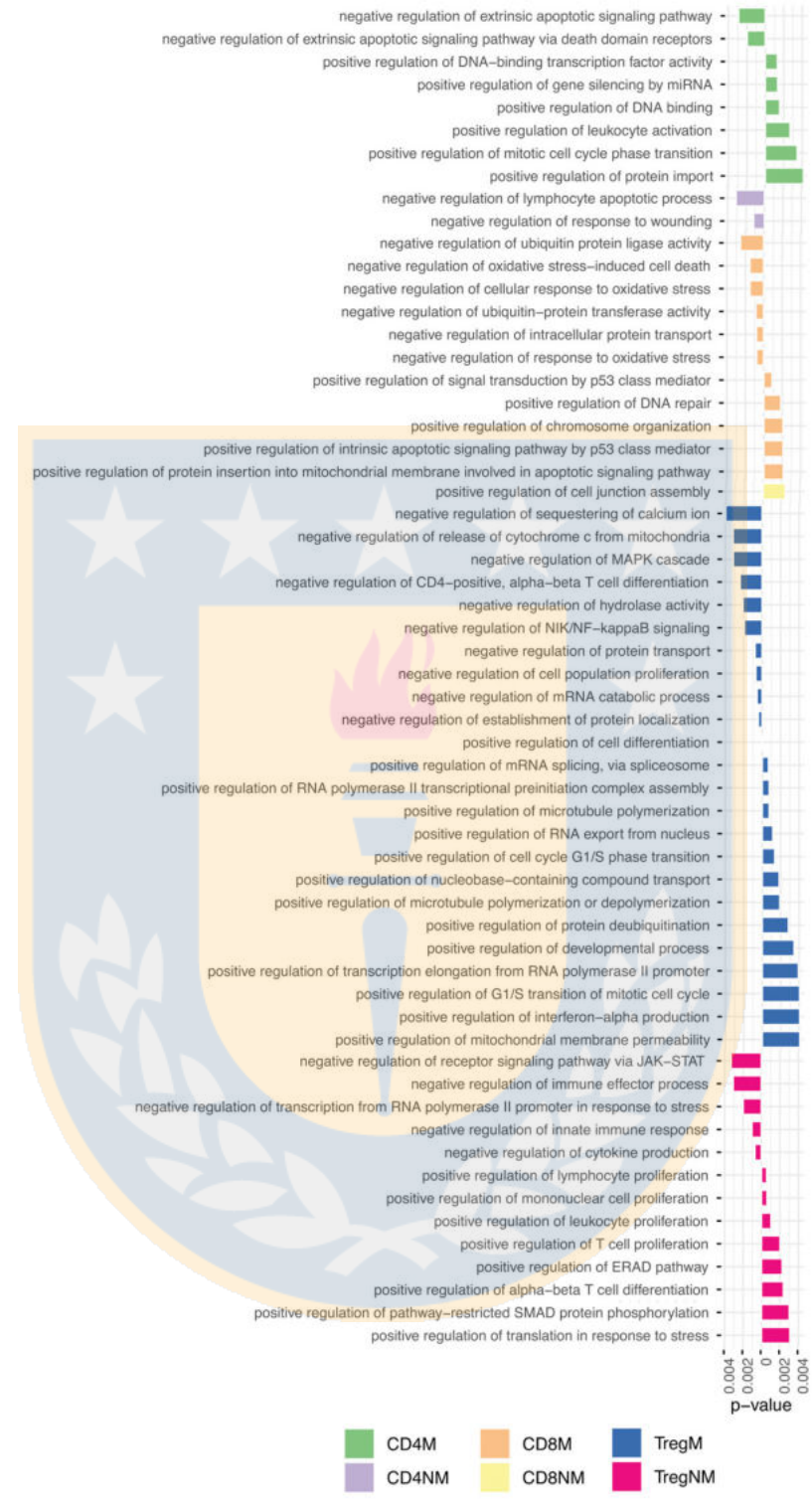


Figure 4.7: Exclusive GO terms for biological processes associated with positive and negative regulations to each T cell subpopulation.

annotations under the immune system pathway, observing that TRAF6 mediated IRF7 activation in TLR7/8 or TLR9 signaling is present only in malignant CD4. In non-malignant CD4 we also observed unique pathways that were TLR3-mediated TICAM1-dependent programmed cell death, TICAM1-dependent activation of IRF3/IRF7, TICAM1, TRAF6-dependent induction of TAK1 complex and TICAM1, RIP-mediated IKK complex recruitment.

In summary, our data revealed that despite the type of cancer, there are pathways in T cells that are common between cancer, but unique in comparison with pathways in T cells from non-malignant tissues.

4.4 Discussion

Over the past years, datasets obtained from scRNA-seq have revealed valuable information about the repertoire of cells contributing or controlling the development of several types of tumors [206]. In this study, we analyzed published scRNA-seq datasets obtained from T cell subsets from malignant and non-malignant origin, in order to identify common pathways between these cells across different types of tumors. Our data revealed that regardless the type of cancer, we observe common functions associated with metabolic process, translation and immune-related pathways for each T cell subset such. Publicly available data was analyzed through an exhaustive in silico analysis where samples were characterized and re-labeled in order to generate a list of pathways and gene ontology (GO) categories, which were further validated by using data from different sources. Finding subgroups of cells and analyzing in tumor tissue versus normal tissue would lead us understand common pathways of T cells in different cancer. In addition, regardless the tumor origin, similar pathways were identified; therefore, potential immunomodulatory therapies associated with these pathways could be applied to different types of tumors.

One of the major limitations in biological analysis is often the high dimensionality of the data [80]. In our study, it was essential to define a strategy to thoroughly label the samples, and thus ensure a more precise and reliable result by using machine learning techniques. Using the classification of cells in terms of their immunological background, we show that a high similarity exists for relevant genes. These genes were mapped to the same biological functions, mainly cancer development functions, and finally, GO terms and

reactome annotations gave us a clear idea of the pathways highlighted to define possible targets to identify key cellular pathways from the immune system in cancer.

A visualization of the overrepresenting GO annotations in T cell subpopulations isolated from both malignant and non-malignant tissues, allows to identify common genes between healthy and tumor-infiltrating T cells, and also common genes across different types of cancer for each T cell subset. We observed the highest percentage of common GO annotations between malignant and non-malignant conditions in CD4-T cells. The most overrepresented terms found only in malignant CD4-T was detection of abiotic stimulus and detection of external stimulus that participates in the perception of the stimulus. Then, it is received by a cell and converted into a molecular signal [36, 102, 113]. Positive regulation of cysteine-type endopeptidase activity was also an overrepresented term in malignant CD4-T. This function is involved in apoptotic processes and inflammasome, being also responsible for the activation of inflammatory response [23]. The data also highlighted positive regulation of antigen receptor-mediated signaling pathway as an important term in malignant helper cells. In fact, it has been associated with key immunological function in ovarian cancer between the four stages of this type of cancer because it is one of the initial triggers of the immune response and can activation of the T cell response [30].

We also observed some overrepresented function in non-malignant CD4-T that indirectly could be important for cancer as the absence of these may also contribute to tumor development. Here, regulation of oxidative stress-induced intrinsic apoptotic signaling pathway, autophagosome maturation and regulation of vascular permeability process were overrepresented only in non-malignant CD4-T. The first function plays an essential regulatory role in promoting cell survival under stress conditions contributing to cancer therapy [126]. For autophagosome maturation, this pathway is crucial in the delivery of cytoplasmic components. Therefore, the role in cancer for damaged proteins and organelles autophagy allows prolonged survival to tumor cells, providing a protective function limiting tumor necrosis and inflammation [174, 135]. In the case of vascular permeability, this pathway is related with blood distribution to all tissues and maintain the homeostasis, lipid transport and immune surveillance. In the particular case of cancer, this permeability can facilitate metastatic spread [33]. Also, vascular permeability is crucial in physiological and pathological angiogenesis, due to normal or healthy blood vessel growth occurring

during tissue repair and it has been reported as a cause of mortality in cancer, among other causes [12].

Overrepresented annotations in CD8-T cells from malignant samples were characterized by a development of biosynthetic processes, showing pathways such as purine nucleoside triphosphate biosynthetic process, ribonucleoside triphosphate biosynthetic process, purine ribonucleoside triphosphate biosynthetic process and ATP biosynthetic process. All these functions in cancer are associated with metabolic requirements for cell growth and proliferation of cancer cells by producing de novo nucleotide synthesis, maintaining normal triphosphate biosynthetic process, as this process is critical for replication and repair the DNA [212, 22]. Another pathway that characterized unique CD8-T responses in malignant conditions was response to epidermal growth factor, which has been associated with the regulation of cell proliferation, differentiation and migration through epidermal growth factor receptor (EGFR) function, that play an important role in tumorigenesis in various types of epithelial cancers. Nowadays, novel therapies that target the EGFR agents have improved patient's therapies with colorectal, lung, head and neck and pancreatic cancers, however, there are some cases using monoclonal antibodies where an activation of signaling pathways downstream of the EGFR could produce resistance to the treatment [180, 108]. Gluconeogenesis was also observed only in malignant CD8-T cells, but the role of this metabolic process in CD8-T cells in cancer is unclear. It is known that it generates free glucose from precursors and is associated to cancer cell plasticity and tumor cell growth. However, it has also been shown that this pathway is inhibited in some types of cancers as it may engage in truncated gluconeogenesis function in fasting conditions [186, 217, 62]. On the other hand, overrepresented annotations in non-malignant CD8-T were characterized by membrane invagination, multicellular organism development and inner ear morphogenesis and anatomical structure development that are associated to developmental processes. Those terms in general are part of membrane organization and developmental processes. Others highlighted pathways observed were glycerolipid biosynthetic process, glycerophospholipid biosynthetic process and lipid biosynthetic process. Glycerolipid processes have been proposed within a new therapy in cancer, neuroscience and metabolic diseases by targeting with small molecule inhibitors [187, 127].

In Tregs, the pathways with the highest p-value from malignant samples were positive regulation of cell differentiation and cellular response to different compound, such as nitrogen, oxygen-containing, glucocorticoids stimulus, organonitrogen among others. Those compounds in cancer studies have been proved that when are altered, they support cancer and immune cells responses [106]. In the case of glucocorticoids, those corticosteroids act primarily on carbohydrate and protein metabolism having anti-inflammatory and immunosuppressive effects [38]. Other pathways associated only to malignant Tregs were circadian regulation of gene expression and circadian rhythm, both processes modulate the frequency of gene expression pattern with a regularity of approximately 24 hours. In cancer studies, those pathways participate over cyclic physiological processes. In addition, cancer has been linked with the disruption of circadian rhythms [134, 94]. Regulation of NIK/NF-kappaB signaling, negative regulation of I-kappaB kinase/NF-kappaB signaling and negative regulation of NIK/NF-kappaB signaling were also associated only to malignant Tregs, affecting a complex network between extracellular stimuli to cell survival, developing an essential role in inflammation, innate immunity and cancer initiation and progression [79, 119, 226]. Non-malignant Treg present annotations such as retrograde vesicle-mediated transport, Golgi to endoplasmic reticulum, maintenance of protein location and positive regulation of transport that belong to localization and transport pathways. Also, we observed pathways associated to cytokines such as positive regulation of response to cytokine stimulus and positive regulation of lymphocyte proliferation that play an important role in modulation of immune and inflammatory responses, due to cytokines key role at clinical cancer research [227, 35, 112, 44].

Finally, terms obtained from the Reactome analysis revealed pathways that play an important role in the immune system as cytokines that regulate and mediate immunity, inflammation and haematopoiesis, promoting intercellular communication between immune cells [153]. In addition, these proteins bind to their cell surface receptor and act in an autocrine and/or paracrine fashion, inducing tissue growth and repair [3, 91]. Moreover, adaptive immunity is involved in roles such as the recognition of particular pathogens or antigens prior presentation by antigen presenting cell from peripheral tissues [159]. TNFR2 non-canonical NF-kB pathway was one of the most significant pathways between

all the datasets analyzed, even though it was found in both conditions (malignant and non-malignant). It has been shown that TNFR2 in normal tissues exhibits basal expression [64], whereas it can be induced to promote cell survival pathways such as cell proliferation by activating transcription factor NF-kB via the alternative non-canonical route [190]. This suggest that the regulation of TNFR2 may be relevant in tumor infiltrating lymphocytes.

4.5 Conclusions

In summary, we used dimensionality reduction and pathways analysis to integrate a large amount of data in order to identify common genetic T cell signatures across different type of cancers. These methodologies allow us to compare those T cell signatures and their core dynamics pathways between malignant and non-malignant samples to identify unique and common pathways in CD4-T, CD8-T and Tregs. Our analysis revealed that pathways related with the immune response, metabolism and viral immunoregulation were observed exclusively in cancer samples. Several other pathways were identified in all three T cell subsets, however future research is required to understand whether these pathways favor effective anti-tumor responses, or they are impaired and therefore do not prevent tumor progression.

So far, we have analyzed only one type of cell, however, we want to explore more cells from the immune system, know how many of them are common in other types of cancer, and how these play a key role in the tumor development, among others biological functions. All these questions are investigated in the next chapter.

Chapter 5

Classification of cancer cells using machine learning and deep learning models

The aim of this chapter was to analyze gene expression profiles of scRNA-seq samples from nine different types of cancer and to develop machine learning and deep learning (DL) models to classify the different types of cells and characterize the tumor heterogeneity in malignant samples. We found that 10,788 genes are common between the datasets. We explored key genes highly weighted modeling the mean-variance relationship inherent in scRNA-seq data and performing a downstream analysis such as PCA, tSNE and Uniform Manifold Approximation and Projection (UMAP). Then, we performed two DL models to classify the cells according to the gene markers observed in the literature. However, given the nature of the scRNA-seq experiment, DL models performed well when we analyzed a single experiment and not a combination of experiments. Under this view, we selected the results from the principal component analysis (PCA) to discover relevant pathways. This result is closely related to the heterogeneity of cells between tumors, characterizing the cellular diversity composition in our data.

5.1 Related work

Regarding genomic data, during the last two decades it has mainly come from DNA or RNA sequencing from next-generation sequencing technology (NGS) [29]. NGS or high-throughput sequencing technologies provide an accurate genome-wide that contains large volumes of sequence data [184, 141] useful for different applications such as mutation mapping and polymorphism discovery [132], biomedical research [189], among others.

The quantification of the level in which a particular gene is expressed within a cell is called "Gene expression" [185]. Most papers that use genomic data for cancer detection and prediction employ gene expression values. This kind of data is very complex due to its high dimensionality and intricacy, making it challenging to use for cancer detection.

The growth of data generated through the NGS has allowed for an advance in the analysis of genetic variants for the diagnosis and cancer treatment [172]. In this sense, the analysis of gene expression data and mutations have been the main strategies in the detection and classification of features and cells [39, 223]. The applications of DL in this field are an efficient method to extract features and interpret this type of data linking genetic variants with diseases [26, 7]. For instance, the use of neural networks in large datasets with a high dimensionality and sparse and noisy data with nonlinear relationships, has been a widely used analysis alternative, as well as the generation of models that can be applied to new datasets of gene expression [131].

The most used architecture for genomic data is multilayer perceptron (MLP) (53.62%) and autoencoder (AE) (30.43%). AEs have mainly been used for dimensionality reduction [205], classification of cancer cells and clustering of gene expression to identify relevant genes for cancer diagnosis and treatment [39, 50]. For genomics, denoising autoencoders (DAE) can be used to extract useful features that will constitute better higher level representations of the data, in the sense of reconstructing a clean repaired output from a corrupted version of it [214]. Danaee et al. [39] state that a DAE model improves classification performance for cancer detection, as a single AE does not allow to extract all the useful representations from noisy and high dimensional data.

For breast cancer, we observed that DAEs and Stacked Denoising Autoencoder (SDAEs) are useful to extract functional features from high dimensional gene expression profiles [39]. Tan et al., [201] implemented a method to identify and extract complex patterns

from microarray samples using a DAE with three layers: an input layer, a hidden layer, and a reconstructed layer. Results show that the DAE successfully constructs features that contain both clinical and molecular information. Features constructed by the AE generalize to an independent dataset collected using a distinct experimental platform. Another model for breast cancer developed by Liu et al., [120] extracts deep features from RNA-seq data and copy number alteration (CNA) data separately and jointly. In this work, two unsupervised DAE were developed to extract deep features and a heat map was used to visualize and cluster patients into subgroups based on these features. All of these models linked together improved the significance of the features associated with breast cancer patients clinical characteristics and outcomes.

Way et al., [218] presented a variational autoencoder (VAE) model applied to RNA-seq data from The Cancer Genome Atlas (TCGA). They called the model “Tybalt", and it was trained with 5,000 input genes encoded to 100 features and reconstructed back to the original 5,000. Tybalt determined that the learned features were generally non-redundant and could disentangle large sources of variation in the data and can be useful in cancer stratification or in the prediction of specific activated expression patterns. Nevertheless, the authors declare it still requires careful validation and evaluation. For breast cancer, Titus et al., [207] developed a framework that uses a VAE architecture to learn latent representations of the DNA methylation landscape from three independent breast tumor datasets. The model works as an extension of the Tybalt VAE model and extracts sets of specific features that contribute to the learned latent dimensions representing Estrogen Receptor (ER)-negative and ER-positive tumors.

GANs have been applied for cancer diagnosis to learn features from unlabeled microarray data for breast and prostate cancer. In the model proposed by Bhat et al., [16] the generative network probabilistically generates output samples, using random noise as input, whereas the inference or discriminator network learns to discriminate the true data distribution samples from the generated fake data. The model learns features that are passed through sigmoid activation functions and used as input to conventional non-DL machine learning models that classify them as cancerous or non-cancerous.

5.2 Our Approach

5.2.1 Data collection and pre-processing

Data collection from previous scRNA-seq expression profiles from malignant and non-malignant cells were included in the analysis. We selected scRNA-seq data from isolated cells from breast (GSE114727 and GSE75688), lung (GSE126030 and GSE99254), colorectal (GSE108989), melanoma (GSE72056 and GSE123139), and head and neck cancer (GSE103322). The datasets were obtained from the Gene Expression Omnibus (GEO) repository and all of them were sequenced on Illumina HiSeq2500/HiSeq4000 or Illumina NextSeq 500 (Homo sapiens) with a similar experimental design. We verified the quality of each sequencing library with FastQC [5], a software package that estimates the number of un-callable and low quality bases. Mapping to the human reference genome (hg38) was done using STAR [41], a high performance community-standard aligner.

Each dataset was analyzed separately as a digital expression matrix. We used transcripts per million (TPM) values as gene expression levels for all the analysis, calculated as:

$$\frac{10^6 \cdot C_{ij}/\text{length of gene i}}{\sum_i C_{ij}/\text{length of gene i}},$$

where C_{ij} is the count value of gene i in cell j. We removed genes with low expression values, considering as cutoff the upper median TPM values [83]. Then, we compared the ten datasets matrices and we searched the common genes across.

5.2.2 Data exploration

Using Seurat R package (version 4.0.4) [181] cells with < 200 genes detected were filtered from downstream analyses. All samples were merged with the *CreateSeuratObject* function into one Seurat object. The merged Seurat object was normalized and scaled with a global-scaling normalization method *LogNormalize* that normalizes the feature expression measurements for each cell by the total expression, multiplies this by scale factor (10,000 by default), and log-transforms the result.

We next calculate a subset of features that exhibit high cell-to-cell variation in the dataset, i.e. highly expressed in some cells, and lowly expressed in others. It is calculated by modeling the mean-variance relationship inherent in single-cell data using the

FindVariableFeatures() function from Seurat.

5.2.3 Cell type annotation

In order to determine cell types, we combined unsupervised clustering and differential expression to compare top differentially expressed genes with cell type specific expression known from the literature. Through this approach, we identified broad categories among all cells, and further delineated cellular subtypes by isolating subsets. Differential expression was performed using *FindAllMarkers* function from Seurat with default parameters (only.pos = TRUE, min.pct = 0.25, logfc.threshold = 0.25) in order to find marker genes that define clusters.

