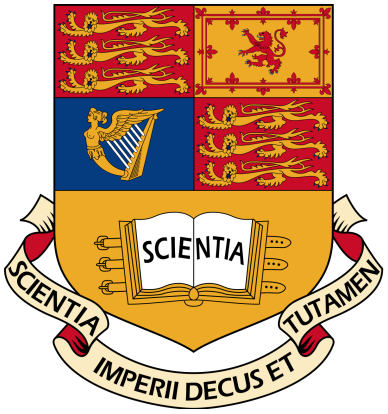


Imperial College London

Department of Computing

BEng Mathematics and Computer Science
Individual Project

Data Science for DLBCL Stratification

Author:
Avish Vijayaraghavan

Supervisor:
Dr. Elsa Angelini

Second Marker:
Dr. Viktoriia Sharmanska

June 17, 2020

Abstract

Diffuse large B-cell lymphoma (DLBCL), the most common type of non-Hodgkin's lymphoma, is a heterogeneous disease which means that it can manifest differently across patients. Extensive research has looked into defining the genetic subtypes of this cancer so that patients with these subtypes can be given more personalised treatment. The influx of high-throughput technologies has resulted in vast quantities of biological data, allowing us to better categorise these cancer subtypes and corresponding patient subgroups, known respectively as biomarker discovery and patient stratification. The application of machine learning methods to this data has shown enormous promise. This project looks at exploratory data analysis, guided by machine learning, to stratify a cohort of DLBCL patients. We survey the main papers in DLBCL stratification to derive our methodology and present a novel stratification based on the activity of a biological pathway called NF-κB and a related gene called GADD45B. We also investigate the impact of tumor purity on the data and determine that low purity samples often produce an inverse effect to the true signals of gene expression data.

We define all necessary biological terms but readers can refer to our biological glossary in Appendix A if required.

Acknowledgements

I would like to thank Richard and Natalie from BRC Genomics Facility for helping me get acquainted with bioinformatics and providing the datasets for the project.

To Guido and Daria at the Immunology and Inflammation department, thank you for keeping me focussed on the core task of GADD45B stratification and being patient with my understanding (or lack thereof!) of biology.

And most importantly, thank you to my supervisor Elsa for helping guide my project and keep me on track. Her enthusiasm for biomedical data science translated to me and has motivated me to carry on in this field.

Contents

Chapter 1

Introduction

1.1 Motivation

Diffuse large B-cell lymphoma (DLBCL), the most common type of non-Hodgkin lymphoma, is an aggressive cancer that develops from abnormal and rapid growth of a particular white blood cell called a B-cell [1]. Although DLBCL is most commonly observed in older patients, it can affect any age group, including children.

Genome-wide studies have established that DLBCL and other cancers exhibit strong heterogeneity which means that the diseases can manifest in patients very differently. There is a growing belief among researchers that part of the reason why patients diagnosed with the same disease respond differently to treatment is because they do not have the exact same disease on a molecular level. Much like a person may arrive at a destination using different modes of transport, it is now thought that patients may present with a similar collection of symptoms that are caused by different underlying disease pathways.

With technological advances in the past 20 years [2], such as tools to decode the human genome more efficiently and large-scale data analysis methods, the whole field of personalised medicine [3] has received significant attention. And so despite the recognised heterogeneity within cancer profiles, dominant features can now be extracted from biological data and clustered together to create cancer subtypes. Patient stratification is an approach to personalised medicine that aims to identify subgroups of patients based on molecular subtypes. The hope is that by finding molecular subtypes - each with distinct disease mechanisms - patients can receive more personalised, and, importantly, more effective treatment.

1.2 Objectives & Challenges

There are three papers that this project revolves around:

- "Genetic and Functional Drivers of Diffuse Large B Cell Lymphoma" [4] (referred to as the Reddy et al paper, or simply *Reddy*)
- "Molecular subtypes of diffuse large B cell lymphoma are associated with distinct pathogenic mechanisms and outcomes" [5] (referred to as *Chapuy*)
- "Genetics and Pathogenesis of Diffuse Large B-Cell Lymphoma" [6] (referred to as *Schmitz*)

Each of these papers proposes a methodology and accompanying DLBCL patient stratification via a dedicated classifier[1]. Each takes a different modality of data as input and my initial objective is to try and replicate some of their findings using the gene expression data of patients from the *Reddy* paper. This ties in to the main aim of the project which is to then propose a stratification for DLBCL patients. The scope covered by the three papers is larger than this project and so picking one specific area to analyse is important.

As suggested by the clinicians, we choose to study the NF-κB biological pathway which is a strong indicator of DLBCL. Specifically, we focus on one of the most important downstream genes in the pathway, GADD45B. We are given tumor gene expression and gene mutation data for a large

[1] An important note: a classifier in biology can be any supervised, semi-supervised, or unsupervised statistical algorithm unlike the stricter definition in machine learning which refers to a supervised classification task. We use the biological terminology throughout.

cohort of patients which we aim to cluster into subgroups with distinct genetic features. As an exploratory data analysis task, labelled data is not provided. We are, however, given clinical data of the patients which we use to evaluate our results.

The main challenge for this project is extracting the key information from the biological data. Biological data can often be very noisy because of the other background molecular processes operating alongside the relevant ones. Also, from a practical standpoint, consistent data collection procedures can be difficult, introducing experimental noise and making relative comparisons more difficult. Thus, finding a method capable of doing this is crucial. Based on the method used in the *Chapuy* paper, we focus on variations of non-negative matrix factorisation (NMF).