Another approach was to search in the literature for the gene markers for each type of cell. Table 5.1 shows the most used genes as marker.

5.2.4 Dimensionality reduction and clustering

For a first approach, we performed PCA using *RunPCA* from Seurat on the scaled data in order to visualize a list of the genes which were the most highly and lowly weighted in the different PCs. To determine the dimensionality to which to reduce the dataset, we plot the explained variance for each principal components. We also run two non-linear dimensional reduction techniques: tSNE and Uniform Manifold Approximation and Projection (UMAP), to visualize the clusters of the dataset according to Seurat analysis.

On the other hand, we implemented two deep learning models using PyTorch [161], an open source machine learning framework. For the two models we split the data in train and test sets into 80% and 20% respectively. The selection of hyperparameters was using *GridSearchCV* method from Scikit-Learn and manual tuning. The implemented models were:

Autoencoder (AE): (Model description in section 3.2.5) an AE is composed of an encoder and a decoder part. First, the encoder takes 128 features and it produces the latent code representation which then goes to the decoder for reconstruction. Next, the decoder, which again keeps increasing the features size until we get the original 128 features. The forward method simply combines the encoder and decoder

Table 5.1: Gene marker found in the literature review.

Type of cell	**Gene markers**	**References**
T cells	*cd3d, cd3e, cd3g, cd8, cd4, foxp3, cd2*	[206, 168, 90, 9, 164, 152]
B cells	*cd79b, blk, hla-dpa1, hla-dra, cd37, cd74, cd19, cd79a*	[206, 168, 43, 90, 9, 164, 152]
Mast cells	*enpp3, kit*	[168, 43, 9, 152]
Macrophages	*cd163, cd14, csf1r, fcer1g, fcgr3a, tyrobp, cd68, c1q*	[206, 168, 154, 90, 9, 164, 152]
Monocytes	*vcan*	[9, 152]
Dendritic	*itgae, itgax, cd8a, clec9a, cd141*	[168, 9, 152]
Neutrophils	*cd33, cd44, ceacam8, cd11b, cd14, cd15, cd16, cd32*	[9]
Endothelial	*pecam1, vwf, cdh5*	[206, 168, 144, 43, 90, 164]
Fibroblast	*col5a1, fbln2, col1a1, col1a2, lum,fbln1, pdgfra*	[168, 144, 43, 90, 164]
Myocytes	*tnnt2, myl2*	[168]
Cancer-associated fibroblast (CAF) cells	*col6a1, col6a2, col6a3, fap, thy1, dcn*	[206]
Myeloid	*cd45, cd19, cds6, cd11b*	[43, 90, 9]
Myofibroblast	*cd34*	[43]
Epithelial	*acta2, mylk, tp63, sytl2, abca3, lpcat1, napsa, sftpb, slc34a2, krt14, krt5, sftpc*	[75, 43, 90]
Natural killer (NK)	*klrd1, klrf1, clic3, cst7, fgfbp2, gnly, gzma, gzmb, hopx, klrb1, ncam1, ncr1, nkg2, nkg7, prf1*	[206, 90, 9, 152]

with the activation function (Leaky ReLU) after each layer and the forward method returns the network.

We combined different loss functions and optimizers, however the selected one were: Mean squared error (MSE) and Adam, with a learning rate = 1e-5 and the regularization method was Early stopping.

Variational autoencoders (VAE): introduced by Kingma and Welling in 2013 [98]. VAEs are generative models [42] which inherited the AE architecture. They are composed of two models supporting each other, an encoder or recognition model and a decoder or generative model. The encoder is parameterized by parameters $\boldsymbol{\phi}$, and models an approximation to its posterior $p_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$, where $\boldsymbol{z}$ is the latent representation of $\boldsymbol{x}$. The decoder model works as a generator by sampling from the distribution $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$, where $\boldsymbol{\theta}$ are the decoder neural network parameters [99]. In other words, by training a VAE, we are able to generate synthetic data that follows the distribution of the real data.

The framework of VAE provides a principled method for jointly learning deep latent-variable models and corresponding inference models using stochastic gradient descent. Latent variables are variables that are part of the model, but which can not be observed, therefore those are not part of the dataset. We use $\boldsymbol{z}$ to denote such latent variables. The marginal distribution over the observed variables $p_{\theta}(\boldsymbol{x})$, is given by:

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \int p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}, \tag{5.1}$$

where $\boldsymbol{\theta}$ parametrizes the joint and marginal distributions. This is also called the (single datapoint) marginal likelihood or the model evidence. If $\boldsymbol{z}$ is discrete and $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ is a different Gaussian distribution for each value of $\boldsymbol{z}$, then $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ is a mixture of Gaussians model. For continuous $\boldsymbol{z}$ (which is generally more efficient to work with due to the reparameterization trick), $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ can be seen as an infinite mixture, which are potentially more powerful than discrete mixtures. Such marginal distributions are also called compound probability distributions. The framework of VAEs provides a computationally efficient way for optimizing deep latent-variable

models (DLVMs) jointly with a corresponding inference model using stochastic gradient descent (SGD). The hyperparameters selected for this model were the same used in the AE model.

5.2.5 Pathways analysis of key genes across the different types of cancer

A pathway enrichment analysis was performed using the Gene Ontology Consortium database (data-version from 2020-05-02) [8, 36]. To visualize the list of GO terms and find how genes are functionally grouped we use Cytoscape v.3.8.2 with the plugin ClueGO v.2.5.7 [17] with a ($p<0.001$) and kappa statistics to calculate the relationships between the terms based on the similarity of their associated genes.

5.3 Results

5.3.1 Single cell analysis and exploration

Single-cell transcriptome profiling of tumors provides an unbiased overview of the heterogeneity of cancer cells and their microenvironment. To generate a comprehensive tumor cells atlas of human cancers, we collected scRNA-seq datasets from nine different types of cancer. A total of 20,574 cells were filtered for this study, including breast cancer, lung cancer, colorectal cancer, head and neck cancer, melanoma, glioblastoma, prostate cancer, liver cancer and squamous cell carcinoma. T cells, B cells, mast cells, macrophages, monocytes, dendritic, neutrophils, endothelials, fibroblasts, myocytes, cancer-associated fibroblast (CAF) cells, myeloids, myofibroblasts, epithelial cells and natural killer cells were reported from the different datasets. Table 5.2 shows the detail for each dataset with its ID from GEO database, the type of cancer, the type of cell that the article associated in the GEO database reported, the total number of cells and the total number of genes. Datasets GSE11472 (breast cancer), GSE103322 (head and neck cancer), GSE137829 (prostate cancer) and GSE144236 (squamous cell carcinoma) include diverse type of cells. On the other hand, we observed that T cells are the most abundant, all the datasets thta report the type of cell include these. Additionally, we calculate the pairwise intersections between the ten datasets, to analyze the data as a single reference, in order to define commonalities and harmonize annotations between the experiments. Table 5.3 shows the total of common genes between the pairwise intersection of the datasets, identified with the GEO accession number (ID). A total of 10,788 genes are common between the ten datasets. These 10,788 genes are the subset that we used for the next steps.

A first approach, was to detect the highly variable genes (HVG) across the scRNA-seq data. This analysis allows to detect genes that contribute to cell-to-cell differences in a mixed cell population. HVG assumes that if genes have large differences in expression across cells some of those differences are due to biological difference between the cells rather than technical noise [221]. This was detected by calculating the average expression and dispersion for each gene, placing these genes into bins, and then calculating a z-score for dispersion within each bin. The minimum/maximum average expression and dispersion parameters (x min, x max, y min) were then used to select the variable genes.

Table 5.2: Summary of the datasets

The Data ID corresponds to the GEO accession number, then it shows the type pf cancer, type of cells declared in the article associated in the GEO website, the total number of cells and the total number of genes.

Dataset ID	**Type of cancer**	**Type of cells**	**Total of cells**	**Total of genes**
GSE114727	Breast	Monocytes, macrophages, dendritic cells (DCs), T cells, B cells, mast cells, and neutrophils	30	20049
GSE75688	Breast	T cells, B cells and macrophages	557	34996
GSE126030	Lung	T cells	2	16296
GSE108989	Colorectal	T cells	11138	23371
GSE103322	Head and neck	T cells, B cells, macrophages, dendritic cells, mast cells, endothelial cells, fibroblasts, and myocytes	3363	23686
GSE72056	Melanoma	T cells, B cells, macrophages, CAFs and endothelial cells	4645	23684
GSE57872	Glioblastoma	Unknown	874	40124
GSE137829	Prostate	T cells, B cells, myeloid cells, mast cells, fibroblasts, myofibroblasts, endothelial cells and epithelial cells	11	25623
GSE146115	Liver	Unknown	1329	24042
GSE144236	Squamous Cell Carcinoma	Fibroblasts, melanocytes, epithelial cells, endothelial cells, B/Plasma cells, natural killer (NK), T cells and myeloid cells	13	34844

Table 5.3: Total of common genes between the datasets

Each dataset is identified with the GEO accession number (ID) and total of genes that contains. The matrix represents the number of genes that are common between pairwise intersections.

Dataset ID (Total genes)	GSE72056 (23684)	GSE75688 (34996)	GSE103322 (23686)	GSE108989 (23371)	GSE114727 (20049)	GSE126030 (16296)	GSE137829 (25623)	GSE144236 (34844)	GSE146115 (24042)
GSE57872 (40124)	19868	34558	19868	21244	19922	16248	25398	34239	23895
GSE72056 (23684)		18683	23682	21661	13394	12052	16591	18653	15419
GSE75688 (34996)			18683	20035	19828	16152	25048	32401	23564
GSE103322 (23686)				21664	13394	12052	16591	18653	15419
GSE108989 (23371)					14235	12729	17668	19990	16413
GSE114727 (20049)						14593	18573	19810	17910
GSE126030 (16296)							15612	16183	14981
GSE137829 (25623)								25065	20928
GSE144236 (34844)									23517

By default, Seurat returns 2,000 features per dataset [196]. As we know a priori that there are about 15 types of cells, Figure 5.1 shows the top 15 variable genes across the single cells of our data. These are: *cxcl10*, *fn1*, *a2m*, *lyz*, *rpl39*, *rpl36a*, *rps28*, *rpl28*, *rpl27a*, *rpl41*, *rpl9*, *dsp*, *rps24*, *mt1x* and *apoe*.

To start with dimensionality reduction, we run classical methods, such as PCA, tSNE and UMAP. When running the PCA, we observed five principal components (PCs) and it gives us a list of the genes which were the most highly and lowly weighted in the different PCs (Appendix Table 6.2). By default, we used the 2,000 most variable genes. Figure 5.2 shows the top 15 genes associated with reduction components that are highly and the top 15 genes lowly weighted in the PCs. For each subplot in the y-axis the first 15 genes are lowly weighted (left from 0) and the next 15 genes are the highly weighted genes (right from 0). The complete list is available in Appendix Table 6.2.

In order to determine the top principal components, we implemented a heuristic method called Elbow plot, a ranking of principle components based on the percentage of variance explained by each one. In our data, we can observe an elbow around PC10-20, suggesting that the majority of true signals are captured in the first 20 PCs. However, we can also observe that the top 15 PCs retain mostly of information, while other PCs contain progressively less (Figure 5.3).

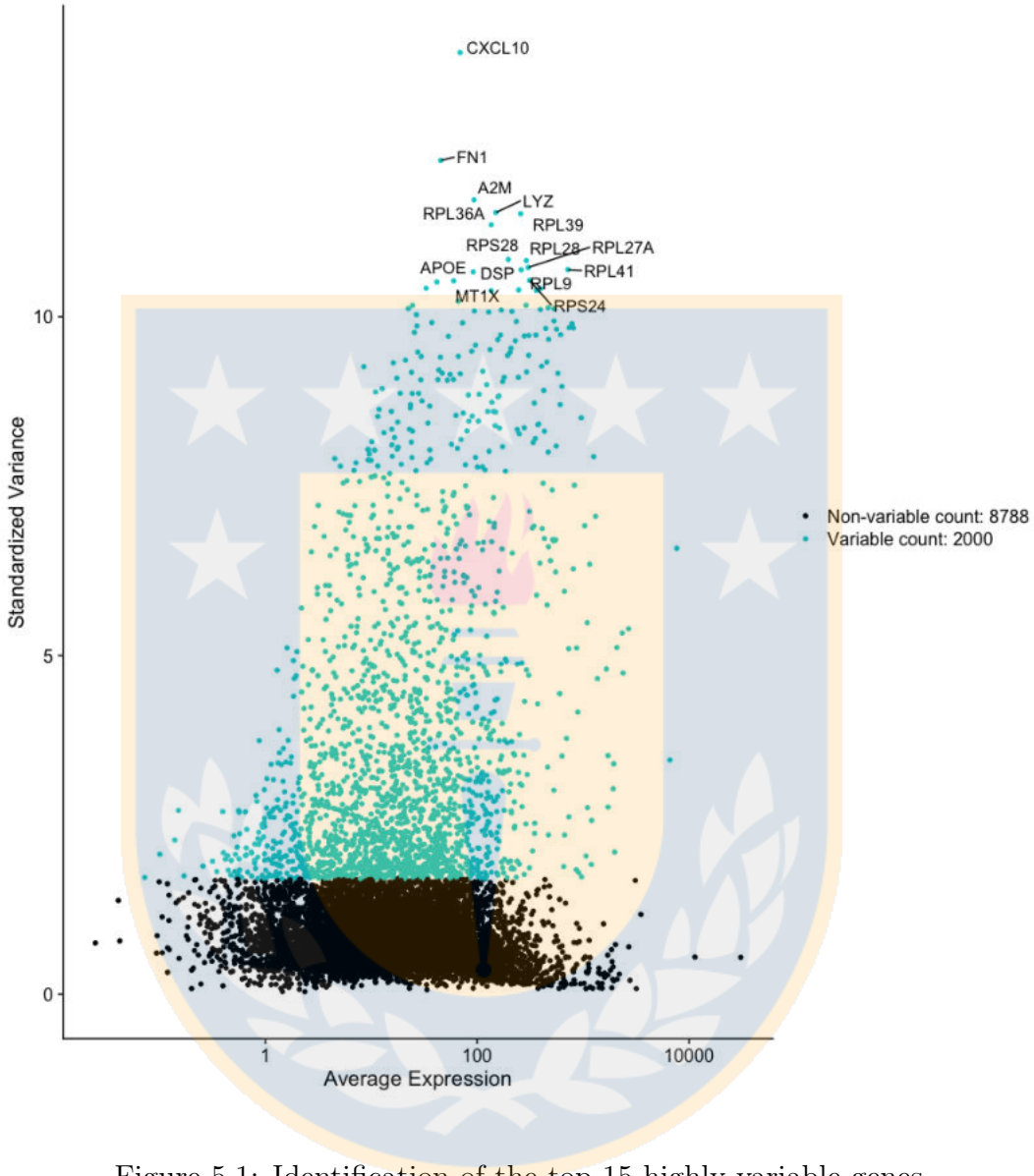


Figure 5.1: Identification of the top 15 highly variable genes

We calculate a subset of features that exhibit high cell-to-cell variation in the dataset. We observed that the 15 most highly variable genes across the cells were *cxcl10*, *fn1*, *a2m*, *lyz*, *rpl39*, *rpl36a*, *rps28*, *rpl28*, *rpl27a*, *rpl41*, *rpl9*, *dsp*, *rps24*, *mt1x* and *apoe* .

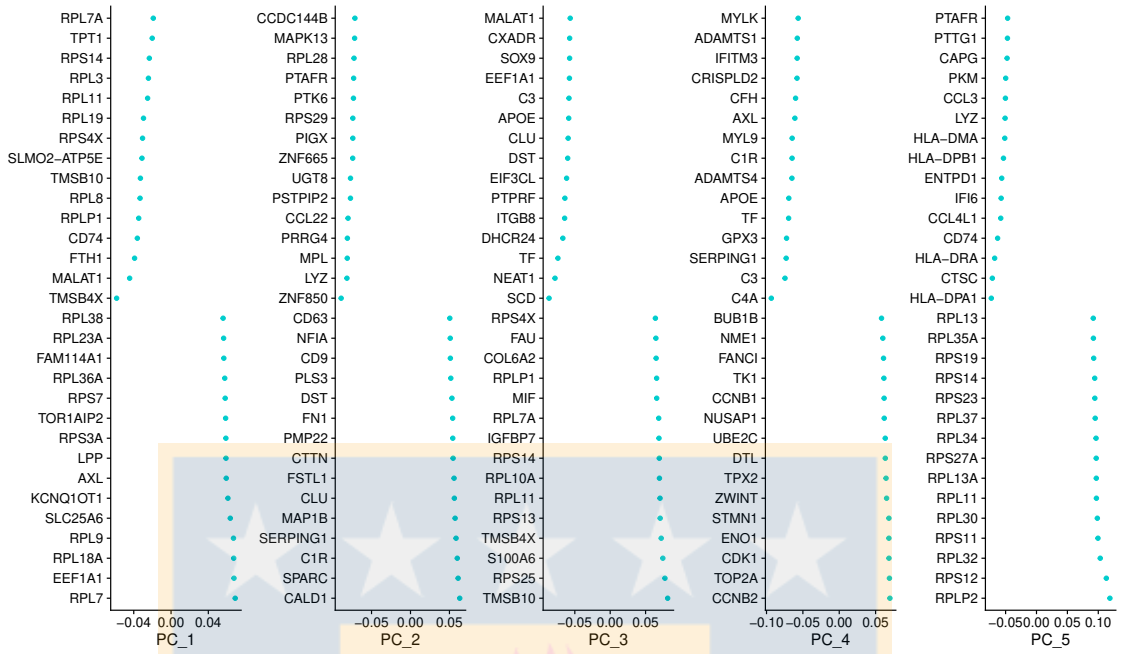


Figure 5.2: Identification of highly and lowly variable features by principal component

Each subplot shows the top 15 genes that are highly (right from) and lowly (left from 0) weighted in each principal component (PC) (x-axis).

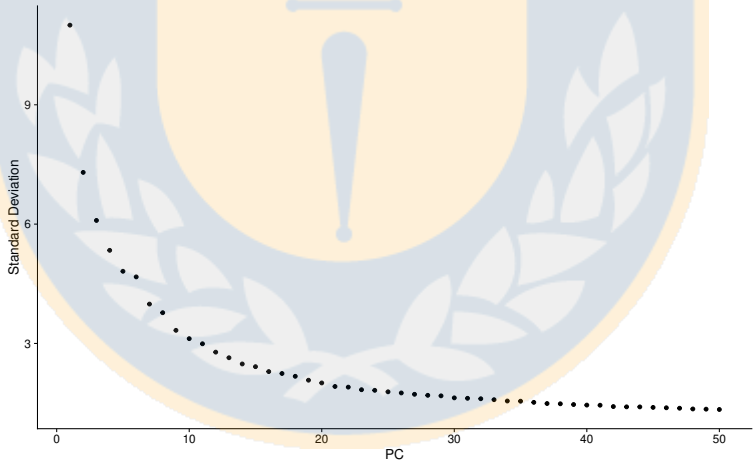


Figure 5.3: Standard deviation of each principal component in a Elbow Plot.

We ranking the principal components (PC) in the x-axis with the standard variation (y-axis) by each one.

For a more detailed view, we performed dimensionality heatmaps. These are plots of PCA weightings for the most highly and lowly weighted genes, shown against the set of cells which are most highly influenced by the PC. The idea is that as long as we are seeing clear structure in one of these plots, then we are still adding potentially useful information to the analysis (Figure 5.4).