In addition to the main stratification task, the clinicians have also proposed the hypothesis that tumor purity (the amount of cancerous cells present in the sample) can affect results significantly. We investigate this throughout the project by comparing results on patients with data recorded on samples scored with two purity levels: low purity ($<70\%$) and high purity ($>70\%$).

1.3 Report Outline & Contributions

In Chapter 2, we first explore the biology to get a better understanding of the problem and motivate clustering as the best potential solution. We then look at survival models that are used in the three papers and our project for analysis of the patient cohort from a clinical point of view. We finish this section by detailing the core machine learning methods deployed in our project.

Chapter 3 looks at deriving a stratification methodology based on those presented in the three papers. We include a discussion of the specific challenges associated with biological clustering, a review of the three papers, and a summary of the NMF clustering algorithms we chose to investigate in this project.

Chapter 4 concerns the *Reddy* dataset. We detail the pre-processing strategies applied to the data and provide an initial analysis to motivate the subsequent clustering. Chapter 5 presents the results obtained with the NMF algorithms discussed in Chapter 3 and evaluates the clusters produced by the best algorithm using clinical data. We also look at gene clustering (as opposed to patient clustering/stratification) to elucidate any biological networks within these derived patient clusters.

Finally, in Chapter 6, we reflect on the project and propose future work in this domain.

In this project, we make the following contributions:

- Reproduce and build off results and strategies from the *Reddy*, *Chapuy*, and *Schmitz* papers where possible.[2]
- Identify correlations between GADD45B gene expression and clinical variables.
- Highlight the potential effect of tumor purity on biological analyses.
- Present an alternative classification method for DLBCL subtypes with comparison of probabilistic variations of the commonly used method: NMF clustering.
- Present a stratification of patients based on GADD45B gene expression using the best performing classifier and evaluate the results with clinical data.
- Investigate the key genetic features within each subgroup.

[2]Due to external issues, we were unable to get the classifier codes and data for all three papers and so we could not apply them to our data.

Chapter 2

Background

2.1 Relevant Biology

This section goes over the biology required to understand the project and its objective.

2.1.1 Cancer Cell Biology

Human bodies are made up of hundreds of trillions of cells. Although cells carry out different functions in different parts of the body, for the most part, they all contain a nucleus. The nucleus is like a control centre for the cell and contains chromosomes which are made up of genes. *Genes* contain the instructions to make molecules which control the cell's behaviour. They decide what sort of cell it will be, its function, when it divides, and when it dies.

Normally, genes make sure that the cells grow, divide and die in a controlled way to keep the body healthy. However, sometimes a change can happen in the genes when a cell divides - this is called a mutation. A mutation can disrupt the cell growth cycle and cause cells to multiply abnormally [7]. Tumors are clusters of these abnormal cells. It is important to note that tumors can be benign (do not invade surrounding tissues) but in this project, all the tumors mentioned are malignant (can invade surrounding tissues). Malignant tumors are much more likely to be harmful since they contain cancerous cells [8].

It is the rapid mutation of cancerous cells that make tumors so dangerous - among hundreds of mutations, one is bound to ensure that they can survive and continue proliferating through the body. In fact, through wide-scale cancer genome sequencing, it has been found that millions or even hundreds of millions of mutations are not uncommon in a typical tumor [9]. Despite this, the current belief is still that a few initial mutations in a cell are key to propagating tumor formation. These are referred to as *driver mutations*.

2.1.2 Heterogeneity

The millions of cellular and micro-environmental perturbations in each patient that lead to genetic mutations are what results in the same cancer appearing differently across patients. Tumor heterogeneity refers to this fact - that tumors do not contain one type of cell. Instead, they often contain different subpopulations of cells which have their own distinct *genotypes* and *phenotypes* that manifest in different biological behaviour [10].

This heterogeneity has posed problems for designing effective cancer treatment. Research has looked into prying out any common biological features (called biomarkers) of the same cancers in different patients so that we can discover novel (and reproduce existing) cancer subtypes, discover the subgroups of patients that associate with these subtypes, and thus deliver more targeted treatment.

2.1.3 Biomarkers

The exact definition of a cancer biomarker is not universally agreed upon [11]. In this project, a general definition has been used which refers to any measurable biological property or combined set of biological properties that could indicate a cancer. Recovering the origins of cancer subtypes is synonymous with figuring out the biomarkers of the cancer. In this project, the biomarkers

used are groups of genes (also called gene signatures) based on gene expression levels (a way of measuring gene activity).

2.1.4 Transcription Factors

A *biological pathway* is a series of interactions among molecules in a cell that leads to another molecule being produced or a change in cell behaviour. Pathways often facilitate gene activity and initiate cell movement. In this project, we focus on the NF-κB pathway which is a transcription factor.

Transcription is the process by which the DNA sequence of a gene is copied into RNA. A transcription factor is a protein that controls the transcription rate of genetic information by binding to a specific DNA sequence - essentially, they help turn genes "on" or "off" by binding to nearby DNA. Transcription factors can be activators (boost a gene's transcription rate) or repressors (decrease a gene's transcription rate).