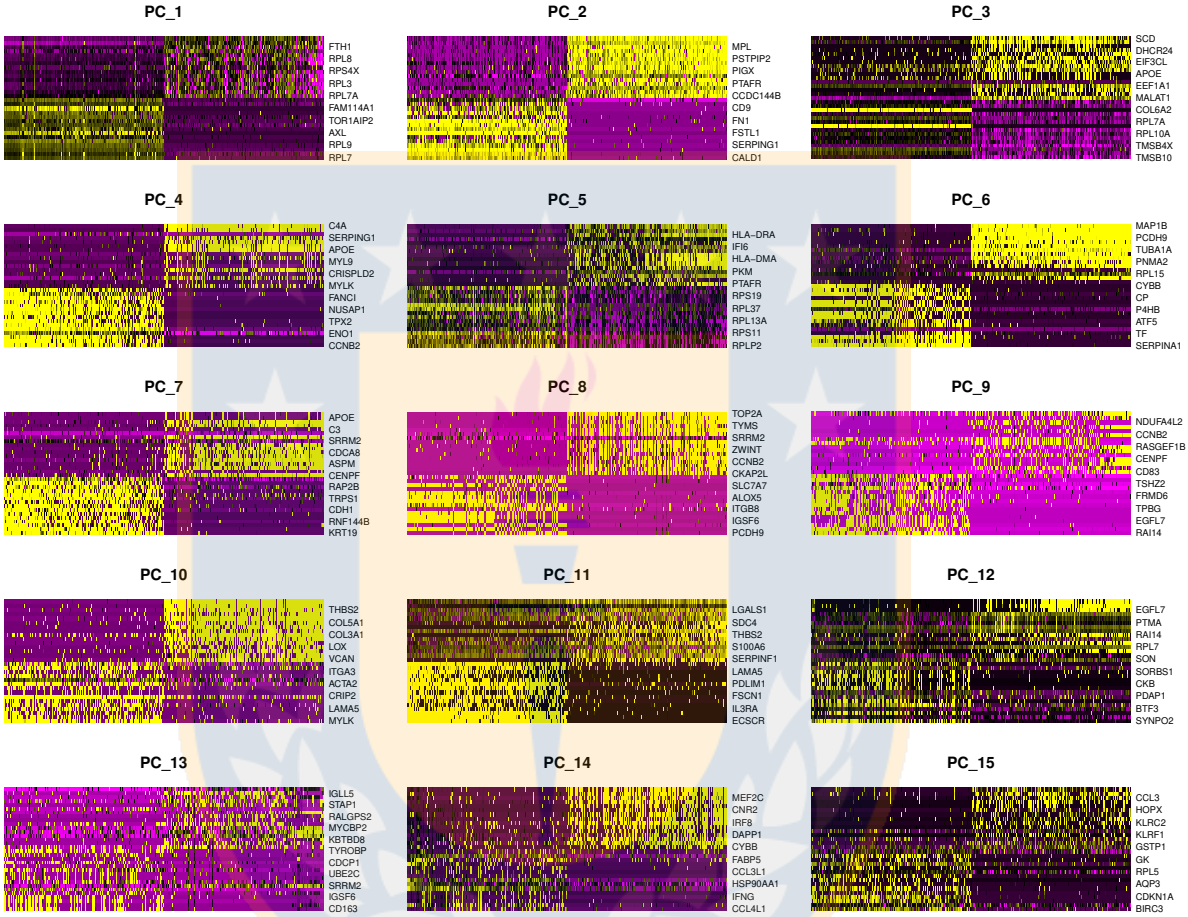


Figure 5.4: Heatmaps of the PCA matrix.

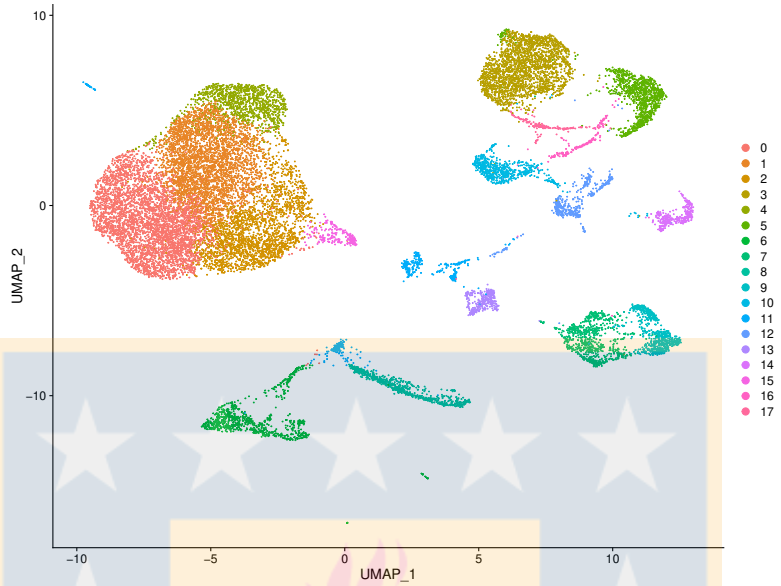
The subfigures represent the principal components (PC) from PCA analysis to visualize the top genes that contribute in each. Both cells (x-axis) and genes (y-axis) are ordered by their PC scores.

For clustering the cells, Seurat includes an approach that generates an embedding of the cells and creates a k-nearest neighbor (KNN) graph, with edges drawn between cells with similar feature expression patterns, and then attempts to partition this graph into

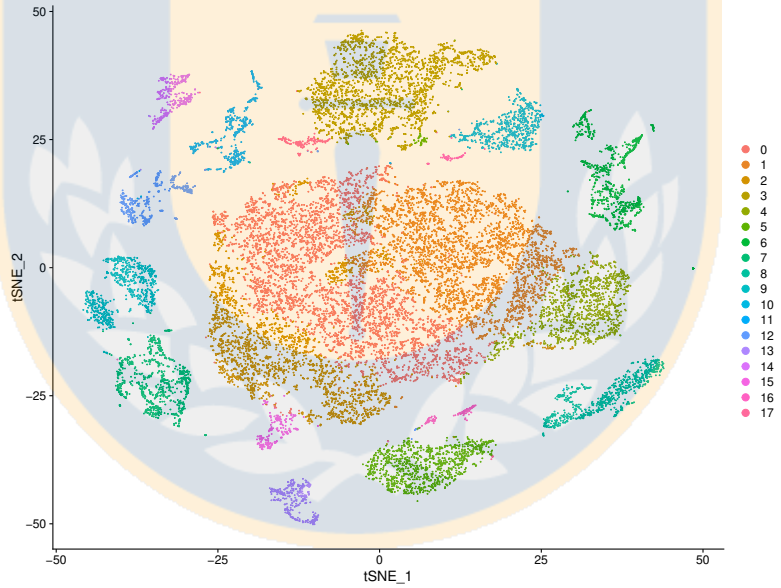
highly interconnected communities. This approach, first construct a KNN graph based on the euclidean distance in PCA space, and refine the edge weights between any two cells based on the shared overlap in their local neighborhoods (Jaccard similarity). This step is performed using the *FindNeighbors()* function, and takes as input the previously defined dimensionality of the dataset (first 15 PCs). To cluster the cells, we applied modularity optimization techniques such as the Louvain algorithm [18], to iteratively group cells together, with the goal of optimizing the standard modularity function. The *FindClusters()* function implements this procedure, and contains a resolution parameter that sets the 'granularity' of the downstream clustering, with increased values leading to a greater number of clusters. In our dataset we found 20,340 nodes and 686,168 edges. The number of communities was 18.

Other approach was apply non-linear dimensionality reduction by using UMAP and t-SNE over the original data. The overall goal of these approaches is to construct low-dimensional manifolds of high-dimensional datasets such that data entries (single cells in this case) that are similar are closer together in manifold space. UMAP generally preserves the global structure of the data instead of just local structures. Qualitatively, the UMAP plot (Figure 5.5a) separates the clusters further apart from one another, while the t-SNE plot (Figure 5.5b) does not), observing that the 10,788 cells from nine cancer types were colored by annotated cell type.

After defining the different clusters, in a first attempt we tried to identify the gene markers that defines clusters via differential expression. We found markers for every cluster compared to all remaining cells, reporting only the positive ones. We used the Wilcox rank sum test to identify genes which are differentially regulated between two groups of cells. This is a non-parametric test, which makes very few assumptions about the structure of the data and just looks for genes, which have expressions which are consistently ranked more highly in one group of cells compared to another. Figure 5.6 shows the most upregulated gene from each cluster (x-axis). The plot shows that for some clusters a gene can uniquely predict its gene marker to identify the cell type (cluster 8, 14 and 17), however, we also observe that a gene selected as a marker picks up more than one cluster, such as *ccr7, rps14, jund, tnfrsf4*. Therefore this analysis was not enough to determine the gene marker for each cluster and we cannot determine the type of cell that



(a) UMAP representation of scRNA-seq data colored by their predicted cell type.



(b) t-SNE representation of scRNA-seq data colored by their predicted cell type.

Figure 5.5: Non-linear dimensionality reduction

each one represents.

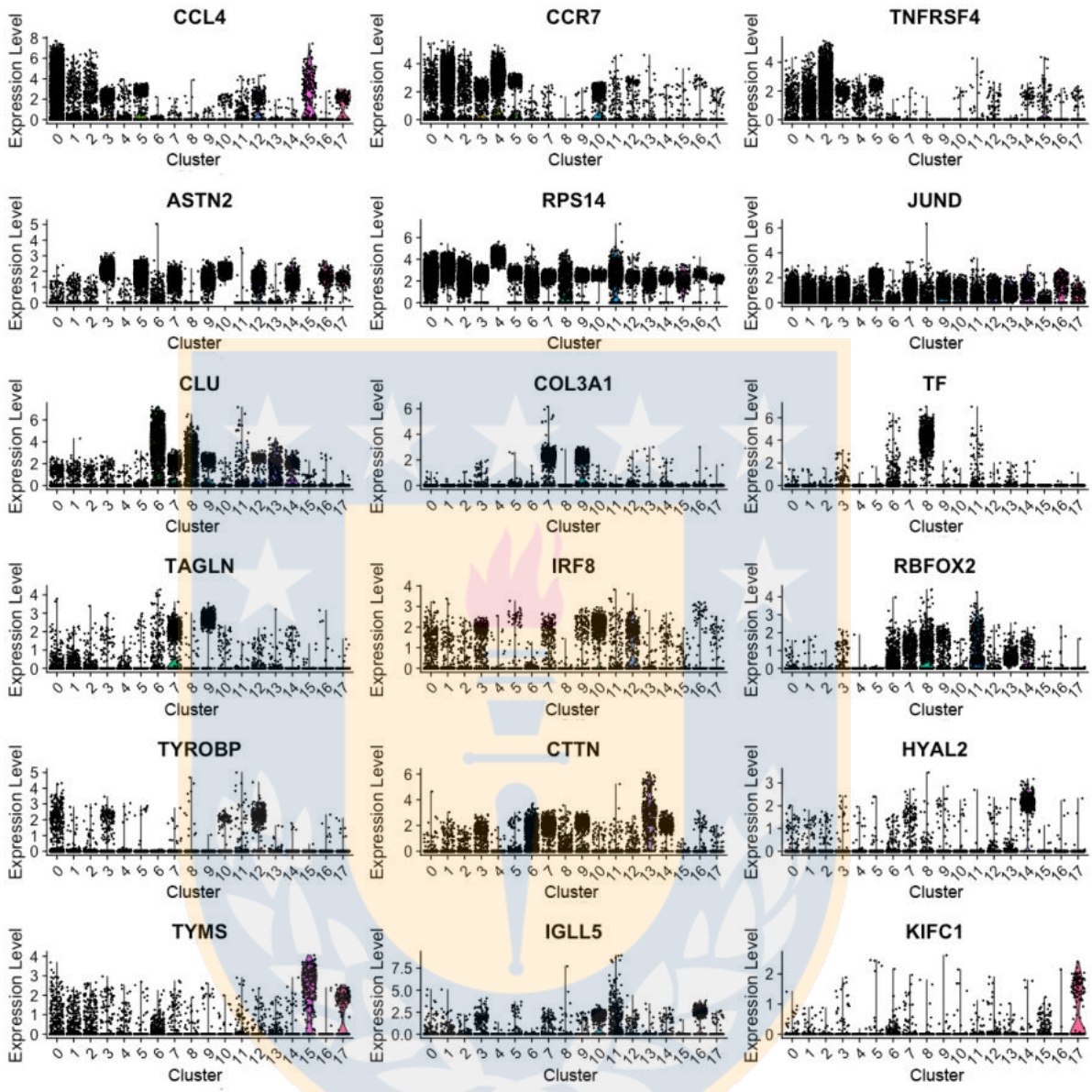


Figure 5.6: Differentially expressed genes identified as cluster biomarkers

To visualize the gene marker expressions we used the Wilcox rank sum test to identify genes which are differentially regulated between the groups of cells. X-axis represent each cluster identified in the UMAP and t-SNE representations. Y-axis shows the expression level of the genes.

To clean this up for the individual clusters, in a second attempt to discover the cell type of each cluster, we used the ROC test, a measure of how specifically a gene can predict the membership of two groups. The ROC curve is produced by calculating and plotting the true positive rate against the false positive rate for a single classifier at a variety of thresholds. The true positive rate, or sensitivity, can be represented as:

$$TPR = Sensitivity = \frac{TP}{TP + FN}, \tag{5.2}$$

where TP is the number of true positives and FN is the number of false negatives. The true positive rate is a measure of the probability that an actual positive instance will be classified as positive. The false positive rate, or 1-specificity, can be represented as:

$$FPR = 1 - Specificity = \frac{FP}{FP + TN}, \tag{5.3}$$

where FP is the number of false positives and TN is the number of true negatives. The false positive rate is essentially a measure of how often an actual negative instance will be classified as positive.

To identify gene markers using ROC test, this for each gene, evaluates (using the area under the ROC curve (AUC)) a classifier built on that gene alone, to classify between two groups of cells. An AUC value of 1 means that expression values for this gene alone can perfectly classify the two groupings (i.e. Each of the cells in cells.1 exhibit a higher level than each of the cells in cells.2). An AUC value of 0 also means there is perfect classification, but in the other direction. A value of 0.5 implies that the gene has no predictive power to classify the two groups. Returns a "predictive power" (abs(AUC-0.5) * 2) ranked matrix of putative differentially expressed genes. This is a non-parametric test, which just cares about the ranked expression measures for each gene. Figure 5.7 shows the multiple prediction for all clusters with an AUC value over ≈ 0.88 for each cluster. According to the ROC test, the genes *ccl5* (AUC = 0.936), *rps6* (AUC = 0.875), *il32* (AUC = 0.862), *loc286437* (AUC = 0.949), *rps14* (AUC = 0.982), *hla-e* (AUC = 0.942), *eef1a1* (AUC = 0.937), *c1r* (AUC = 0.942), *apoe* (AUC = 0.997), *myl9* (AUC = 0.989), *hla-dra* (AUC = 0.963), *rpl21* (AUC = 0.884), *tyrobp* (AUC = 0.927), *krt19* (AUC = 0.989), *pecam1* (AUC = 0.956), *stmn1* (AUC = 0.960), *igll5* (AUC = 0.898) and

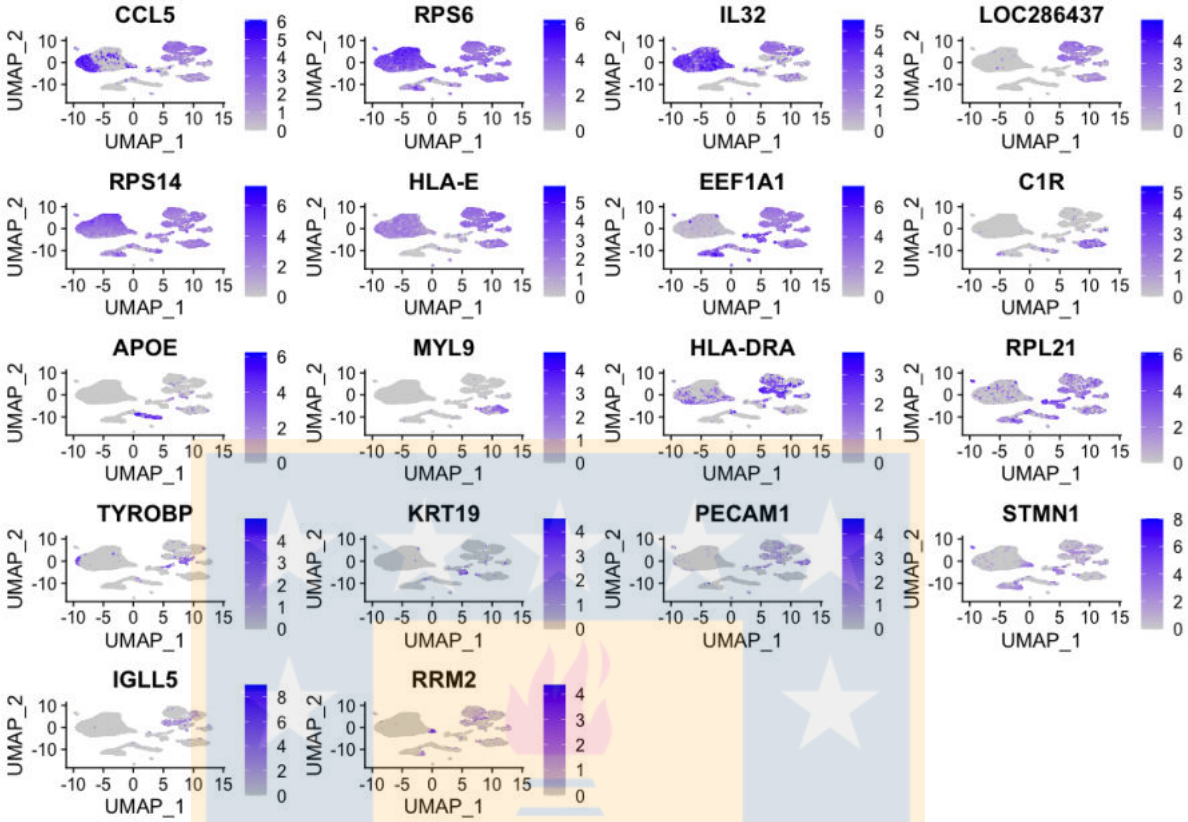


Figure 5.7: Gene markers for each cluster using ROC test

rrm2 (AUC = 0.879) are the gene markers of each cluster. Four of this genes were highly weighted in the different PCs from PCA analysis: *cctn*, *clu*, *col3a1* and *rps14*. However, we still observe that some genes are part of more than one cluster.

A third attempt to identify cell types was performing an analysis from the literature review. The dataset profiles in the GEO database indicated that there were a total of 15 cell types. Then, we plot the gene markers reported in Table 5.1 and visualize them as genes of interest onto the embedding of the UMAP dimensionality reduction performed. Nevertheless, when exploring these gene markers in our data, only 11 cell types were observed: T cells (Figure 5.8), B cells (Figure 5.9), mast (Figure 5.10), macrophages (Figure 5.11), monocytes (Figure 5.12), dendritic (Figure 5.13), neutrophils (Figure 5.14), fibroblast (Figure 5.15), CAF (Figure 5.16), epithelial (Figure 5.17) and natural killer (Figure 5.18). In the case of T cells (Figure 5.8), we visualized the genes *cd2, cd3d,*

cd3e and *cd3g*, missing the other genes mentioned in the Table 5.1. For B cells (Figure 5.9) we observed six genes from the list *cd79b, blk, hla-dpa1, hla-dra, cd37* and *cd74*. Mast cells (Figure 5.10) have a low expression and only the gene *enpp3* was observed. In macrophages (Figure 5.11) four genes were observed *cd163, fcgr3a, tyrobp* and *cd68*. For monocytes in the state-of-the-art we only found one gene marker (*vcan*) and it was expressed in our data (Figure 5.12). Dendritic cells (Figure 5.13) expressed two genes *itgae,itgax* from the list. For neutrophils we found eight gene markers in the literature, but only two genes were expressed in our data, however *cd33* has a minimum expression and *cd44* is present in different clusters. In the case of fibroblast cells (Figure 5.15) we observed two genes *col5a1* and *fbln2* expressed in our data. CAF cells (Figure 5.16) expressed three gene markers from the list *col6a1, col6a2* and *col6a3*. For epithelial cells (Figure 5.17) we observed nine gene markers from the list *acta2, mylk, tp63, sytl2, abca3, lpcat1, napsa, sftpb* and *slc34a2* but lowly expressed and in different clusters. Finally, for NK cells (Figure 5.18) we observed twelve gene markers *klrd1, klrf1, clic3, cst7, fgfbp2, gnly, gzma, gzmb, hopx, klrb1, nkg7, prf1*. Gene markers reported in Table 5.1 for endothelials, myocytes, myeloid and myofibroblast were not expressed in our data.