2.1.5 The Problem: Understanding NF-κB

The swiss army knife of anti-cancer tools has traditionally consisted of a limited number of minimally effective chemo-therapeutics. Over the past 25 years, a better understanding of the driving mechanisms of oncogenesis (cancer) have opened the door for therapies that target specific oncogenic modules. This has enabled the era of stratified oncology and drastically improved the potential for cancer treatment. While several of these cancer-driving mechanisms and pathways have been successfully targeted therapeutically, there are still a group of pathways that have been resistant to treatment. Among these, "nuclear factor kappa-light-chain-enhancer of activated B cells"-family transcription factors (shortened to NF-κB) have been one of the most studied examples, and, simultaneously, the source of greatest disappointment due to their complexity [12].

NF-κB plays an essential role in several vital physiological processes like regulating the immune system and cell survival. For this reason, it is an ideal candidate for cancer to hijack and utilise for its own survival. In fact, abnormal activation of NF-κB is common in a large majority of cancers, including DLBCL. This aberrant NF-κB signalling dysregulates regular cell death and inflammation, instead enabling cancer cells to survive and orchestrating an inflammatory reaction in the tumor micro-environment. The mechanism used by NF-κB to carry out these two processes simultaneously is not fully understood.

A Potential Solution: GADD45B

Instead of investigating NF-κB head on, University of Chicago researchers inverted the problem to investigate downstream modules of the pathway. In other words, they focused on the effects of the pathway rather than the pathway itself. They identified that NF-κB was linked to another pathway involved in cell death, called JNK, through a gene called GADD45B [13] (see diagram in Fig 2.1). Further investigation of the GADD45B gene has shown that it could hold the key to understanding NF-κB's role in cancer. The gene was initially identified as a therapeutic target in multiple myeloma and subsequent research has looked at finding similar activity in different cancers for a multi-purpose drug.

Having identified a biological basis for therapy, the next step is to identify groups of cancer patients. While groups can be separated simply based on high and low GADD45B expression levels and then analysed, this method does not account for the complexity of human biology. Bioinformatics research has looked at discovering homogeneous subgroups of DLBCL patients based on large quantities of biological data, which can then be targeted for treatment if the subgroups exhibit significant relative differences in GADD45B expression. For these purposes, we turn to biological data analysis and the core of this project, gene expression analysis.

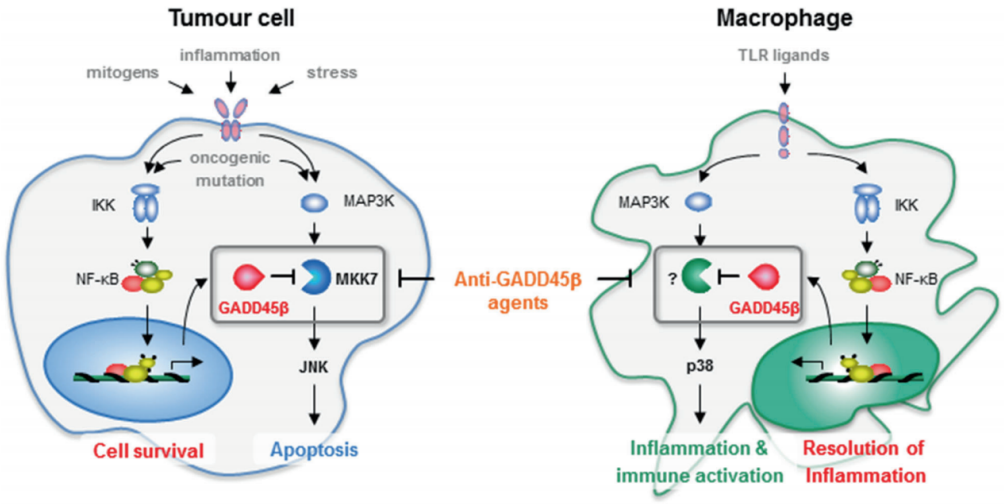


Figure 2.1: Simplified diagram to show GADD45β's[1] dual role in cancer and how therapeutics (anti-GADD45β agents) could allow regular NF-κB activity to continue whilst inhibiting GADD45β. [14]

2.1.6 Gene Expression Analysis

Gene expression is the process by which genes produce useful biological components, usually a protein. Gene expression data refers to the levels of these components produced in a sample and is a measure of gene activity. Acquiring expression data is a lengthy process and requires the use of sophisticated wet-lab machinery like microarrays. Microarrays are used to detect the expression levels of multiple genes simultaneously across a variety of samples. This produces a matrix of dimensions $n \times p$ corresponding to n samples (which are patients in this project) and p genes. Gene expression analysis looks at different ways of analysing this matrix. Methods fall into three broad categories:

- Class comparison (supervised): comparing different classes of genes to figure out which genes are statistically relevant to a certain class. Popular methods include significance analysis of microarrays or gene set enrichment analysis.
- Class detection (unsupervised): usually through cluster analysis to determine groups of genes or patients.
- Class prediction (supervised): creating models (usually with the help of machine learning) that can be used in disease diagnosis and prognosis.

This project is centred around class detection with the constraint that as little *a priori* biological knowledge as possible is used - we aim to keep the project purely data-driven to create reproducible results. Clustering can be used to group genes that have a similar expression patterns across different patient samples. The underlying assumption is that genes that express at similar levels are part of the same biological network and so genes in clusters are involved in concurrent biological processes. In the case of patient clustering, we define patient subgroups based on biological processes that are common to that subgroup.

[1]GADD45β is the protein encoded by the GADD45B gene. For our purposes, we do not need to distinguish between the two and use them interchangeably.

2.2 Survival Analysis

Survival analysis is a set of statistical methods which address the question "how long would it be before a particular event happens". This is especially important in medicine where doctors need to estimate a patient's *prognosis* (their chance of survival) based on statistics from previous patients.

Methods rely on a non-negative continuous random variable T, representing the time until some event of interest happens. In our case, T denotes the overall survival time (in years) i.e. time from diagnosis to death of a given patient.

The **cumulative distribution function (CDF)** of T, also known as the lifetime distribution function, is the probability that a subject dies by time t and is defined as

$$F(t) = P(T \leq t) \tag{2.1}$$

The **survival function** of T is the probability of a subject surviving beyond time t. It is the complement of the CDF and is defined as

$$S(t) = P(T > t) = 1 - F(t) \tag{2.2}$$

From this, we can find the **probability density function (PDF)** of T which is defined as

$$f(t) = \frac{d}{dt}F(t) = -\frac{d}{dt}S(t) \tag{2.3}$$

.

The **hazard function** of T is the rate at which events occur, given no previous events have occurred - "out of the people who have survived till time t, what is their instantaneous rate of dying?". In this project, the hazard function can be thought of as a measure of risk: the greater the hazard between two times, the more likely a patient will die in that time interval. It is defined as

$$h(t) = \lim_{\epsilon \to 0} \frac{P(T \leq t + \epsilon | T > t)}{\epsilon} = \frac{f(t)}{S(t)} = -\frac{S'(t)}{S(t)} \tag{2.4}$$

Censoring

Samples are called "censored" when information about their survival time is incomplete; the most commonly encountered form (and only form we use in this project) is right censoring. This corresponds to three main cases:

- When a patient does not experience the event of interest for the duration of the study.
- When a patient cannot be followed up during the study period e.g. because he/she dropped out of the study.
- A patient experiences a different event that makes further follow-up impossible.

2.2.1 Kaplan-Meier Estimator

A Kaplan-Meier estimator (KME) can be used to estimate the survival function [15]. The survival curve is created by computing probabilities of the event occurring at certain points in time, and then multiplying these probabilities together to get the final estimate.

For any time $t \in [t_i, t_{i+1}]$, with $0 \leq t_i \leq t_{i+1}$, we have:

$$S(t) = P(T > t) = P(\text{survive in}[t_{i-1}, t_i)) \cdot P(\text{survive in}[t_i, t) | \text{survive in}[t_{i-1}, t_i)) \tag{2.5}$$

$$= \frac{n_{i-1} - d_{i-1}}{n_{i-1}} \cdot \frac{n_i - d_i}{n_i} \tag{2.6}$$

where n_i is the number of patients "at risk" just before the i-th event occurs; d_i is the number of events at time t_i.

The KME is used to approximate the true survival function $S(t) = P(T > t)$ constructed by chaining all of the probabilities for each time interval together to give:

$$\hat{S}(t) = \prod_{i:t_i \leq t}^{k} \frac{n_i - d_i}{n_i} = \prod_{i:t_i \leq t}^{k} \left(1 - \frac{d_i}{n_i}\right) \tag{2.7}$$

where $\{t_i\}_{i=1}^{k}$ are the k distinct *uncensored* event times.

The idea is that if the sample size is large enough, the curve should approach the true survival function for the population being investigated. We often plot curves for different subsets of a group to see if there is a statistical difference in their respective survival times.

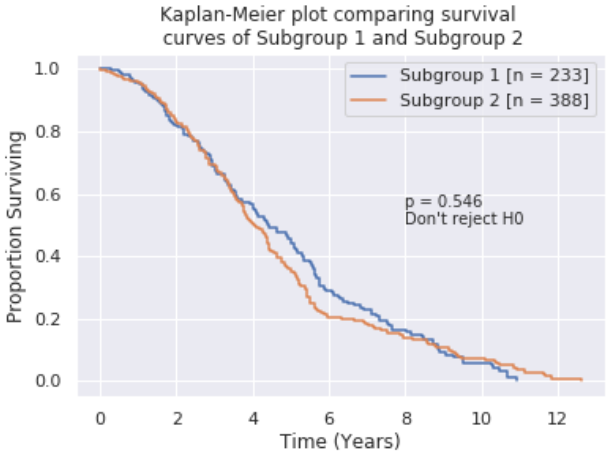


Figure 2.2: Kaplan-Meier plot of two different subgroups.

KMEs are used to examine survival time differences between n populations as shown in the Fig 2.2. We perform the pairwise logrank test (and multivariate logrank test if n>2) to determine statistically whether the n populations are different based on their survival curves. In our example, the logrank test returns a p-value=0.546 and the populations are not significantly different.

2.2.2 Cox Proportional-Hazards Model

The Cox Proportional-Hazards model (Cox PH) is a regression model to looks at the link between survival and one or more predictor variables (called *covariates*). It allows us to examine how these covariates affect the rate of an event occurring at a particular point in time, given that that event has not occurred already [16]. As described in Section 2.2, this rate is also known as the hazard function.