In summary, under the three methods used Wilcox rank sum test, ROC test and literature review to determine cell types we observed that *tyrobp*, a gene marker for macrophages was found by the methods. The genes *rps14* and *igll5* were found through Wilcox rank sum test and ROC test, and the genes *hla-dra* and *pecam1* were found through ROC test and literature review to determine B and endothelial cells.

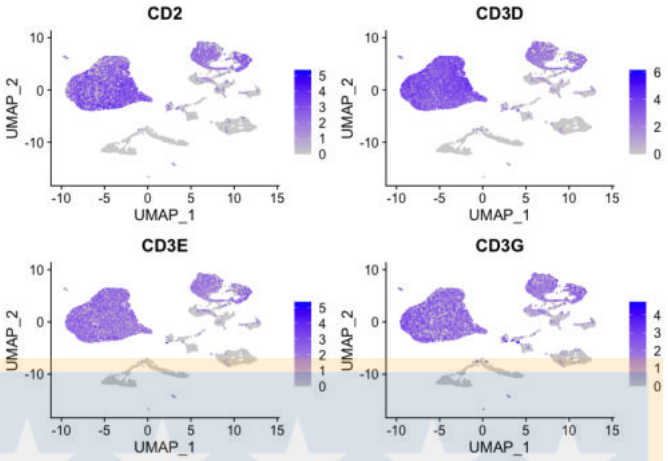


Figure 5.8: Gene markers of T cells from literature review

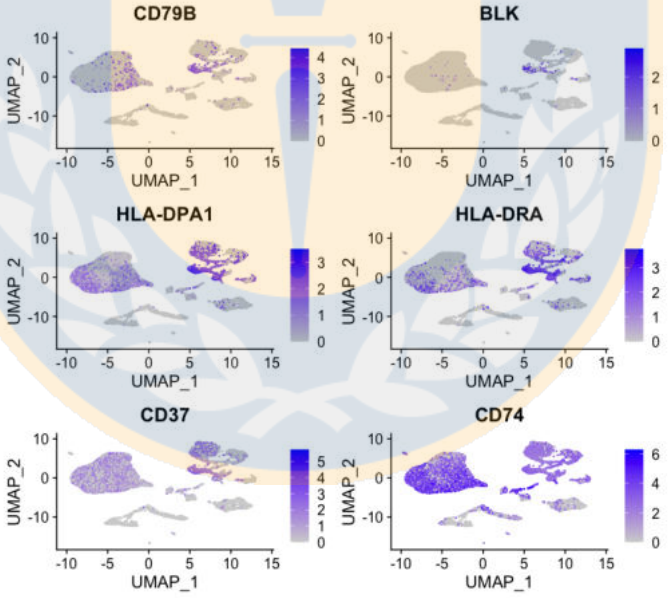


Figure 5.9: Gene markers of B cells from literature review

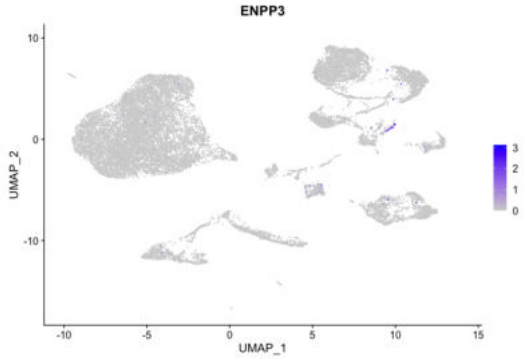


Figure 5.10: Gene markers of Mast cells from literature review

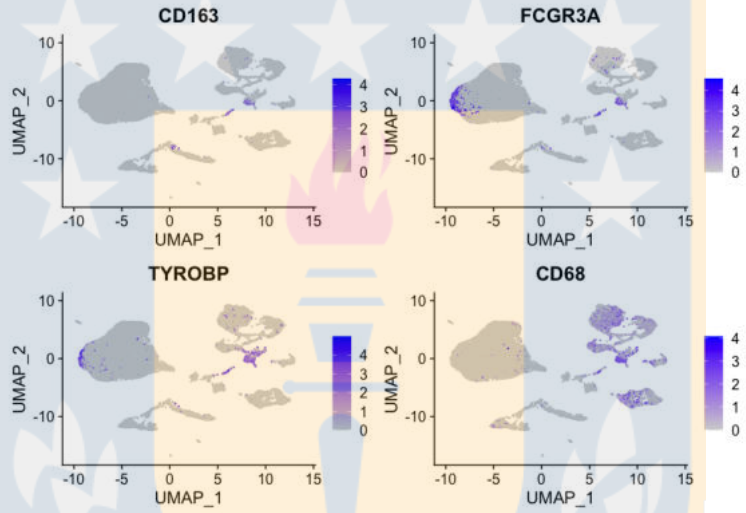


Figure 5.11: Gene markers of macrophage cells from literature review

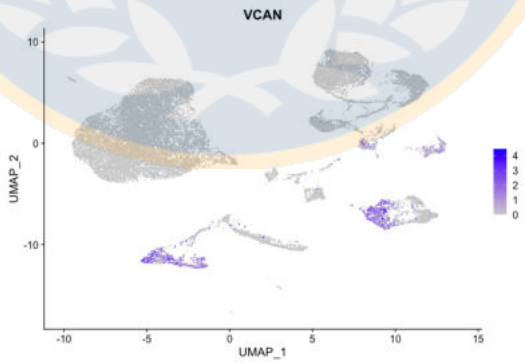


Figure 5.12: Gene markers of monocyte cells from literature review

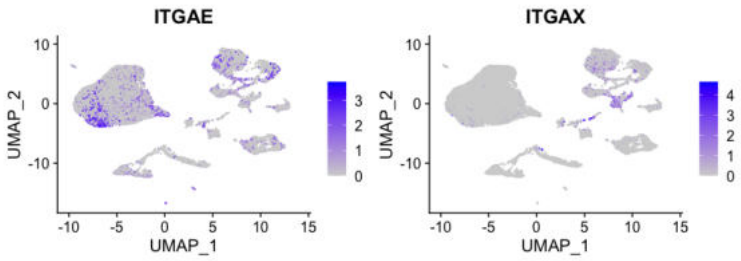


Figure 5.13: Gene markers of dendritic cells from literature review

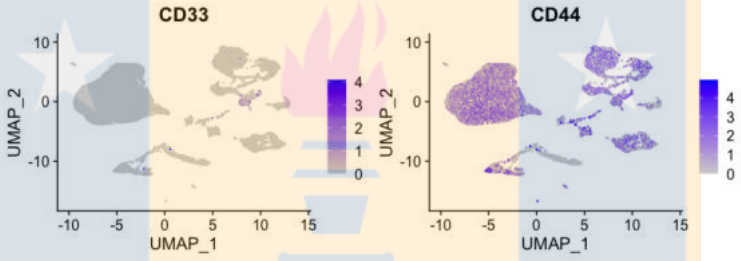


Figure 5.14: Gene markers of neutrophil cells from literature review

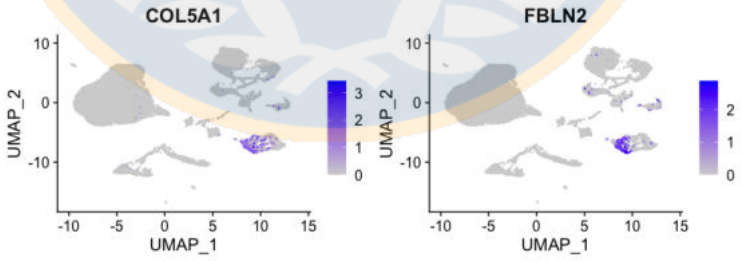


Figure 5.15: Gene markers of Fibroblast cells from literature review

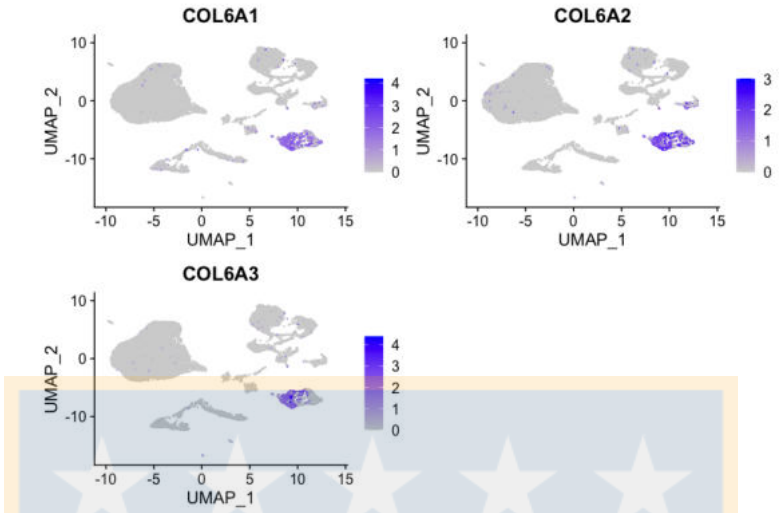


Figure 5.16: Gene markers of CAF cells from literature review

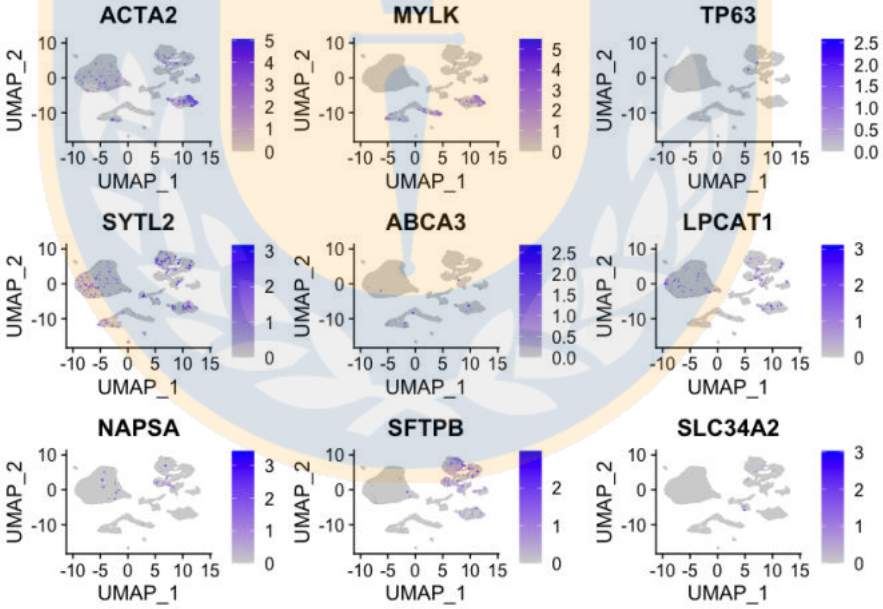


Figure 5.17: Gene markers of Epithelial cells from literature review

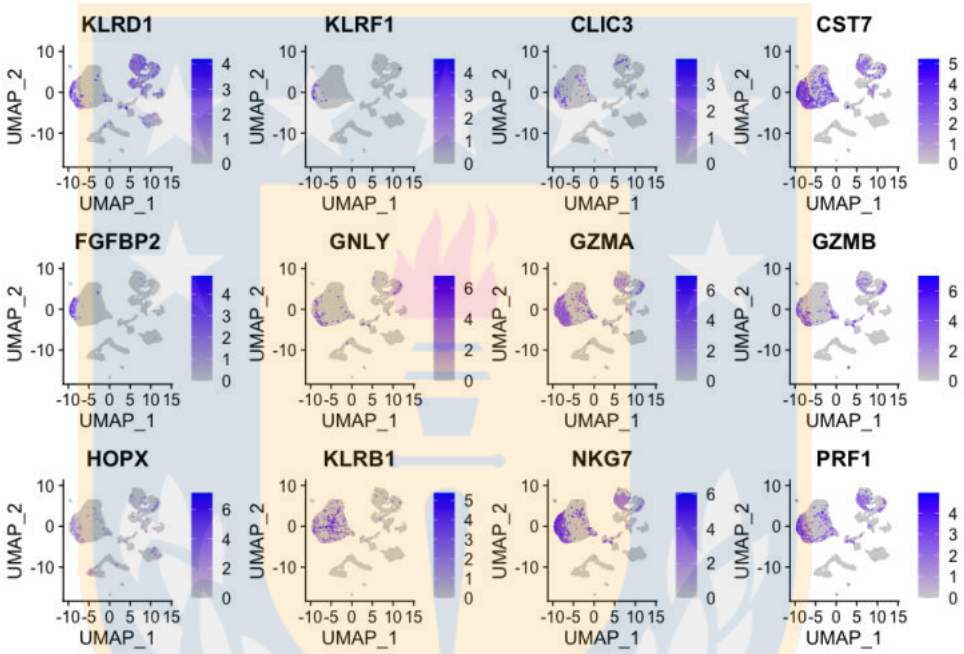


Figure 5.18: Gene markers of Natural Killer cells from literature review

5.3.2 Clustering with the DL models

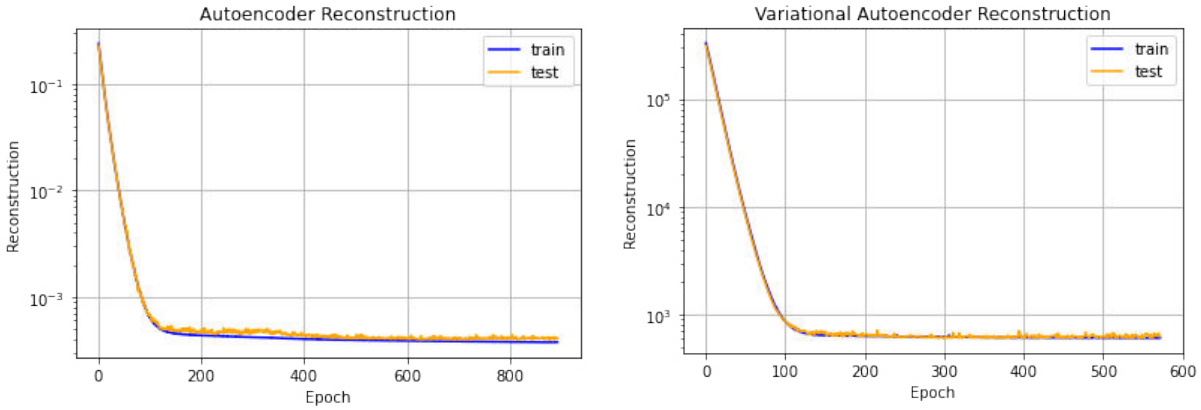
In this section, we explored and implemented two unsupervised DL models to extract relevant biological information and handle the complexities of scRNA-seq. It has been theoretically shown that the neural networks are the universal approximators capable of performing the dimensionality reduction [81].

During the data pre-processing stage, we defined a matrix where the rows correspond to the cell samples, and the columns correspond to the feature vectors containing the gene expression values. To reduce the computational complexity, we worked with the intersection of the genes across the ten datasets (10,788 genes in total). Here, we aim to provide as much data as possible for our DL algorithms to capture the true data structure across all types of cells from different cancers.

The two models that we implemented were with an autoencoder (AE) and a variational autoencoder (VAE) architecture. Both architectures are artificial neural networks that are used in unsupervised learning to automatically learn features from unlabeled data. There are multiple parameters in the DL architectures that can be optimized. These hyperparameters control the behavior of training algorithms and have a significant effect on the performance of the resulting DL models. The hyperparameter search was using the *GridSearchCV* method from Scikit-Learn and manual tuning. The selected hyperparameters were: batch size = 128, number of epochs = 10000, learning rate = 1e-5, latent dimension = 32, training optimization algorithm = Adam, activation function = leaky ReLU, loss function = MSE, and also Kullback-Leibler divergence in VAE, regularization = early stopping and patience = 200.

Figure 5.19 shows the training curves information for the AE y VAE models, indicating that the neural networks have been trained until convergence, even considering the use of early stopping.

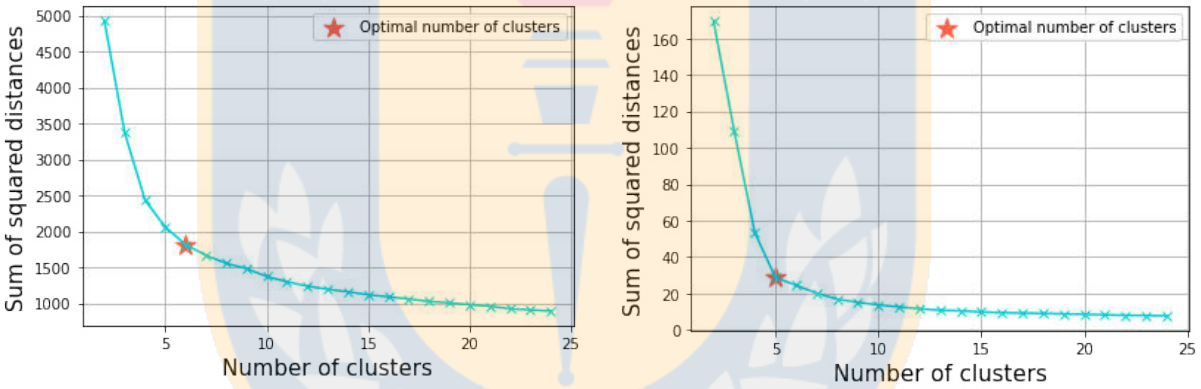
To identify the optimal number of clusters we used the Elbow method, that looks at the percentage of variance explained as a function of the number of cluster. In the Elbow method we run the K-means algorithm multiple times over a loop, with an increasing number of cluster choice (2 to 25). Clustering score is the sum of squared distances of samples to their closest center. Elbow is the point on the plot where clustering score slows down, and the value of cluster at that point give us the optimum number of clusters to



(a) Autoencoder (b) Variational autoencoder

Figure 5.19: Training curves for autoencoder and variational autoencoder models

have. Figure 5.20 shows the plots of the Elbow curve method for the (a) AE model and the (b) VAE model. In the AE we observed that the optimal numer of clusters (K) is 6, and in the VAE model it is 5.

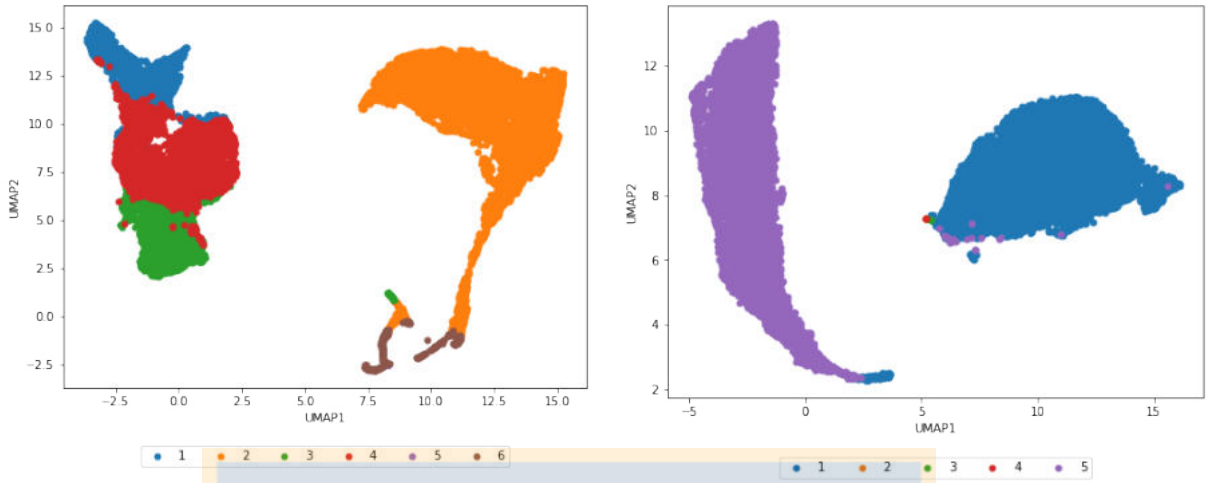


(a) Autoencoder (b) Variational autoencoder

Figure 5.20: Elbow method for optimal number of clusters

(a) Elbow method in autoencoder model tends to change slowly for a number of clusters K = 6. (b) Elbow method in variational autoencoder model tends to change slowly for a number of clusters K = 5.

Figure 5.21 shows the visualization of the two-dimensional embedding space of the ten datasets generated by UMAP for the AE and VAE models. We observe that the cluster has not relationship with the type of cells and it is difficult to determine its biological interpretation with an unsupervised method. This could be a consequence of a biological



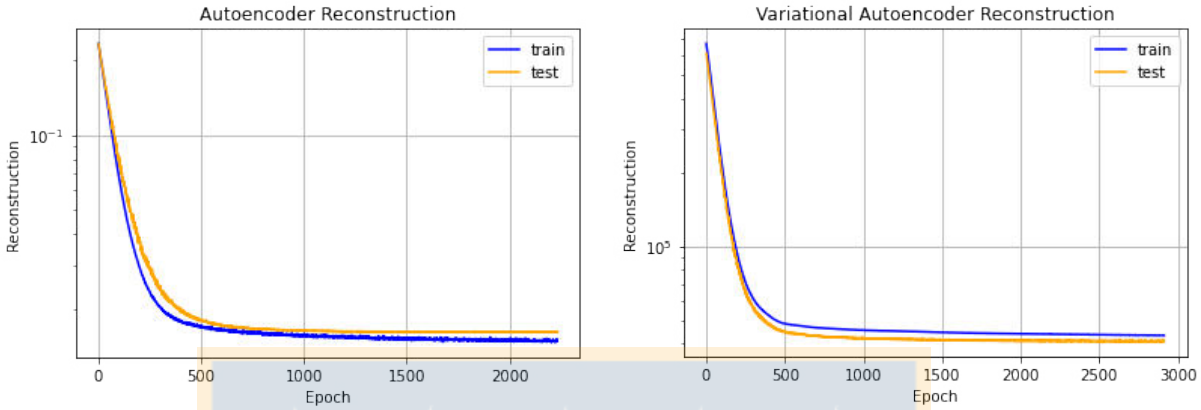
(a) Autoencoder (b) Variational autoencoder

Figure 5.21: Clustering cell using DL models

and batch effect due to the single-cell isolation techniques vary in the experiments. The most common method is the flow-activated cell sorting (FACS). A limitation of this technique include the need of different target proteins of interest, therefore given the nature and interest of each experiment the weight of the gene expression values are bias by the gene markers used [83].

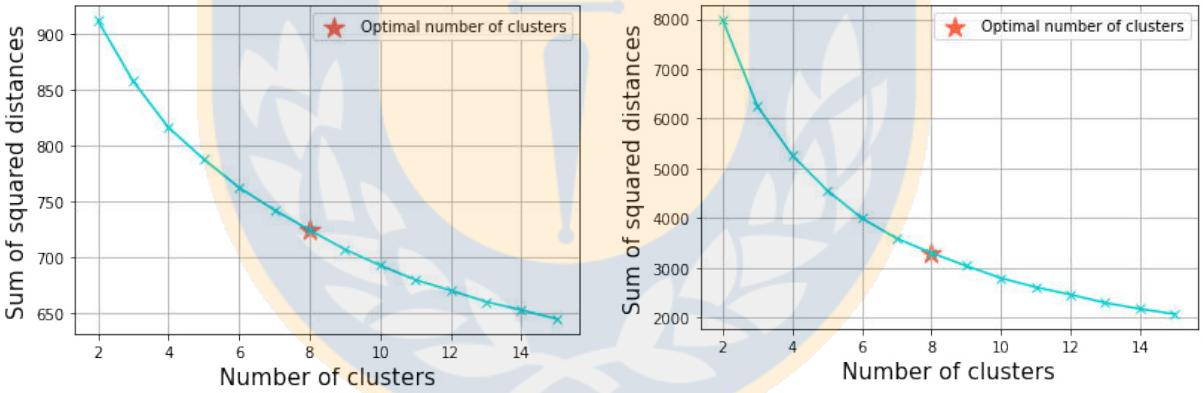
In addition, we evaluate our models only for the dataset GSE103322 (head and neck cancer) with the same hyperparameters used before. Figure 5.22 shows the training curves for the AE and VAE models indicating that the neural networks have been trained until convergence (epoch 2,200 in the AE model and 2,900 in the VAE model.) To identify the optimal number of clusters we used the Elbow method in the same way as in the previous experiment, but considering a loop with a increasing number of cluster between 2 to 16. Figure 5.23 shows the Elbow curves for the (a) AE model and the (b) VAE model, for both models we observe that the optimal number of clusters (K) is 8.

Figure 5.24 shows the visualization of the two-dimensional embedding space of the head and neck cancer dataset generated by UMAP for the AE and VAE models. The head and neck cancer data was colored by type of cells. Using the gene markers from Table 5.1 we identified the different types of cells, including fibroblast, endothelial, T cells, B cells, macrophages, dendritic, myocyte and mast cells in the clusters.



(a) Autoencoder (b) Variational autoencoder

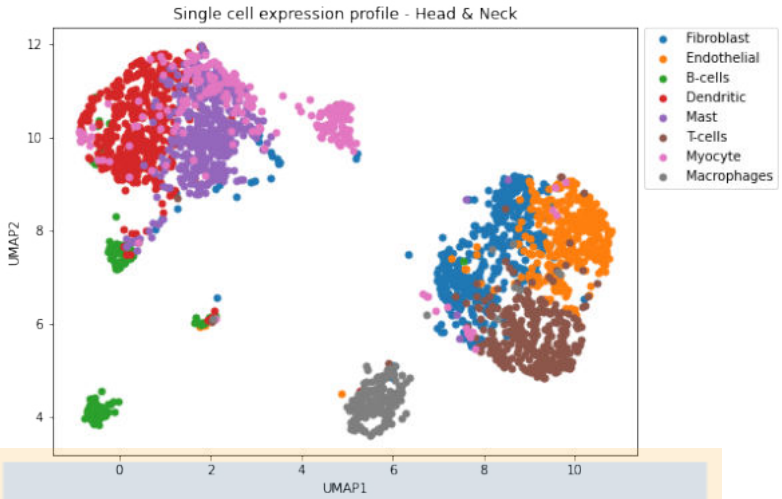
Figure 5.22: Training curves for autoencoder and variational autoencoder models on head and neck cancer data.



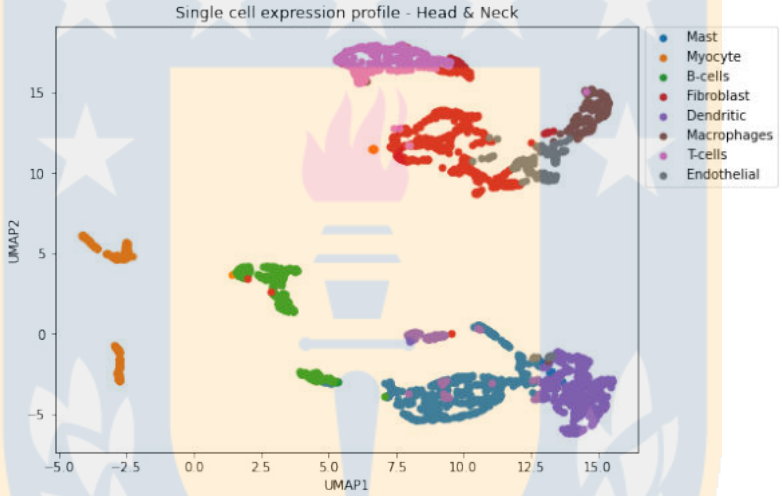
(a) Autoencoder (b) Variational autoencoder

Figure 5.23: Elbow method to calculate the optimal number of cluster on head and neck cancer data

Elbow method in (a) AE model and (b) VAE model tends to change slowly for a number of clusters K = 8.



(a) Autoencoder



(b) Variational autoencoder

Figure 5.24: Clustering cell using DL models on head and neck cancer

Single cell plot of gene expression profiles using (a) AE model and (b) VAE model. The head and neck cancer data is colored by type of cells.

Therefore, we can observe that given the biological nature of the scRNA-seq experiment, using DL it is possible to identify the different type of cells for samples coming from the same experiment. However, given that the data used in this thesis come from different laboratories, times and study interests, it compromises the integration and interpretation of the data and eventually the cell population identification.

5.3.3 Pathways analysis of key genes across datasets

To understand the variation in the observed clusters in terms of its biological functions, we performed a pathway analysis. GO annotations and biological pathways were analyzed using the key genes highly weighted in the different principal components from previous analysis (Section 5.3.1). We observed 10 clusters in the interaction network for biological processes terms using the Cytoscape software and ClueGO plugin. Figure 5.25 indicates that the main biological process are: SRP-dependent cotranslational protein targeting to membrane, chromosome condensation, ribosome biogenesis, negative regulation of complement activation, platelet degranulation, regulation of intrinsic apoptotic signaling pathway in response to DNA damage, regulation of intrinsic apoptotic signaling pathway in response to DNA damage, sequestering of actin monomers, cytoplasmic translation, peptide cross linking and positive regulation of axon extension.

5.4 Discussion

DL is a promising approach for the study of complex biological systems, but issues remain regarding our ability to delineate the learned biology from these models. In general, we only find a weak correlation when calculating the number of clusters and the previous knowledge from the state-of-the-art. In this analysis it was necessary to explore the data according to the gene markers that we found. With unsupervised methods, we can build models, but then it is needed to interrogate the latent dimension to make interpretations.

Analysis of the heterogeneity within tumors normally enables two perspectives: cells are grouped into cell types or in the diversity of the cell state that may be assigned to distinct phases based on relative expression of cell-cycle. However, in our dataset, a possible insight should be looking into the different classes of tumors or in the tumor microenvironment.

Similar to the results from Chapter 4, using the results from PCA analysis and regardless of the type of cancer, we observe that common functions associated with SRP components have been correlated with a growing list of diseases, such as cancer progression, myopathies and bone marrow genetic diseases, suggesting a potential for development of

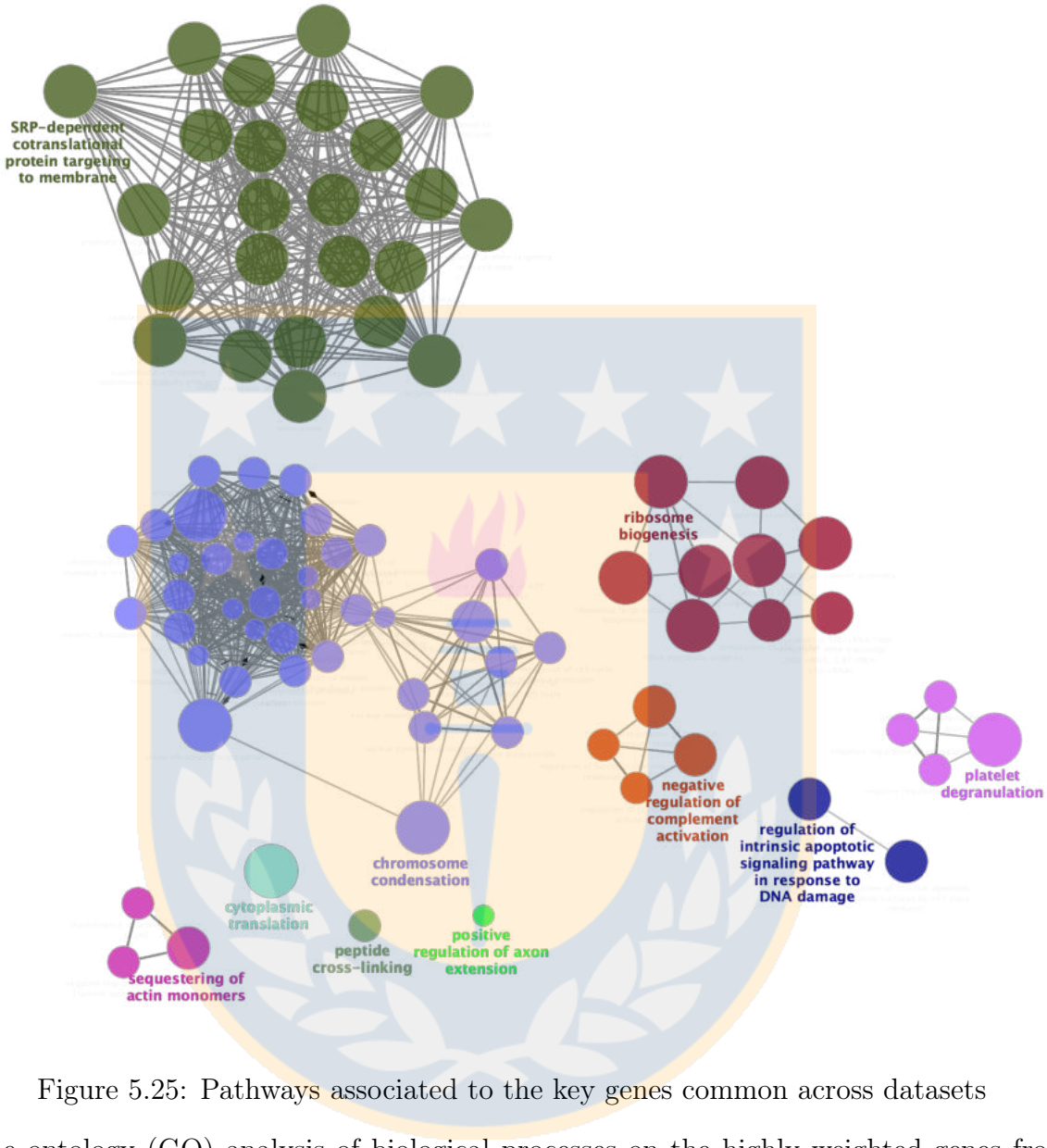


Figure 5.25: Pathways associated to the key genes common across datasets

Gene ontology (GO) analysis of biological processes on the highly weighted genes from PCA analysis. The ClueGO plugin of Cytoscape permitted find a rich cluster of over-represented GO processes and a network, where each node represents a GO biological process, and the colors refer to the GO group. The edges reflect the relationships between the terms based on the similarity of their associated genes.

SRP-target therapies of each individual component [28]. Chromosome condensation is another important term, that plays a key role in oncogenesis of most cancers, determining the regulation of tumor cell proliferation, invasion, metastasis, and radio-chemotherapeutic resistance [67]. The third biological process, ribosome biogenesis, is one of the most multifaceted and energy-demanding processes because it involves a large number of assembly and maturation factors, elevating cancer risk [163]. Another important process in our data was the negative regulation of complement activation, that plays a key role in the immune system that has developed as a first defense against non-self cells, due to inhibitory mechanisms of complement activation allow cancer cells to escape from complement-mediated elimination and hamper the clinical efficacy of monoclonal antibody–based cancer immunotherapies [167]. On the other hand, platelet degranulation plays an important role in response to tumor cell stimuli profoundly influences biology in the cardiovascular system. Numerous growth factors are released from α-granules during the formation of tumor cell-platelet emboli that stimulate tumor cell growth and angiogenesis and also initiate and regulate microenvironmental and systemic immune responses during the formation of tumors, therefore, this pathway influences tumor cell-induces biological changes in the vascular microenvironment that can be promoted by calcium and magnesium ions [140]. Finally, the pathways observed in our data indicate which are closely related to tumor cell proliferation, progression and regulation of the microenvironmental and systemic immune response.

5.5 Conclusions

By applying machine learning and DL models, we found over 10,000 common genes across nine different types of cancer that present a diversity of cells that helps us understand the heterogeneity between tumors. The computational method applied by calculating cell-specific modality weights, allows us to perform a downstream analysis on a weighted combination of data that finally improved the characterization of the cellular diversity.

We explored the response of the immune system according to the pathways associated with the key genes discovered in our analysis that play an important role in tumor proliferation, progression and regulation of the microenvironmental immune response integrating 15 types of cells.

However, a limitation of the study was to integrate high dimensionality data from different experiments. Using gene markers from the state-of-the-art and machine learning tools provided best overview of the markers that may be used in the flow cytometry and we identified the type of cell in the clusters. On the other hand, the DL models can classify the type of cells for a single experiment or dataset, but it does not perform well when we analyze the ten datasets as a unique matrix. In addition, our models give us another perspective about the cluster, where we observed that the clusters are grouped in the diversity of the tumor microenvironment.

Chapter 6

Conclusions and Future Work

Machine learning and deep learning methods have proven to be useful in modern oncology research for accurate decision-making, predicting cancer prognosis and tumor growth, and classifying cells and/or tissues, among other tasks. This thesis was useful to investigate clustering methods for cancer analysis, using publicly available scRNA-seq data. However, working with a high dimensionality of data coming from different experiments is a complex scenario when the task is to work with unsupervised methods.

In a first approach we applied machine learning methods for dimensionality reduction and pathway analysis to integrate a large amount of data to identify common genetic T cells signatures across five different types of cancers. We have analyzed CD4-T, CD8-T and Treg cells, that are subpopulations of T cells, and were isolated from melanoma, breast, lung, colorectal and head and neck cancer. After dimensionality reduction, clustering and selection of the different subpopulations from malignant and non-malignant datasets, we compared those T cell signatures and their core dynamics pathways between malignant and non-malignant samples to identify unique and common pathways in CD4-T, CD8-T and Tregs. Our analysis revealed that pathways related to the immune response, metabolism and viral immunoregulation were observed exclusively in cancer samples. Several other pathways were identified in all three T cell subpopulations, however future research is required to understand whether these pathways favor effective anti-tumor responses, or they are impaired and therefore do not prevent tumor progression.

A second approach was to apply DL models to classify cells from nine different types of cancers (breast, lung, colorectal, head and neck, melanoma, glioblastoma, prostate, liver and squamous cell carcinoma) as an unsupervised method. However, given the nature of the scRNA-seq experiment, DL models performed well when we analyzed a single experiment and not a combination of experiments. Alternatively, with the same data, we applied machine learning models to classify the cells, considering a list of gene markers

from the literature review, in order to determine the type of cell in these models, but we observed noise and overlapping of the gene markers on the clusters. Furthermore, we performed a pathway analysis using the key genes highly weighted in the five principal components (from PCA analysis). We observed ten clusters in the interaction network of biological process terms obtained from Cytoscape software and ClueGO plugin. The main biological processes observed in our data indicate which are closely related to tumor cell proliferation, progression and regulation of the microenvironmental and systemic immune response.

To summarize, our analysis of tumour infiltrating T-cells revealed unique pathways related to the immune response, metabolism and viral immunoregulation exclusively in cancer samples. Moreover, our second approach led us to explore the challenges of applying DL methods to scRNA-Seq, where with machine learning we can explore in detail gene markers of our interest. However, machine learning and deep learning are promising and powerful tools for analyzing different types of data, particularly useful when addressing problems in cancer research such as cancer prediction, prevention, diagnosis, prognosis and even therapy.