Cox PH is defined as:

$$h(t|\boldsymbol{Z}) = h_0(t)\exp\{\boldsymbol{\beta} \cdot \boldsymbol{Z}\} \tag{2.8}$$

where $\boldsymbol{Z}$ is a vector of covariates of interest (e.g. age, gender), $\boldsymbol{\beta}$ corresponds to the coefficients of the covariates, and $h_0(t)$ is the baseline hazard function.

The **hazard ratio** of two subjects $\boldsymbol{Z_1}$ and $\boldsymbol{Z_2}$ is defined as:

$$\frac{h(t|\boldsymbol{Z_1})}{h(t|\boldsymbol{Z_2})} = \frac{\exp\{\boldsymbol{\beta} \cdot \boldsymbol{Z_1}\}}{\exp\{\boldsymbol{\beta} \cdot \boldsymbol{Z_2}\}} = \exp\{\boldsymbol{Z_1} - \boldsymbol{Z_2}\} \tag{2.9}$$

Taking logs, we find that Cox PH can be expressed as a simple additive model, given by:

$$\log(h(t|\boldsymbol{Z})) = \alpha_0 + \sum_{i=1}^{\infty} \beta_i Z_i \tag{2.10}$$

The Cox PH model allows us to isolate the effects of individual covariates by assuming that the hazards are proportional to each other. This is useful as it allows us to compare the prognostic impact of different variables on survival.

- A variable with hazard ratio > 1 (coefficient = log(hazard ratio) > 0) is a bad prognostic factor. As we increase the variable it lowers the chance of survival.
- A variable with hazard ratio < 1 (coefficient = log(hazard ratio) < 0) is a good prognostic factor. As we increase the variable it increases the chance of survival.

Thus, interpreting the Cox PH model involves examining the hazard ratios (or coefficients which are log of the hazard ratios) for each explanatory variable. These variables can be visualised in a hazard ratio plot shown in Fig 2.3 along with a corresponding survival plot to show the effect of changing one of the variables on survival more clearly.

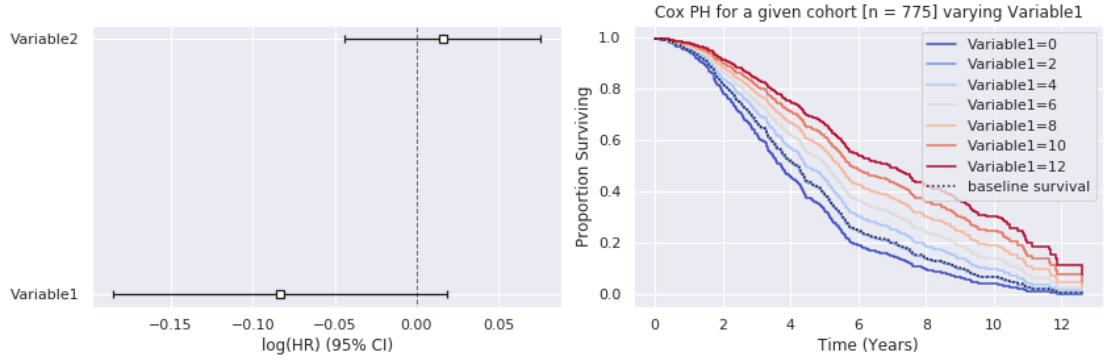


Figure 2.3: Cox PH plots: hazard ratios (left) with clearer visualisation of changing a variable (right).

In Fig 2.3, we see a simple Cox PH model with two variables. The hazard ratio plots tell us that `Variable1` has a coefficient of -0.08 and `Variable2` has a coefficient of 0.02. Thus, `Variable1` is a good prognostic factor and `Variable2` is a bad prognostic factor.

As a note: a univariate Cox PH model is equivalent to a KME but we use the two for different purposes in this project. KMEs do not take predictor variables into account and so are used solely to compare the survival curves of different cohorts. On the other hand, Cox PH is used to measure the effects of clinical factors on survival in a specific cohort.

Cox PH can use any regression technique as long as the outcome variable is the hazards ratio. In Section 5.4, following the method in *Reddy*, Cox PH is performed using elastic net regression to mediate model complexity, as discussed in Section 2.3.1.

2.3 Machine Learning Methods

Machine learning is a subset of artificial intelligence that tries to describe patterns and structures inherent in a given dataset. In my experiments, I will use elastic net regression to look at prognostic factors, and a range of unsupervised algorithms to help draw out subgroups of DLBCL patients.

2.3.1 Elastic Net Regression

Linear regression is a form of *supervised* learning that models the relationship between a continuous output variable and one or more independent variables. Given the data $\mathbf{X}$, we want to predict n observations of the response variable, $\mathbf{Y}$, by finding the unknown coefficients vector β for the following model:

$$\mathbf{Y} = \mathbf{X}\beta \tag{2.11}$$

In order to find β, we can use the ordinary least squares (OLS) approach to minimise the sum of squared residuals. This is the same as minimising the loss function given by:

$$L_{OLS}(\beta) = \sum_{i=1}^{n}(y_i - x_i\beta)^2 = ||\mathbf{Y} - \mathbf{X}\beta||^2 \tag{2.12}$$

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} ||\mathbf{Y} - \mathbf{X}\beta||^2 \tag{2.13}$$