Future lines of research in Chapter 4 must be to validate the identified pathways with our in-house experiments of RNA-seq and proteomic data obtained from T cell subsets cultured under malignant environment, processed in the Molecular & Translational Immunology Lab, Department of Clinical Biochemistry & Immunology, Pharmacy Faculty, University of Concepcion.

Another future work, could be the creation of an atlas for the classification of the different cells studied in this thesis. This could be under a semi-supervised approach, combining the labeled data that we debugged in Chapter 4 and 5. Additionally, we could redefine the training data, avoiding data coming from few cell types, thus we could improve our deep learning models.

References

[1] Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, and Gerald Penn. Applying convolutional neural networks concepts to hybrid nn-hmm model for speech recognition. In *2012 IEEE international conference on Acoustics, speech and signal processing (ICASSP)*, pages 4277–4280. IEEE, 2012.

[2] Claudio Acuña-Castillo, Mabel Vidal, Ailen Inostroza-Molina, Eva Vallejos-Vidal, Roberto Luraschi, Maximiliano Figueroa, Carlos Barrera-Avalos, Rodrigo Vera, Sergio Vargas, Daniel Valdes, et al. First identification of reinfection by a genetically different variant of sars-cov-2 in a homeless person from the metropolitan area of santiago, chile. *Journal of environmental and public health*, 2022, 2022.

[3] Mübeccel Akdis, Alar Aab, Can Altunbulakli, Kursat Azkur, Rita A. Costa, Reto Crameri, Su Duan, Thomas Eiwegger, Andrzej Eljaszewicz, Ruth Ferstl, Remo Frei, Mattia Garbani, Anna Globinska, Lena Hess, Carly Huitema, Terufumi Kubo, Zsolt Komlosi, Patricia Konieczna, Nora Kovacs, Umut C. Kucuksezer, Norbert Meyer, Hideaki Morita, Judith Olzhausen, Liam O'Mahony, Marija Pezer, Moira Prati, Ana Rebane, Claudio Rhyner, Arturo Rinaldi, Milena Sokolowska, Barbara Stanic, Kazunari Sugita, Angela Treis, Willem van de Veen, Kerstin Wanke, Marcin Wawrzyniak, Paulina Wawrzyniak, Oliver F. Wirz, Josefina Sierra Zakzuk, and Cezmi A. Akdis. Interleukins (from il-1 to il-38), interferons, transforming growth factor β, and tnf-α: Receptors, functions, and roles in diseases. *Journal of Allergy and Clinical Immunology*, 138(4):984–1010, 2016.

[4] Shafiq Alam, Gillian Dobbie, Yun Sing Koh, Patricia Riddle, and Saeed Ur Rehman. Research on particle swarm optimization based clustering: a systematic review of literature and techniques. *Swarm and Evolutionary Computation*, 17:1–13, 2014.

[5] Simon Andrews et al. Fastqc: a quality control tool for high throughput sequence data, 2010.

[6] Marcin Andrychowicz, Misha Denil, Sergio Gómez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando de Freitas. Learning to learn by gradient descent by gradient descent. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 3981–3989. Curran Associates, Inc., 2016.

[7] Christof Angermueller, Tanel Pärnamaa, Leopold Parts, and Oliver Stegle. Deep learning for computational biology. *Molecular systems biology*, 12(7):878, 2016.

[8] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T

Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.

[9] Elham Azizi, Ambrose J Carr, George Plitas, Andrew E Cornish, Catherine Konopacki, Sandhya Prabhakaran, Juozas Nainys, Kenmin Wu, Vaidotas Kiseliovas, Manu Setty, et al. Single-cell map of diverse immune phenotypes in the breast tumor microenvironment. *Cell*, 174(5):1293–1308, 2018.

[10] Rhonda Bacher and Christina Kendziorski. Design and computational analysis of single-cell rna-sequencing experiments. *Genome Biology*, 17(1):63, 2016.

[11] Carlos Barrera-Avalos, Roberto Luraschi, Claudio Acuña-Castillo, Mabel Vidal, Andrea Mella-Torres, Ailen Inostroza-Molina, Rodrigo Vera, Sergio Vargas, Iván Hernández, Christian Perez, et al. Description of symptoms caused by the infection of the sars-cov-2 b. 1.621 (mu) variant in patients with complete coronavac vaccination scheme: First case report from santiago of chile. *Frontiers in Public Health*, 10, 2022.

[12] DO Bates and SJ Harper. Regulation of vascular permeability by vascular endothelial growth factors. *Vascular pharmacology*, 39(4-5):225–237, 2002.

[13] Mariana Belgiu and Lucian Drăguţ. Random forest in remote sensing: A review of applications and future directions. *ISPRS journal of photogrammetry and remote sensing*, 114:24–31, 2016.

[14] Yoshua Bengio. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, jan 2009.

[15] Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade*, pages 437–478. Springer, 2012.

[16] Rajendra Rana Bhat, Vivek Viswanath, and Xiaolin Li. Deepcancer: Detecting cancer via deep generative learning through gene expressions. *2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress*, pages 901–908, 2017.

[17] Gabriela Bindea, Bernhard Mlecnik, Hubert Hackl, Pornpimol Charoentong, Marie Tosolini, Amos Kirilovsky, Wolf-Herman Fridman, Franck Pagès, Zlatko Trajanoski, and Jérôme Galon. Cluego: a cytoscape plug-in to decipher functionally grouped gene ontology and pathway annotation networks. *Bioinformatics*, 25(8):1091–1093, 2009.

[18] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[19] Léon Bottou. Stochastic gradient descent tricks. In *Neural networks: Tricks of the trade*, volume 7700, pages 430–445. Springer, 2012.

[20] Arthur E Bryson. A gradient method for optimizing multi-stage allocation processes, 1961.

[21] Anuradha Budhu, Marshonna Forgues, Qing-Hai Ye, Hu-Liang Jia, Ping He, Krista A Zanetti, Udai S Kammula, Yidong Chen, Lun-Xiu Qin, Zhao-You Tang, et al. Prediction of venous metastases, recurrence, and prognosis in hepatocellular carcinoma based on a unique immune response signature of the liver microenvironment. *Cancer cell*, 10(2):99–111, 2006.

[22] Raquel Buj and Katherine M Aird. Deoxyribonucleotide triphosphate metabolism in cancer and metabolic disease. *Frontiers in endocrinology*, 9:177, 2018.

[23] Tilmann Bürckstümmer, Christoph Baumann, Stephan Blüml, Evelyn Dixit, Gerhard Dürnberger, Hannah Jahn, Melanie Planyavsky, Martin Bilban, Jacques Colinge, Keiryn L Bennett, et al. An orthogonal proteomic-genomic screen identifies aim2 as a cytoplasmic dna sensor for the inflammasome. *Nature immunology*, 10(3):266, 2009.

[24] Massimo Buscema. Back propagation neural networks. *Substance use & misuse*, 33(2):233–270, 1998.

[25] Diogo M Camacho, Katherine M Collins, Rani K Powers, James C Costello, and James J Collins. Next-generation machine learning for biological networks. *Cell*, 2018.

[26] Chensi Cao, Feng Liu, Hai Tan, Deshou Song, Wenjie Shu, Weizhong Li, Yiming Zhou, Xiaochen Bo, and Zhi Xie. Deep learning and its applications in biomedicine. *Genomics, Proteomics & Bioinformatics*, 16(1):17 – 32, 2018.

[27] Ronald A Castellino. Computer aided detection (cad): an overview. *Cancer Imaging*, 5(1):17, 2005.

[28] L Nicolas Gonzalez Castro, Itay Tirosh, and Mario L Suvà. Decoding cancer biology one cell at a time. *Cancer discovery*, 11(4):960–970, 2021.

[29] Vijender Chaitankar, Gökhan Karakülah, Rinki Ratnapriya, Felipe O Giuste, Matthew J Brooks, and Anand Swaroop. Next generation sequencing technology and genomewide data analysis: Perspectives for retinal research. *Progress in retinal and eye research*, 55:1–31, 2016.

[30] Cheng-Chang Chang, Kuo-Min Su, Kai-Hsi Lu, Chi-Kang Lin, Peng-Hui Wang, Hsin-Yang Li, Mong-Lien Wang, Cheng-Kuo Lin, Mu-Hsien Yu, and Chia-Ming

Chang. Key immunological functions involved in the progression of epithelial ovarian serous carcinoma discovered by the gene ontology-based immunofunctionome analysis. *International journal of molecular sciences*, 19(11):3311, 2018.

[31] Min Chen, Simone A Ludwig, and Keqin Li. Clustering in big data. In *Big Data Management and Processing*, pages 333–346. Chapman and Hall/CRC, 2017.

[32] Woosung Chung, Hye Hyeon Eum, Hae-Ock Lee, Kyung-Min Lee, Han-Byoel Lee, Kyu-Tae Kim, Han Suk Ryu, Sangmin Kim, Jeong Eon Lee, Yeon Hee Park, et al. Single-cell rna-seq enables comprehensive tumour and immune cell profiling in primary breast cancer. *Nature communications*, 8(1):1–12, 2017.

[33] Lena Claesson-Welsh. Vascular permeability—the essentials. *Upsala journal of medical sciences*, 120(3):135–143, 2015.

[34] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, pages 1–14, 2016.

[35] Kevin C Conlon, Milos D Miljkovic, and Thomas A Waldmann. Cytokines in the treatment of cancer. *Journal of Interferon & Cytokine Research*, 39(1):6–21, 2019.

[36] Gene Ontology Consortium. The gene ontology resource: 20 years and still going strong. *Nucleic acids research*, 47(D1):D330–D338, 2019.

[37] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. Binaryconnect: Training deep neural networks with binary weights during propagations. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 3123–3131. Curran Associates, Inc., 2015.

[38] Agnes E Coutinho and Karen E Chapman. The anti-inflammatory and immunosuppressive effects of glucocorticoids, recent developments and mechanistic insights. *Molecular and cellular endocrinology*, 335(1):2–13, 2011.

[39] Padideh Danaee, Reza Ghaeini, and David Hendrix. A deep learning approach for cancer detection and relevant gene identification. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, 22:219–229, 2017.

[40] Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Andrew Senior, Paul Tucker, Ke Yang, Quoc V Le, et al. Large scale distributed deep networks. In *Advances in neural information processing systems*, pages 1223–1231, 2012.

[41] Alexander Dobin, Carrie A Davis, Felix Schlesinger, Jorg Drenkow, Chris Zaleski, Sonali Jha, Philippe Batut, Mark Chaisson, and Thomas R Gingeras. Star: ultrafast universal rna-seq aligner. *Bioinformatics*, 29(1):15–21, 2013.

[42] Carl Doersch. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*, 2016.

[43] Baijun Dong, Juju Miao, Yanqing Wang, Wenqin Luo, Zhongzhong Ji, Huadong Lai, Man Zhang, Xiaomu Cheng, Jinming Wang, Yuxiang Fang, et al. Single-cell analysis supports a luminal-neuroendocrine transdifferentiation in human prostate cancer. *Communications biology*, 3(1):1–15, 2020.

[44] Glenn Dranoff. Cytokines in cancer pathogenesis and cancer therapy. *Nature Reviews Cancer*, 4(1):11–22, 2004.

[45] Stuart E Dreyfus. Artificial neural networks, back propagation, and the kelley-bryson gradient procedure. *Journal of guidance, control, and dynamics*, 13(5):926–928, 1990.

[46] Yue Du, Roy Zhang, Abolfazl Zargari, Theresa C Thai, Camille C Gunderson, Katherine M Moxley, Hong Liu, Bin Zheng, and Yuchen Qiu. A performance comparison of low-and high-level features learned by deep convolutional neural networks in epithelium and stroma classification. In *Medical Imaging 2018: Digital Pathology*, volume 10581, page 1058116. International Society for Optics and Photonics, 2018.

[47] Charles Dugas, Yoshua Bengio, François Bélisle, Claude Nadeau, and René Garcia. Incorporating second-order functional knowledge for better option pricing. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 472–478. MIT Press, 2001.

[48] Michael Egmont-Petersen, Dick de Ridder, and Heinz Handels. Image processing with neural networks—a review. *Pattern recognition*, 35(10):2279–2301, 2002.

[49] Frank Emmert-Streib, Ricardo de Matos Simoes, Galina Glazko, Simon McDade, Benjamin Haibe-Kains, Andreas Holzinger, Matthias Dehmer, and Frederick Charles Campbell. Functional and genetic analysis of the colon cancer network. *BMC bioinformatics*, 15(S6):S6, 2014.

[50] Rasool Fakoor, Faisal Ladhak, Azade Nazi, and Manfred Huber. Using deep learning to enhance cancer diagnosis and classification, 2013.

[51] Francesca Finotello and Zlatko Trajanoski. New strategies for cancer immunotherapy: targeting regulatory t cells. *Genome medicine*, 9(1):1–3, 2017.

[52] Marco Fraga, Milly Yáñez, Macarena Sherman, Faryd Llerena, Mauricio Hernandez, Guillermo Nourdin, Joaquin Urrizola, César Rivera, Liliana Lamperti, María-Lorena

Nova, et al. Immunomodulation of t helper cells by tumor microenvironment in oral cancer is associated with ccr8 expression and rapid membrane vitamin d signaling pathway. *Frontiers in immunology*, 12:1668, 2021.

[53] John V Frangioni. New technologies for human cancer imaging. *Journal of clinical oncology*, 26(24):4012, 2008.

[54] Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36:193–202, 1980.

[55] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 262–270, Cambridge, MA, USA, 2015. MIT Press.

[56] Charles Gawad, Winston Koh, and Stephen R Quake. Single-cell genome sequencing: current state of the science. *Nature Reviews Genetics*, 17(3):175–188, 2016.

[57] Todd M Gierahn, Marc H Wadsworth II, Travis K Hughes, Bryan D Bryson, Andrew Butler, Rahul Satija, Sarah Fortune, J Christopher Love, and Alex K Shalek. Seq-well: portable, low-cost rna sequencing of single cells at high throughput. *Nature methods*, 14(4):395, 2017.

[58] Anthony TC Goh. Back-propagation neural networks for modeling complex systems. *Artificial Intelligence in Engineering*, 9(3):143–151, 1995.

[59] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[60] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[61] John C Gore, H Charles Manning, C Chad Quarles, Kevin W Waddell, and Thomas E Yankeelov. Magnetic resonance in the era of molecular imaging of cancer. *Magnetic resonance imaging*, 29(5):587–600, 2011.

[62] Gabriele Grasmann, Elisabeth Smolle, Horst Olschewski, and Katharina Leithner. Gluconeogenesis in cancer cells–repurposing of a starvation-induced metabolic pathway? *Biochimica et Biophysica Acta (BBA)-Reviews on Cancer*, 2019.

[63] Casey S Greene, Jie Tan, Matthew Ung, Jason H Moore, and Chao Cheng. Big data bioinformatics. *Journal of cellular physiology*, 229(12):1896–1900, 2014.

[64] Matthias Grell, Eleni Douni, Harald Wajant, Matthias Löhden, Matthias Clauss, Beate Maxeiner, Spiros Georgopoulos, Werner Lesslauer, George Kollias, Klaus Pfizenmaier, et al. The transmembrane form of tumor necrosis factor is the prime activating ligand of the 80 kda tumor necrosis factor receptor. *Cell*, 83(5):793–802, 1995.

[65] Jordan Guerguiev, Timothy P Lillicrap, and Blake A Richards. Towards deep learning with segregated dendrites. In *ELife*, volume 6, page e22901. eLife Sciences Publications Limited, 2017.

[66] Sin Yee Gun, Sharon Wei Ling Lee, Je Lin Sieow, and Siew Cheng Wong. Targeting immune cells for cancer therapy. *Redox biology*, page 101174, 2019.

[67] Kun Guo, Cheng Zhao, Bin Lang, Huiqin Wang, Hang Zheng, and Feng Zhang. Regulator of chromosome condensation 2 modulates cell cycle progression, tumorigenesis, and therapeutic resistance. *Frontiers in Molecular Biosciences*, page 467, 2021.

[68] Xinyi Guo, Yuanyuan Zhang, Liangtao Zheng, Chunhong Zheng, Jintao Song, Qiming Zhang, Boxi Kang, Zhouzerui Liu, Liang Jin, Rui Xing, et al. Global characterization of t cells in non-small-cell lung cancer by single-cell sequencing. *Nature medicine*, 24(7):978–985, 2018.

[69] Leena Halim, Marco Romano, Reuben H C McGregor, Isabel Leal Azevedo Corrêa, P Pavlidis, Nathali Grageda, Sec-Julie Hoong, Muhammed Yuksel, Wayel Jassem, Rosalind F. Hannen, M W S Ong, Olivia Mckinney, B Hayee, Sophia N Karagiannis, Nicholas Powell, R. Lechler, Estefania Andrea Nova-Lamperti, and Giovanna Lombardi. An atlas of human regulatory t helper-like cells reveals features of th2-like tregs that support a tumorigenic environment. *Cell reports*, 20(3):757–770, 2017.

[70] Jun Han and Claudio Moraga. The influence of the sigmoid function parameters on the speed of backpropagation learning. In *International Workshop on Artificial Neural Networks*, pages 195–201. Springer, 1995.

[71] Yixing Han, Shouguo Gao, Kathrin Muegge, Wei Zhang, and Bing Zhou. Advanced applications of rna sequencing and challenges. *Bioinformatics and biology insights*, 9:BBI–S28991, 2015.

[72] Ashraful Haque, Jessica A Engel, Sarah A. Teichmann, and Tapio Lönnberg. A practical guide to single-cell rna-sequencing for biomedical research and clinical applications. *Genome Medicine*, 9(1):1–12, 2017.

[73] Mohammad Haris, Santosh K Yadav, Arshi Rizwan, Anup Singh, Ena Wang, Hari Hariharan, Ravinder Reddy, and Francesco M Marincola. Molecular magnetic resonance imaging in cancer. *Journal of Translational Medicine*, 13(1):313, 2015.

[74] Henry Heberle, Gabriela Vaz Meirelles, Felipe R da Silva, Guilherme P Telles, and Rosane Minghim. Interactivenn: a web-based tool for the analysis of sets through venn diagrams. *BMC bioinformatics*, 16(1):1–7, 2015.

[75] Gervaise H Henry, Alicia Malewska, Diya B Joseph, Venkat S Malladi, Jeon Lee, Jose Torrealba, Ryan J Mauck, Jeffrey C Gahan, Ganesh V Raj, Claus G Roehrborn, et al. A cellular anatomy of the normal adult human prostate and prostatic urethra. *Cell reports*, 25(12):3530–3542, 2018.

[76] Lars Hertel, Huy Phan, and Alfred Mertins. Comparing time and frequency domain for audio event recognition using deep learning. *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 3407–3411, 2016.

[77] Geoffrey E. Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29:82–97, 2012.

[78] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[79] Bastian Hoesel and Johannes A Schmid. The complexity of nf-κb signaling in inflammation and cancer. *Molecular cancer*, 12(1):86, 2013.

[80] Chung-Chau Hon, Jay W Shin, Piero Carninci, and Michael JT Stubbington. The human cell atlas: Technical approaches and challenges. *Briefings in functional genomics*, 17(4):283–294, 2017.

[81] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[82] Xiaoming Huo, Xuelei Sherry Ni, and Andrew K Smith. A survey of manifold-based learning methods. *Recent advances in data mining of enterprise data*, pages 691–745, 2007.

[83] Byungjin Hwang, Ji Hyun Lee, and Duhee Bang. Single-cell rna sequencing technologies and bioinformatics pipelines. *Experimental & molecular medicine*, 50(8):1–14, 2018.

[84] Sherrif F Ibrahim and Ger van den Engh. Flow cytometry and cell sorting. *Advances in biochemical engineering/biotechnology*, 106:19–39, 2007.

[85] Andrea Iellem, Margherita Mariani, Rosmarie Lang, Helios Recalde, Paola Panina-Bordignon, Francesco Sinigaglia, and Daniele D'Ambrosio. Unique chemotactic response profile and specific expression of chemokine receptors ccr4 and ccr8 by

cd4+ cd25+ regulatory t cells. *Journal of Experimental Medicine*, 194(6):847–854, 2001.