While the OLS estimator is unbiased, it can have a large variance. This means the model does not generalise well to unseen data since it fits too strongly to the dataset it was trained on. A common remedy to this problem is to reduce variance at the cost of introducing some bias - usually through the addition of a "penalty". This process is known as *regularisation*. **Elastic net regression (ENR)** is a regularised regression framework that aims to minimise the loss function given by:

$$L_{enet}(\beta) = \frac{||\mathbf{Y} - \mathbf{X}\beta||^2}{2n} + \lambda\left(\frac{1-\alpha}{2}\sum_{i=1}^{m}\beta_j^2 + \alpha\sum_{j=1}^{m}|\beta_j|\right) \tag{2.14}$$

ENR combines two penalties: ridge and LASSO. This eventually results in a simplified formula for the coefficient parameters vector, given by

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} ||\mathbf{Y} - \mathbf{X}\hat{\beta}||^2 + \lambda_2||\hat{\beta}||_2 + \lambda_1||\hat{\beta}||_1 \tag{2.15}$$

Ridge penalty favours keeping more important features in the model with larger coefficients to preserve information, while LASSO penalty gets rid of the less correlated features to reduce the model's complexity. There is a trade-off between these two regularisions - and ENR aims to find the optimal combination of both penalties.

2.3.2 Non-Negative Matrix Factorisation

Dimensionality reduction is a method to reduce the number of features being considered. A popular paradigm for dimensionality reduction is matrix factorisation. Non-negative matrix factorisation (NMF) is a class of matrix factorisation algorithms that has become commonplace in bioinformatics over the past 15 years due its ability to accurately summarise high-dimensional genetic data.

From an algorithmic standpoint, NMF is a matrix decomposition method that factorises a non-negative matrix $\mathbf{X} \in {\mathbb{R}_+}^{n\times p}$ into a non-negative basis $\mathbf{W} \in {\mathbb{R}_+}^{n\times k}$ and non-negative coefficients $\mathbf{H} \in {\mathbb{R}_+}^{k\times p}$, where $k \ll min(n,p)$ such that

$$\mathbf{X} \approx \mathbf{WH} \tag{2.16}$$

The k columns that make up the $\mathbf{W}$ basis can be regarded as latent components that describe the main signals in the data. In the case of gene-based NMF, a latent component represents a subgroup of genes, and for patient-based NMF, a latent component represents a subgroup of patients. To this end, we refer to these latent components throughout the project as *metagenes* and *metapatients* respectively. Each of the k corresponding rows of the $\mathbf{H}$ matrix can be regarded as profiles for the corresponding latent components. Carrying on with our naming convention, we call these *metaprofiles*.

NMF algorithms generally optimise the following cost function:

$$\min_{\mathbf{W},\mathbf{H}} f(\mathbf{W},\mathbf{H}) = \frac{1}{2}||\mathbf{X} - \mathbf{WH}||_F^2 = \sum_i\sum_j (X_{ij} - (WH)_{ij})^2 \tag{2.17}$$

where $||\cdot||_F$ is the Frobenius norm (matrix equivalent of the Euclidean norm) such that $||\mathbf{A}||_F = (\sum_i \sum_j {A_{ij}}^2)^{\frac{1}{2}}$.

While the Frobenius norm is the most common cost function (and the one we use in this project), an adapted version of the Kullback-Leibler divergence for positive matrices [17] has also been quite popular due to its flexibility.

Most NMF algorithms initialise the basis **W** and coefficients **H** with random non-negative matrices and then utilise an iterative updating procedure based on gradient descent to calculate the final decomposition. NMF solutions are not unique and the iterative nature of the algorithms means that NMF solutions usually only converge to local minima. Importantly though, this is normally sufficient to acquire a descriptive summary of the data which is why the algorithm has seen continued usage in the field. We detail these algorithms in more depth in Section 3.3.

2.3.3 Cluster Analysis

Cluster analysis is the formal study of grouping, or clustering, objects according to their intrinsic characteristics and similarity [18]. It can be thought of as a way to find some structure in the given data. The absence of ground truth labels makes clustering an *unsupervised* learning task.

Hierarchical Clustering

Hierarchical clustering (HC), as the name suggests, involves organising the data into a hierarchy. The most common approach is the *agglomerative* approach. This is a bottom-up approach, where one starts by grouping individual data points into clusters and then grouping the resulting clusters until eventually the entire dataset is one big cluster. This is the approach used in this project.

There are two functions we predefine for the algorithm. The first is the **distance metric** for computing distances between clusters, and the second is the **linkage** which is the criterion we use to merge clusters. As an example for the linkage, if we are using complete linkage, we merge clusters based on the two clusters with the smallest *maximum* pairwise distance whereas if we are using a criterion like single linkage, we merge on the smallest *minimum* pairwise distance. We explain our choices in Section 3.2.5.

Hierarchical clustering is recursive and is as per the following steps:

1. Figure out which datapoint is the closest (based on our distance metric and linkage) to another datapoint, then merge them into a cluster.
2. Repeat this for each singular datapoint.
3. Repeat steps (1) and (2) for each cluster, but treat the cluster as an individual entity. In order to treat a cluster as an individual entity we need some way of averaging the data points associated in a cluster.
4. Terminate when all the clusters have been combined into one big cluster that contains the whole dataset.

More formally, we can describe the initial data set **D** as being partitioned into n ordered sets: $S_1, ..., S_n$. If subsets C_i and C_j satisfy $C_i \in S_m$, $C_j \in S_l$, and $m > l$, then either $C_i \subset C_j$ or $C_i \cap C_j = \emptyset$ for all $i \neq j$ for $m, l = 1, ..., n$. That is, for any two subsets, either one subset contains the other subset completely or they are disjoint. This structure of nested subsets can be represented by a binary tree called a *dendrogram* (shown in Fig 2.4) where the leaf nodes are the individual data points. The relative vertical lengths of a parent node to its children give an indication of difference between the children.