[86] Navdeep Jaitly and Geoffrey E Hinton. Vocal tract length perturbation (vtlp) improves speech recognition, 2013.

[87] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.

[88] Katarzyna Janocha and Wojciech Czarnecki. On loss functions for deep neural networks in classification. *arXiv preprint arXiv:1702.05659*, 2017.

[89] Bijay Jassal, Lisa Matthews, Guilherme Viteri, Chuqiao Gong, Pascual Lorente, Antonio Fabregat, Konstantinos Sidiropoulos, Justin Cook, Marc Gillespie, Robin Haw, Fred Loney, Bruce May, Marija Milacic, Karen Rothfels, Cristoffer Sevilla, Veronica Shamovsky, Solomon Shorser, Thawfeek Varusai, Joel Weiser, Guanming Wu, Lincoln Stein, Henning Hermjakob, and Peter D'Eustachio. The reactome pathway knowledgebase. *Nucleic Acids Research*, 48(D1):D498–D503, 11 2019.

[90] Andrew L Ji, Adam J Rubin, Kim Thrane, Sizun Jiang, David L Reynolds, Robin M Meyers, Margaret G Guo, Benson M George, Annelie Mollbrink, Joseph Bergenstråhle, et al. Multimodal analysis of composition and spatial architecture in human squamous cell carcinoma. *Cell*, 182(2):497–514, 2020.

[91] Steve Jupe, Keith Ray, Corina Duenas Roca, Thawfeek Varusai, Veronica Shamovsky, Lincoln Stein, Peter D'Eustachio, and Henning Hermjakob. Interleukins and their signaling pathways in the reactome biological pathway database. *Journal of Allergy and Clinical Immunology*, 141(4):1411–1416, 2018.

[92] David R Kelley, Jasper Snoek, and John L Rinn. Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome research*, 26:990–999, 2016.

[93] H. J. Kelley. Gradient theory of optimal flight paths. *ARS Journal*, 30(10):947–954, 1960.

[94] Mahtab Keshvari, Mahdieh Nejadtaghi, Farnaz Hosseini-Beheshti, Ali Rastqar, and Niraj Patel. Exploring the role of circadian clock gene and association with cancer pathophysiology. *Chronobiology International*, 37(2):151–175, 2020.

[95] Fabian Kiessling, Jessica Bzyl, Stanley Fokong, Monica Siepmann, Georg Schmitz, and Moritz Palmowski. Targeted ultrasound imaging of cancer: an emerging technology on its way to clinics. *Current pharmaceutical design*, 18(15):2184–2199, 2012.

[96] Minje Kim and Paris Smaragdis. Adaptive denoising autoencoders: A fine-tuning scheme to learn from test mixtures. In Emmanuel Vincent, Arie Yeredor, Zbyněk

Koldovský, and Petr Tichavský, editors, *Latent Variable Analysis and Signal Separation*, pages 100–107, Cham, 2015. Springer International Publishing.

[97] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[98] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes, 2014.

[99] Diederik P. Kingma and Max Welling. An introduction to variational autoencoders. *CoRR*, abs/1906.02691, 2019.

[100] Vladimir Yu Kiselev, Tallulah S Andrews, and Martin Hemberg. Challenges in unsupervised clustering of single-cell rna-seq data. *Nature Reviews Genetics*, 20(5):273–282, 2019.

[101] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. In *Advances in neural information processing systems*, pages 971–980, 2017.

[102] Susanne Kohl, Britta Baumann, Thomas Rosenberg, Ulrich Kellner, Birgit Lorenz, Maria Vadala, Samuel G Jacobson, and Bernd Wissinger. Mutations in the cone photoreceptor g-protein α-subunit gene gnat2 in patients with achromatopsia. *The American Journal of Human Genetics*, 71(2):422–425, 2002.

[103] Kishore Konda, Roland Memisevic, and David Krueger. Zero-bias autoencoders and the benefits of co-adapting features. *arXiv preprint arXiv:1402.3337*, 2014.

[104] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60:84–90, 2012.

[105] Brahma V Kumar, Thomas J Connors, and Donna L Farber. Human t cell development, localization, and function throughout life. *Immunity*, 48(2):202–213, 2018.

[106] Kiran Kurmi and Marcia C Haigis. Nitrogen metabolism in cancer and immunity. *Trends in Cell Biology*, 2020.

[107] Pedro Larrañaga, Borja Calvo, Roberto Santana, Concha Bielza, Josu Galdiano, Iñaki Inza, José A. Lozano, Rubén Armañanzas, Guzmán Santafé, Aritz Pérez, and Victor Robles. Machine learning in bioinformatics. *Briefings in Bioinformatics*, 7(1):86–112, 2006.

[108] Pierre Laurent-Puig, Astrid Lievre, and Hélène Blons. Mutations and response to epidermal growth factor receptor inhibitors. *Clinical cancer research*, 15(4):1133–1139, 2009.

[109] Yann LeCun, Yoshua Bengio, and Geoffrey E. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[110] Hae Lim Lee, Jeong Won Jang, Sung Won Lee, Sun Hong Yoo, Jung Hyun Kwon, Soon Woo Nam, Si Hyun Bae, Jong Young Choi, Nam Ik Han, and Seung Kew Yoon. Inflammatory cytokines and change of th1/th2 balance as prognostic indicators for hepatocellular carcinoma in patients treated with transarterial chemoembolization. *Scientific reports*, 9(1):3260, 2019.

[111] June-Goo Lee, Sanghoon Jun, Young-Won Cho, Hyunna Lee, Guk Bae Kim, Joon Beom Seo, and Namkug Kim. Deep learning in medical imaging: general overview. *Korean journal of radiology*, 18(4):570–584, 2017.

[112] Sylvia Lee and Kim Margolin. Cytokines in cancer immunotherapy. *Cancers*, 3(4):3856–3893, 2011.

[113] Connie L Lerea, Ann H Bunt-Milam, and James B Hurley. α transducin is present in blue-, green-, and red-sensitive cone photoreceptors in the human retina. *Neuron*, 3(3):367–376, 1989.

[114] Hanjie Li, Anne M van der Leun, Ido Yofe, Yaniv Lubling, Dikla Gelbard-Solodkin, Alexander CJ van Akkooi, Marlous van den Braber, Elisa A Rozeman, John BAG Haanen, Christian U Blank, et al. Dysfunctional cd8 t cells form a proliferative, dynamically regulated compartment within human melanoma. *Cell*, 176(4):775–789, 2019.

[115] Haoxiang Li, Zhe Lin, Xiaohui Shen, Jonathan Brandt, and Gang Hua. A convolutional neural network cascade for face detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5325–5334, 2015.

[116] Lantian Li, Weizhi Xu, and Hui Yu. Character-level neural network model based on nadam optimization and its application in clinical concept extraction. *Neurocomputing*, 414:182–190, 2020.

[117] Mu Li, Tong Zhang, Yuqiang Chen, and Alexander J Smola. Efficient mini-batch training for stochastic optimization. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 661–670. ACM, 2014.

[118] Peijie Lin, Michael Troup, and Joshua WK Ho. Cidr: Ultrafast and accurate clustering through imputation for single-cell rna-seq data. *Genome biology*, 18(1):59, 2017.

[119] Bin Liu, Randy Yang, Kelly A Wong, Crescent Getman, Natalie Stein, Michael A Teitell, Genhong Cheng, Hong Wu, and Ke Shuai. Negative regulation of nf-κb signaling by pias1. *Molecular and cellular biology*, 25(3):1113–1123, 2005.

[120] Qian Liu and Pingzhao Hu. Association analysis of deep genomic features extracted by denoising autoencoders in breast cancer. *Cancers*, 11(4):494, 2019.

[121] Yang Liu and Duolin Wang. Application of deep learning in genomic selection. *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 2280–2280, 2017.

[122] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[123] Rohan Lowe, Neil Shirley, Mark Bleackley, Stephen Dolan, and Thomas Shafee. Transcriptomics technologies. *PLoS computational biology*, 13(5):e1005457, 2017.

[124] Xugang Lu, Yu Tsao, Shigeki Matsuda, and Chiori Hori. Speech enhancement based on deep denoising autoencoder. In *Interspeech*, pages 436–440. ISCA, 2013.

[125] Rishi Vishal Luckheeram, Rui Zhou, Asha Devi Verma, and Bing Xia. Cd4+t cells: Differentiation and functions. *Clinical and Developmental Immunology*, 2012:925135, 2012.

[126] Jianyuan Luo, Anatoly Y Nikolaev, Shin-ichiro Imai, Delin Chen, Fei Su, Ariel Shiloh, Leonard Guarente, and Wei Gu. Negative control of p53 by sir2α promotes cell survival under stress. *Cell*, 107(2):137–148, 2001.

[127] Xiangjian Luo, Can Cheng, Zheqiong Tan, Namei Li, Min Tang, Lifang Yang, and Ya Cao. Emerging roles of lipid metabolism in cancer metastasis. *Molecular cancer*, 16(1):76, 2017.

[128] Agnes Lydia and Sagayaraj Francis. Adagrad—an optimizer for stochastic gradient descent. *International Journal of Information and Computing Science*, 6(5), 2019.

[129] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the 30th International Conference on Machine Learning*, volume 30, page 3, 2013.

[130] Andrzej Maćkiewicz and Waldemar Ratajczak. Principal components analysis (pca). *Computers & Geosciences*, 19(3):303–342, 1993.

[131] Polina Mamoshina, Armando Vieira, Evgeny Putin, and Alex Zhavoronkov. Applications of deep learning in biomedicine. *Molecular pharmaceutics*, 13(5):1445–1454, 2016.

[132] Elaine R Mardis. The impact of next-generation sequencing technology on genetics. *Trends in genetics*, 24(3):133–141, 2008.

[133] Vivien Marx. Biology: The big challenges of big data, 2013.

[134] Selma Masri and Paolo Sassone-Corsi. The emerging link between cancer, metabolism, and circadian rhythms. *Nature medicine*, 24(12):1795, 2018.

[135] Robin Mathew, Vassiliki Karantza-Wadsworth, and Eileen White. Role of autophagy in cancer. *Nature Reviews Cancer*, 7(12):961–967, 2007.

[136] Davis J McCarthy, Kieran R Campbell, Aaron TL Lun, and Quin F Wills. Scater: pre-processing, quality control, normalization and visualization of single-cell rna-seq data in r. *Bioinformatics*, 33(8):1179–1186, 2017.

[137] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[138] Laura M McLane, Mohamed S Abdel-Hakeem, and E John Wherry. Cd8 t cell exhaustion during chronic viral infection and cancer. *Annual review of immunology*, 37:457–495, 2019.

[139] Nijat Mehdiyev, Joerg Evermann, and Peter Fettke. A multi-stage deep learning approach for business process event prediction. *2017 IEEE 19th Conference on Business Informatics (CBI)*, 01:119–128, 2017.

[140] David G Menter, Stephanie C Tucker, Scott Kopetz, Anil K Sood, John D Crissman, and Kenneth V Honn. Platelets and cancer: a casual or causal relationship: revisited. *Cancer and Metastasis Reviews*, 33(1):231–269, 2014.

[141] Michael L Metzker. Sequencing technologies—the next generation. *Nature reviews genetics*, 11(1):31, 2010.

[142] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Eleventh Annual Conference of the International Speech Communication Association*, pages 1045–1048. ISCA, 2010.

[143] Seonwoo Min, Byunghan Lee, and Sungroh Yoon. Deep learning in bioinformatics. *Briefings in bioinformatics*, 18(5):851–869, 2017.

[144] Lars Muhl, Guillem Genové, Stefanos Leptidis, Jianping Liu, Liqun He, Giuseppe Mocci, Ying Sun, Sonja Gustafsson, Byambajav Buyandelger, Indira V Chivukula, et al. Single-cell analysis uncovers fibroblast heterogeneity and criteria for fibroblast and mural cell identification and discrimination. *Nature communications*, 11(1):1–18, 2020.

[145] Mahesh Chandra Mukkamala and Matthias Hein. Variants of rmsprop and adagrad with logarithmic regret bounds. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 2545–2553. PMLR, 2017.

[146] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[147] Anthony J Myles, Robert N Feudale, Yang Liu, Nathaniel A Woody, and Steven D Brown. An introduction to decision tree modeling. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 18(6):275–285, 2004.

[148] Jawad Nagi, Frederick Ducatelle, Gianni A Di Caro, Dan Cireşan, Ueli Meier, Alessandro Giusti, Farrukh Nagi, Jürgen Schmidhuber, and Luca Maria Gambardella. Max-pooling convolutional neural networks for vision-based hand gesture recognition. In *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pages 342–347. IEEE, 2011.

[149] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[150] Yurii Nesterov. A method for unconstrained convex minimization problem with the rate of convergence o (1/kˆ 2). In *Doklady AN USSR*, volume 269, pages 543–547, 1983.

[151] Andrew Y Ng. Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In *Proceedings of the Twenty-first International Conference on Machine Learning*, page 78, New York, NY, USA, 2004. ACM.

[152] Paula Nieto, Marc Elosua-Bayes, Juan L Trincado, Domenica Marchese, Ramon Massoni-Badosa, Maria Salvany, Ana Henriques, Juan Nieto, Sergio Aguilar-Fernández, Elisabetta Mereu, et al. A single-cell tumor immune atlas for precision oncology. *Genome research*, 31(10):1913–1926, 2021.

[153] Andreas Oberholzer, Caroline Oberholzer, and Lyle L Moldawer. Cytokine signaling-regulation of the immune response in normal and critically ill states. *Critical care medicine*, 28(4):N3–N12, 2000.

[154] Natalia Ochocka, Pawel Segit, Kacper Adam Walentynowicz, Kamil Wojnicki, Salwador Cyranowski, Julian Swatler, Jakub Mieczkowski, and Bozena Kaminska. Single-cell rna sequencing reveals functional heterogeneity of glioma-associated brain macrophages. *Nature communications*, 12(1):1–14, 2021.

[155] World Health Organization et al. *Guide to cancer early diagnosis*. World Health Organization, 2017.

[156] Valeska Ormazabal, Estefanía Nova-Lampeti, Daniela Rojas, Felipe A Zúñiga, Carlos Escudero, Paola Lagos, Alexa Moreno, Yanara Pavez, Camila Reyes, Milly Yáñez, et al. Secretome from human mesenchymal stem cells-derived endothelial cells promotes wound healing in a type-2 diabetes mouse model. *International journal of molecular sciences*, 23(2):941, 2022.

[157] Dmitrij Ostroumov, Nora Fekete-Drimusz, Michael Saborowski, Florian Kühnel, and Norman Woller. Cd4 and cd8 t lymphocyte interplay in controlling tumor growth. *Cellular and molecular life sciences*, 75(4):689–713, 2018.

[158] Patrick Alexander Ott, Frank Stephen Hodi, and Caroline Robert. Ctla-4 and pd-1/pd-l1 blockade: new immunotherapeutic modalities with durable clinical benefit in melanoma patients. *Clinical Cancer Research*, 19(19):5300–5309, 2013.

[159] Zeev Pancer and Max D Cooper. The evolution of adaptive immunity. *Annu. Rev. Immunol.*, 24:497–518, 2006.

[160] Drew Mark Pardoll. The blockade of immune checkpoints in cancer immunotherapy. *Nature Reviews Cancer*, 12(4):252–264, 2012.

[161] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[162] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jacob VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[163] Joffrey Pelletier, George Thomas, and Siniša Volarević. Ribosome biogenesis in cancer: new players and therapeutic avenues. *Nature Reviews Cancer*, 18(1):51–63, 2018.

[164] Junya Peng, Bao-Fa Sun, Chuan-Yuan Chen, Jia-Yi Zhou, Yu-Sheng Chen, Hao Chen, Lulu Liu, Dan Huang, Jialin Jiang, Guan-Shen Cui, et al. Single-cell rna-seq highlights intra-tumoral heterogeneity and malignant progression in pancreatic ductal adenocarcinoma. *Cell research*, 29(9):725–738, 2019.

[165] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *ArXiv*, 2017.

[166] Emma Pierson and Christopher Yau. Zifa: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome biology*, 16(1):241, 2015.

[167] Ruben Pio, Leticia Corrales, and John D Lambris. The role of complement in tumor growth. *Tumor microenvironment and cellular stress*, pages 229–262, 2014.

[168] Sidharth V Puram, Itay Tirosh, Anuraag S Parikh, Anoop P Patel, Keren Yizhak, Shawn Gillespie, Christopher Rodman, Christina L Luo, Edmund A Mroz, Kevin S Emerick, et al. Single-cell transcriptomic analysis of primary and metastatic tumor ecosystems in head and neck cancer. *Cell*, 171(7):1611–1624, 2017.

[169] Ning Qian. On the momentum term in gradient descent learning algorithms. *Neural networks*, 12(1):145–151, 1999.

[170] Gabriel A Rabinovich, Dmitry Gabrilovich, and Eduardo M Sotomayor. Immunosuppressive strategies that are mediated by tumor cells. *Annu. Rev. Immunol.*, 25:267–296, 2007.

[171] Felix Radford, Sanjay Tyagi, Maria Laura Gennaro, Richard Pine, and Yuri Bushkin. Flow cytometric characterization of antigen-specific t cells based on rna and its advantages in detecting infections and immunological disorders. *Critical reviews in immunology*, 36(5):359–378, 2016.

[172] Anand Ramachandran, Huiren Li, Eric Klee, Steven S. Lumetta, and Deming Chen. Deep learning for better variant calling for cancer diagnosis and treatment. In *2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 16–21, 2018.

[173] Siegel Rebecca, Miller Kimberly, and Jemal Ahmedin. Cancer statistics, 2017. *CA: A Cancer Journal for Clinicians*, 67(1):7–30, 2017.

[174] Fulvio Reggiori and Christian Ungermann. Autophagosome maturation and fusion. *Journal of molecular biology*, 429(4):486–496, 2017.

[175] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386–408, 1958.

[176] Raghd Rostom, Valentine Svensson, Sarah A Teichmann, and Gozde Kar. Computational approaches for interpreting scrna-seq data. *FEBS letters*, 591(15):2213–2225, 2017.

[177] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[178] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

[179] Saskia J. A. M. Santegoets, Eveline M. Dijkgraaf, Alessandra Battaglia, Philipp Beckhove, Cedrik M. Britten, Awen Gallimore, Andrew J. Godkin, Cécile Gouttefangeas, Tanja D. de Gruijl, Hans J P M Koenen, Alexander Scheffold, Ethan M Shevach, Janet S Staats, Kjetil Taskén, Theresa L Whiteside, Judith Kroep, Marij J. P. Welters, and Sjoerd H van der Burg. Monitoring regulatory t cells in clinical samples: consensus on an essential marker set and gating strategy for regulatory t cell analysis by flow cytometry. *Cancer Immunology, Immunotherapy*, 64(10):1271–1286, 2015.

[180] Tetsuro Sasada, Koichi Azuma, Junya Ohtake, and Yuki Fujimoto. Immune responses to epidermal growth factor receptor (egfr) and their application for cancer treatment. *Frontiers in pharmacology*, 7:405, 2016.

[181] Rahul Satija, Jeffrey A Farrell, David Gennert, Alexander F Schier, and Aviv Regev. Spatial reconstruction of single-cell gene expression data. *Nature biotechnology*, 33(5):495–502, 2015.

[182] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85 – 117, 2015.

[183] Bertil Schmidt and Andreas Hildebrandt. Next-generation sequencing: big data meets high performance computing. *Drug discovery today*, 22(4):712–717, 2017.

[184] Stephan C Schuster. Next-generation sequencing transforms today's biology. *Nature methods*, 5(1):16, 2007.

[185] Björn Schwanhäusser, Dorothea Busse, Na Li, Gunnar Dittmar, Johannes Schuchhardt, Jana Wolf, Wei Chen, and Matthias Selbach. Global quantification of mammalian gene expression control. *Nature*, 473(7347):337, 2011.

[186] Polina Schwartsburd. Cancer-induced reprogramming of host glucose metabolism:«vicious cycle» supporting cancer progression. *Frontiers in oncology*, 9:218, 2019.

[187] Sarah A Scott, Thomas P Mathews, Pavlina T Ivanova, Craig W Lindsley, and H Alex Brown. Chemical modulation of glycerolipid signaling and metabolic pathways. *Biochimica et Biophysica Acta (BBA)-Molecular and Cell Biology of Lipids*, 1841(8):1060–1084, 2014.

[188] Paul T. Shannon, Andrew Markiel, Owen Ozier, Nitin S. Baliga, Jonathan T. Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome research*, 13(11):2498–504, 2003.

[189] Jay Shendure and Hanlee Ji. Next-generation dna sequencing. *Nature biotechnology*, 26(10):1135, 2008.

[190] Jian-Hong Shi and Shao-Cong Sun. Tumor necrosis factor receptor-associated factor regulation of nuclear factor κb and mitogen-activated protein kinase pathways. *Frontiers in immunology*, 9:1849, 2018.

[191] Adam J Shuhendler, Deju Ye, Kimberly D Brewer, Magdalena Bazalova-Carter, Kyung-Hyun Lee, Paul Kempen, K Dane Wittrup, Edward E Graves, Brian Rutt, and Jianghong Rao. Molecular magnetic resonance imaging of tumor response to therapy. *Scientific reports*, 5:14759, 2015.

[192] Korsuk Sirinukunwattana, Shan e Ahmed Raza, Yee-Wah Tsang, David R. J. Snead, Ian A. Cree, and Nasir M. Rajpoot. Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. *IEEE Transactions on Medical Imaging*, 35:1196–1206, 2016.

[193] C. Spearman. General intelligence objectively determined and measured. *The American Journal of Psychology*, 15(2):201–292, 1904.

[194] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[195] Siddharth Srivastava, Sumit Soman, Astha Rai, and Praveen K. Srivastava. Deep learning for health informatics: Recent trends and future directions. *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1665–1670, 2017.

[196] Tim Stuart, Andrew Butler, Paul Hoffman, Christoph Hafemeister, Efthymia Papalexi, William M Mauck III, Yuhan Hao, Marlon Stoeckius, Peter Smibert, and Rahul Satija. Comprehensive integration of single-cell data. *Cell*, 177(7):1888–1902, 2019.

[197] Susmit Suvas, Ahmet Kursat Azkur, Bum Seok Kim, Uday Kumaraguru, and Barry T. Rouse. Cd4+ cd25+ regulatory t cells control the severity of viral immunoinflammatory lesions. *The Journal of Immunology*, 172(7):4123–4132, 2004.

[198] Valentine Svensson, Kedar Nath Natarajan, Lam-Ha Ly, Ricardo J. Miragaia, Charlotte Labalette, Iain Macaulay, Ana Cvejic, and Sarah A. Teichmann. Power analysis of single cell rna-sequencing experiments. *Nature Methods*, 14(4):381, 2017.

[199] Peter A Szabo, Hanna Mendes Levitin, Michelle Miron, Mark E Snyder, Takashi Senda, Jinzhou Yuan, Yim Ling Cheng, Erin C Bush, Pranay Dogra, Puspa Thapa, et al. A single-cell reference map for human blood and tissue t cell activation reveals functional states in health and disease. *bioRxiv*, page 555557, 2019.

[200] Sona Taheri and Musa Mammadov. Learning the naive bayes classifier with optimization models. *International Journal of Applied Mathematics and Computer Science*, 23(4), 2013.

[201] Jie Tan, Matthew Ung, Chao Cheng, and Casey S Greene. Unsupervised feature construction and knowledge extraction from genome-wide assays of breast cancer with denoising autoencoders. In *Pacific Symposium on Biocomputing Co-Chairs*, pages 132–143. World Scientific, 2014.

[202] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to data mining*. Pearson Education India, 2016.

[203] Fuchou Tang, Catalin Barbacioru, Yangzhou Wang, Ellen Nordman, Clarence Lee, N Xu, Xiaohui Wang, John P. Bodeau, Brian B. Tuch, Asim Siddiqui, Kai Qin Lao, and M Azim Surani. mrna-seq whole-transcriptome analysis of a single cell. *Nature Methods*, 6(5):377–382, 2009.

[204] Sarang Tartey and Osamu Takeuchi. Pathogen recognition and toll-like receptor targeted therapeutics in innate immune cells. *International Reviews of Immunology*, 36(2):57–73, 2017.

[205] Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[206] Itay Tirosh, Benjamin Izar, Sanjay M Prakadan, Marc H Wadsworth, Daniel Treacy, John J Trombetta, Asaf Rotem, Christopher Rodman, Christine Lian, George Murphy, et al. Dissecting the multicellular ecosystem of metastatic melanoma by single-cell rna-seq. *Science*, 352(6282):189–196, 2016.

[207] Alexander J. Titus, Owen M. Wilkins, Carly A. Bobak, and Brock C. Christensen. An unsupervised deep learning framework with variational autoencoders for genome-wide dna methylation analysis and biologic feature extraction applied to breast cancer. *bioRxiv*, 2018.

[208] Warren S Torgerson. Multidimensional scaling: I. theory and method. *Psychometrika*, 17(4):401–419, 1952.

[209] Rashmi Tripathi, Pawan Sharma, Pavan Chakraborty, and Pritish Kumar Varadwaj. Next-generation sequencing revolution through big data analytics. *Frontiers in Life Science*, 9(2):119–149, 2016.

[210] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[211] Laurens Van Der Maaten, Eric Postma, Jaap Van den Herik, et al. Dimensionality reduction: a comparative. *J Mach Learn Res*, 10(66-71):13, 2009.

[212] Elodie Villa, Eunus S Ali, Umakant Sahu, and Issam Ben-Sahra. Cancer cells tune the signaling pathways to empower de novo synthesis of nucleotides. *Cancers*, 11(5):688, 2019.

[213] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 1096–1103, New York, NY, USA, 2008. ACM.

[214] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11:3371–3408, dec 2010.

[215] Duolin Wang and Dongpeng Liu. Musitedeep: A deep-learning framework for protein post-translational modification site prediction. *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 2327–2327, 2017.

[216] Yijun Wang, Pengyu Zhou, and Wenya Zhong. An optimization strategy based on hybrid algorithm of adam and sgd. In *MATEC Web of Conferences*, volume 232, page 03007. EDP Sciences, 2018.

[217] Zhanyu Wang and Chenfang Dong. Gluconeogenesis in cancer: Function and regulation of pepck, fbpase, and g6pase. *Trends in cancer*, 5(1):30–45, 2019.

[218] Gregory P Way and Casey S Greene. Extracting a biologically relevant latent space from cancer transcriptomes with variational autoencoders. *Pacific Symposium on Biocomputing*, 23:80–91, 2018.

[219] Rikiya Yamashita, Mizuho Nishio, Richard Kinh Gian Do, and Kaori Togashi. Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, 9(4):611–629, Aug 2018.

[220] Yuan Yao, Lorenzo Rosasco, and Andrea Caponnetto. On early stopping in gradient descent learning. *Constructive Approximation*, 26(2):289–315, 2007.

[221] Shun H Yip, Pak Chung Sham, and Junwen Wang. Evaluation of tools for highly variable gene discovery from single-cell rna-seq data. *Briefings in bioinformatics*, 20(4):1583–1589, 2019.

[222] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing. *IEEE Computational intelligenCe magazine*, 13(3):55–75, 2018.

[223] Yuchen Yuan, Yi Shi, Changyang Li, Jinman Kim, Tom Weidong Cai, Zeguang Han, and David Dagan Feng. Deepgene: an advanced cancer type classifier based on deep learning and somatic point mutations. *BMC bioinformatics*, 17(17):476, 2016.

[224] Matthew D. Zeiler. ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701, 2012.

[225] Matthew D. Zeiler and Rob Fergus. Stochastic pooling for regularization of deep convolutional neural networks. *arXiv preprint arXiv:1301.3557*, 2013.

[226] Kristen P Zeligs, Monica K Neuman, and Christina M Annunziata. Molecular pathways: the balance between cancer and the immune system challenges the therapeutic specificity of targeting nuclear factor-κb signaling for cancer treatment. *Clinical Cancer Research*, 22(17):4302–4308, 2016.

[227] Jun-Ming Zhang and Jianxiong An. Cytokines, inflammation and pain. *International anesthesiology clinics*, 45(2):27, 2007.

[228] Lei Zhang, Xin Yu, Liangtao Zheng, Yuanyuan Zhang, Yansen Li, Qiao Fang, Ranran Gao, Boxi Kang, Qiming Zhang, Julie Y Huang, et al. Lineage tracking reveals dynamic relationships of t cells in colorectal cancer. *Nature*, 564(7735):268, 2018.

[229] Shuqin Zhang. Comparisons of gene coexpression network modules in breast cancer and ovarian cancer. *BMC systems biology*, 12(1):8, 2018.

[230] Marta Łuksza, Nadeem Riaz, Vladimir Makarov, Vinod P. Balachandran, Matthew D Hellmann, Alexander Solovyov, Naiyer A Rizvi, Taha Merghoub, Arnold J. Levine, Timothy A. Chan, Jedd D Wolchok, and Benjamin D. Greenbaum. A neoantigen fitness model predicts tumour response to checkpoint blockade immunotherapy. *Nature*, 551:517–520, 2017.

Appendix

Table 6.1: Gene markers for T cells

ACKR2	ACKR3	ACKR4	ADA	ADAR	ADIPOQ	AICDA	AIMP1	APCS
BCL2	BCL2L1	BCL6	BLNK	BMP2	BMP4	BMP6	BMP7	BST2
BTK	BTLA	C3	C3AR1	C5	C5AR1	CASP1	CASP8	CAV1
CBLB	CCL1	CCL11	CCL13	CCL14	CCL15	CCL16	CCL17	CCL18
CCL19	CCL2	CCL20	CCL21	CCL22	CCL23	CCL24	CCL25	CCL26
CCL27	CCL28	CCL3	CCL4	CCL5	CCL7	CCL8	CCR1	CCR10
CCR2	CCR3	CCR4	CCR5	CCR6	CCR7	CCR8	CCR9	CCRL2
CD14	CD180	CD1A	CD1B	CD1C	CD1D	CD2	CD209	CD27
CD274	CD276	CD28	CD3D	CD3E	CD3G	CD4	CD40	CD40LG
CD44	CD47	CD5	CD7	CD70	CD74	CD80	CD81	CD86
CD8A	CDC42	CDK2	CDKN1A	CDKN1B	CEBPA	CEBPB	CHUK	CIITA
CKLF	CLEC4C	CLEC4E	CLEC7A	CMKLR1	CNTF	CRLF2	CRP	CSF1
CSF1R	CSF2	CSF2RA	CSF3	CSF3R	CTLA4	CX3CL1	CX3CR1	CXCL1
CXCL10	CXCL11	CXCL12	CXCL13	CXCL14	CXCL16	CXCL2	CXCL3	CXCL5
CXCL6	CXCL8	CXCL9	CXCR1	CXCR2	CXCR3	CXCR4	CXCR5	CXCR6
DDX58	DGKZ	DPP4	EBI3	EGF	EGFR	EGR1	EGR2	EGR3
EIF2AK2	ELK1	EOMES	ERBB2	F3	FADD	FAS	FASLG	FCER1A
FCER2	FCGR1A	FIGF	FLT3	FLT3LG	FOS	FOSL1	FOXP1	FOXP3
FPR1	GATA3	GBP1	GFI1	GPI	GZMA	GZMB	HAVCR2	HDAC9
HIF1A	HLA-A	HLA-B	HLA-C	HLA-DMA	HLA-DPA1	HLA-E	HLA-G	HMGB1
HSPD1	ICAM1	ICOS	ID2	IDO1	IFI16	IFI27	IFI30	IFI44
IFI44L	IFI6	IFIH1	IFIT1	IFIT2	IFIT3	IFITM1	IFITM2	IFITM3
IFNA1	IFNA14	IFNA16	IFNA2	IFNA21	IFNA4	IFNA5	IFNA6	IFNA7
IFNA8	IFNAR1	IFNAR2	IFNB1	IFNE	IFNG	IFNGR1	IFNGR2	IFNK
IFNL1	IFNLR1	IFNW1	IGF1	IGSF6	IKBKB	IL10	IL10RA	IL10RB
IL11	IL11RA	IL12A	IL12B	IL12RB1	IL12RB2	IL13	IL13RA1	IL15
IL16	IL17A	IL17B	IL17C	IL17F	IL17RA	IL17RB	IL17RE	IL18
IL18R1	IL18RAP	IL19	IL1A	IL1B	IL1R1	IL1R2	IL1RAP	IL1RL1
IL1RN	IL2	IL20	IL20RB	IL21	IL21R	IL22	IL22RA2	IL23A
IL23R	IL24	IL25	IL27	IL27RA	IL2RA	IL2RB	IL2RG	IL3
IL31	IL31RA	IL33	IL3RA	IL4	IL4R	IL5	IL5RA	IL6
IL6R	IL7	IL7R	IL9	IL9R	INHA	INHBA	IRAK1	IRAK2
IRAK4	IRF1	IRF2	IRF2BP1	IRF3	IRF4	IRF5	IRF6	IRF7
IRF8	IRF9	IRGM	ISG15	ISG20	ITCH	ITGA1	ITGAM	ITGB2
JAK1	JAK2	JAK3	JUN	KITLG	KNG1	LAG3	LAT	LCK
LEP	LEPR	LGALS3	LIF	LRP1	LTA	LTB	LY86	LY96
LYN	LYZ	MAF	MAP2K3	MAP3K1	MAP3K7	MAPK1	MAPK8	MBL2
MET	MICA	MICB	MIF	MMP3	MMP9	MPL	MPO	MS4A1
MSTN	MX1	MX2	MYC	MYD88	NCK1	NFATC1	NFATC2	NFATC3
NFKB1	NFKB2	NFKBIA	NFRKB	NLRP3	NMI	NOD1	NOD2	NODAL
NOS2	NOTCH1	NR2C2	NR3C1	NR4A1	NR4A3	OAS1	OAS2	OSM
PDCD1	PELI1	PML	POU2F2	PPARA	PPARG	PPBP	PRF1	PRKCG
PRKCZ	PRKRA	PSME1	PSME2	PTGDR2	PTGER2	PTGS2	PTPRC	RAC1
RAG1	REL	RELA	RELB	RIPK2	RNF128	RORA	RORC	RUNX1
RUNX3	S1PR1	SELE	SELL	SFTPD	SH2D1A	SIGIRR	SLC11A1	SLIT2
SOCS1	SOCS3	SOCS5	SPP1	STAT1	STAT2	STAT3	STAT4	STAT5A
STAT6	SYK	TAP1	TAP2	TAPBP	TBK1	TBX21	TGFA	TGFB1
TGFB2	TGFB3	THBS1	THPO	TICAM1	TICAM2	TIMP1	TIRAP	TLR1
TLR10	TLR2	TLR3	TLR4	TLR5	TLR6	TLR7	TLR8	TLR9
TMEM173	TNF	TNFRSF10A	TNFRSF11B	TNFRSF14	TNFRSF18	TNFRSF1A	TNFRSF4	TNFRSF8
TNFRSF9	TNFSF10	TNFSF11	TNFSF12	TNFSF13	TNFSF13B	TNFSF14	TNFSF4	TNFSF8
TOLLIP	TP53	TP53INP1	TRAF3	TRAF6	TXLNA	TYK2	TYMP	UBE2N
UTS2	VAV1	VCAM1	VEGFA	XCL1	XCR1	ZBTB7B		

Table 6.2: Genes highly and lowly weighted in the different principal components (PCs).

PC ID	Positive genes	Negative genes
PC-1	RPL7, EEF1A1, RPL18A, RPL9, SLC25A6, KCNQ1OT1, AXL, LPP, RPS3A, TOR1AIP2, RPS7, RPL36A, FAM114A1, RPL23A, RPL38, RPS28, CTTN, RPL41, CST3, CD99, GSN, SRSF10, RPL39, UGT8, TSC22D1, ADAMTS4, RPL23, TCF4, PNPO, MORF4L1	TMSB4X, MALAT1, FTH1, CD74, RPLP1, RPL8, TMSB10, SLMO2-ATP5E, RPS4X, RPL19, RPL11, RPL3, RPS14, TPT1, RPL7A, FTL, CYBA, RPL13A, CLEC2B, IFNG, PNRC1, RPS19, RPS3, S100A10, BIRC3, RGS10, FGFBP2, PTTG1, CCL3L1, AQP3
PC-2	CALD1, SPARC, C1R, SERPING1, MAP1B, CLU, FSTL1, CTTN, PMP22, FN1, DST, PLS3, CD9, NFIA, CD63, CCDC80, CD59, IGFBP7, MALAT1, DPYSL2, CLIC4, COL6A1, COL6A2, COL3A1, CFH, A2M, CALU, TSC22D1, MYL9, THBS1	ZNF850, LYZ, MPL, PRRG4, CCL22, PSTPIP2, UGT8, ZNF665, PIGX, RPS29, PTK6, PTAFR, RPL28, MAPK13, CCDC144B, BNIPL, EXPH5, PNPO, IL10, RPS18, CSF2RA, MCTS1, SNHG7, RPS15A, PIGR, SPRED1, CLSPN, TINCR, CFLAR, OCLN
PC-3	TMSB10, RPS25, S100A6, TMSB4X, RPS13, RPL11, RPL10A, RPS14, IGFBP7, RPL7A, MIF, RPLP1, COL6A2, FAU, RPS4X, RPS16, VIM, RPL19, RPS12, MYL9, GSN, COL3A1, TPT1, RPL26, MFGE8, RPL24, AEBP1, TPM2, HSPB1, RPL34	SCD, NEAT1, TF, DHCR24, ITGB8, PTPRF, EIF3CL, DST, CLU, APOE, C3, EEF1A1, SOX9, CXADR, MALAT1, NUDT3, DSP, CP, SERPINA1, MAP2, PCDH9, TOP2A, PRRC2C, CD24, CENPF, BROX, GATM, GAN, STMN1, ATF5
PC-4	CCNB2, TOP2A, CDK1, ENO1, STMN1, ZWINT, TPX2, DTL, UBE2C, NUSAP1, CCNB1, TK1, FANCI, NME1, BUB1B, SMC2, CDCA8, KIF23, BUB1, SRP9, SUB1, PSAT1, DBI, RPS3, RPL8, TUBA1B, CDCA3, NCAPG2, RPL19, CCT5	C4A, C3, SERPING1, GPX3, TF, APOE, ADAMTS4, C1R, MYL9, AXL, CFH, CRISPLD2, IFITM3, ADAMTS1, MYLK, SYNPO2, CALD1, TAGLN, TPM2, COL18A1, IGFBP7, AEBP1, SERPINF1, COL6A2, ATF5, NEAT1, COL3A1, THBS1, A2M, RASD1
PC-5	RPLP2, RPS12, RPL32, RPS11, RPL30, RPL11, RPL13A, RPS27A, RPL34, RPL37, RPS23, RPS14, RPS19, RPL35A, RPL13, RPS13, RPS8, RPL19, RPL18, RPS10, RPLP1, RPL26, RPS5, RPL14, RPL31, RPS16, RPL27, RPL35, RPS20, RPL12	HLA-DPA1, CTSC, HLA-DRA, CD74, CCL4L1, IFI6, ENTPD1, HLA-DPB1, HLA-DMA, LYZ, CCL3, PKM, CAPG, PTTG1, PTAFR, FKBP1A, NAMPT, TPM4, CD68, CSF2RA, TPMT, STX11, LPCAT2, S1PR2, CD83, CDKN1A, ITGAX, IFNG, EPSTI1, ATP1B3