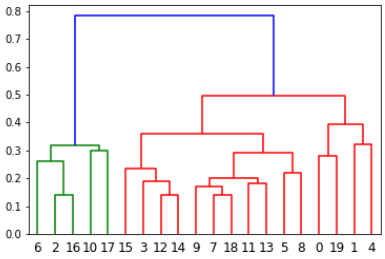


Figure 2.4: Dendrogram showing hierarchical clustering of 20 patients into two clusters.

Once we have the dendrogram structure of our data, we then need to "cut" it in such a way that we obtain an optimal number of clusters. **Choosing the "right" number of clusters** is not an easy problem since hierarchical clustering algorithms always create a hierarchy, regardless of whether one actually exists. The simplest method is to inspect the hierarchy once it has been produced and search for the biggest "gap" between two clusters in the dendrogram. However, this gap is often not clearly identifiable, and more importantly, may not be very meaningful, especially when dealing with real-valued numbers like our gene expression data. We manage to circumvent this problem using a compound hierarchical clustering method detailed in Section 3.3.

Dendrograms are often used in genomics alongside *heatmaps* to help visualise genetic data more clearly. We call this combined diagram a *clustermap* (see Fig 2.5). In this diagram, genes are encoded over rows, and the columns represent individual patient samples. Groups of squares of similar colours indicate clusters.

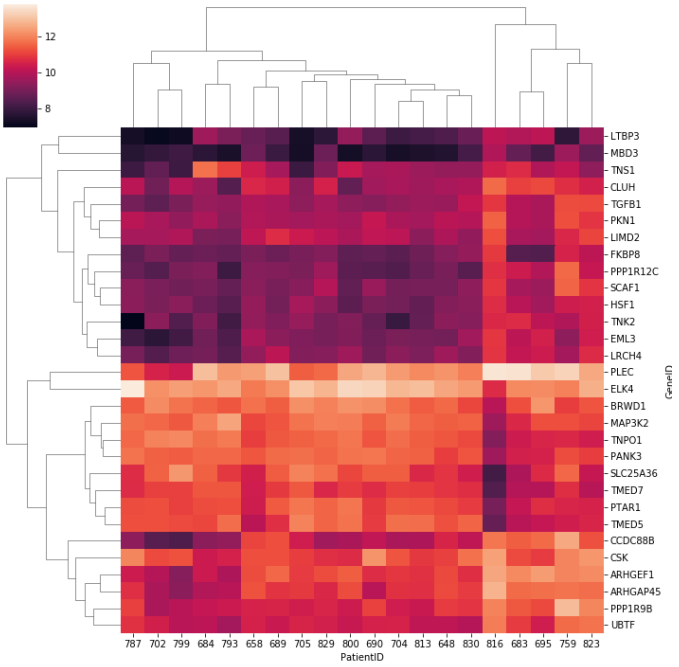


Figure 2.5: Clustermap showing patient and gene clustering of 20 patients with 30 genes.

Chapter 3

Stratification Methods

Here, we describe the challenges associated with biological data clustering [19] and provide a review of the key papers in DLBCL omics research. We then explain our direction for the project based on these papers.

3.1 Challenges in Omics Clustering

3.1.1 "Large P, Small N" Problem

Omics data is notorious for having thousands of genomic features with a small (<1000) number of samples - referred to as the "large p, small n" problem. This falls under the well-known *curse of dimensionality* issue. To summarise the issues, as dimensionality increases beyond three-dimensional spaces, geometry starts to behave non-intuitively, sparsity is commonplace, and important signals in the data can get **masked by noise**. In the case of data with a small number of samples, this masking effect is amplified and the inherent structure of the data may only be detectable via lower-dimensional representations.

In omics data, noise either comes from experimental procedures or other biological processes which are less relevant to the main process being considered. To obtain biologically-relevant clusters from high-dimensional data, we need to deal with this noise. The two most common approaches are dimensionality reduction and subspace clustering.

Defining Pairwise Similarity

As mentioned above, geometry behaves strangely in higher dimensions and so it is important that we pick an appropriate distance metric for our non-negative, real-valued gene expression data. In biological clustering, there are different stages which require distance metrics: feature selection, feature extraction, and clustering.

3.1.2 Cluster Requirements

Identifying Number of Clusters

How many clusters are present in the data and distinct are they? These questions are related to the chosen **clustering paradigm**. By definition of patient stratification, we want the patient clusters to be separate or contained (i.e. no overlaps unless they are proper subsets of another cluster) since this makes treating groups of patients easier. While soft clustering may be relevant (and perhaps even more accurate) on a biological level, it makes drug development much harder since complex interactions between patient subtypes have to be considered.

Given a dataset of n patients, we can get different sizes of patient subgroups depending on the number of clusters, k, that we set. Evidently, as we increase k, the average size $\frac{n}{k}$ of each subgroup will become smaller. This produces a trade-off: we want small enough subgroups so that patients within a subgroup are sufficiently similar, but we also want big enough subgroups for further study to develop reliable targeted therapeutics. For this reason, methods that incorporate this trade-off into the solution are useful. Note that when average cluster size increases, some individual subgroups may be not get bigger. For example, given 300 patients and $k = 3$, the average size is 100 but we may get subgroups of sizes 140, 140, and 20. In this case, it makes more sense to flag the smaller subgroup as requiring further verification, or as an "unclassified" subgroup. Overall,

we are looking for a combination of subgroup size near to the expected average, clear homogeneity within a group, and clear heterogeneity between different groups.

Ensuring Robust Clusters

Given the complexity and dynamic nature of biological systems, it is clear that there is no one-size-fits-all approach in biology. This statement applies to exploratory data analysis too. We also note that clustering algorithms can often find clusters even if there is no inherent structure in the data. However, to ensure a rigorous evaluation, we can take some measures to produce robust clusters. Here, we define 'robust' as reproducible and reliable clustering results. The way to ensure this is by approaching the problem more holistically. The two most common methods of doing this are performing ensemble clustering or integrating different modes of data.

3.1.3 Project-Specific Requirements

An important part of this project is **interpretability** for clinicians and biologists. The easiest way to do this is by using graphs to visualise our results. Common visualisations in the biological literature are scatter graphs associated with simple partitioning algorithms like k-means and dendrograms associated with hierarchical clustering. Heatmaps are also used frequently to visualise genetic patterns.

Another requirement is that this project is kept as **data-driven** as possible. Cluster analysis used here is meant to reveal inherent biological network structures and so as little as possible *a priori* knowledge on the genetic relationships should be encoded into the methods to avoid introducing some bias.

Clinical Relevance

In Section 3.1.2, we describe some options to obtain robust clustering from a statistical point of view. However, fundamentally, these results can be inconsequential if they fail to show clinical validity. Thus, the most important project-specific requirement is clinical evaluation through measures like GADD45B gene expression in clusters and survival curve separation.

3.2 Stratification Methods in DLBCL

As mentioned in the introduction, there are three key papers (*Reddy*, *Chapuy*, and *Schmitz*) that this project is based upon. We review the methods used in each of them, compare them and assess how they solve the challenges described above, and then develop our stratification method. The scope covered by the three papers is broad, and so we restrict this review to the methods relevant to our exercise: **pre-processing**, **classification** and **post-clustering procedure**.

3.2.1 Reddy

The *Reddy* paper found genetic alterations associated with different DLBCL gene signatures and then analysed the combinatorial combinations of these alterations with expression data to create a new prognostic factor called the "Genomic Risk Model". Their methodological outline is as follows:

- **Pre-processing**: They have gene expression data for 775 patients from which the authors defined a core group of 624 for further analysis. They identified 9,500 lymphoma-based gene sets from curated databases and lymphoma-specific sources. From these 9,500 gene sets, they identified 1,228 which were highly correlated with the expression data they had for the 624 patients.
- **Classification**: They observed a large amount of correlation and redundancy between these 1,228 gene sets. To cancel out the noise and quantify the key gene sets, they fed the remaining gene sets into an **affinity propagation clustering** algorithm [20] leading to 31 clusters. Each of the 31 clusters was associated with an "exemplar" gene set to represent it.

- **Post-clustering**: They identified gene mutations associated with high and low expression of each exemplar gene set. Next, they explored the role of these mutations in clinical outcome through survival analysis. Finally, they implemented a *Coxnet* model with cross-validation to define association of 150 driver genes and three gene expression markers with patient survival. This model separated patients into three levels of risk, creating the "Genomic Risk Model".

3.2.2 Chapuy

The *Chapuy* paper used a Bayesian NMF clustering method to analyse the somatic mutations of 304 DLBCL patients, producing five new subtypes of DLBCL cancer. Their methodological outline is as follows:

- **Pre-processing**: They have different types of mutations (binary data) for 304 patients. They encoded mutation types based on severity on a scale of 0, 1, 2, 3 stored into a matrix.
- **Classification**: They passed the mutation matrix into a **Bayesian NMF consensus clustering** algorithm. The algorithm exploits automatic relevance determination (ARD) - a shrinkage technique to allow for a sparse representation of the mutation signatures and clearer distinction for the number of signatures by removing less important features [21]. This produced five well-defined patient subgroups (and one unclassified subgroup). Feature selection using a one-sided Fisher's exact test was then used to get the key mutations in each patient cluster.
- **Post-clustering**: They compared relevant clinical variables between the clusters including survival. Groups that are heavily *COO-classified* were combined and compared. They also compared the hazard ratios for each group (based on their genetic features) with an existing risk assessment called the international prognostic index (IPI).

3.2.3 Schmitz

The *Schmitz* paper created and used a classifier algorithm called "GenClass" on somatic mutations of 574 patients, producing four new subtypes for DLBCL. Their methodological outline is as follows:

- **Pre-processing**: They worked with the main genetic features that were found significant in 574 samples. They also identified whether the selected mutations were correlated with the most studied existing stratification in DLBCL known as the COO classification.
- **Classification**: They used **GenClass** to cluster patients into four classes. GenClass initialises four classes with the main genetic features found earlier and then iteratively moves samples between classes to maximise association of classes with features. They then used permutation testing to verify that the clusters were better than would be expected by chance.
- **Post-clustering**: They performed clinical validation of clusters by looking at relations to COO classification and separation in terms of survival times. To demonstrate clinical efficacy, they also developed a supervised random forest classifier using the created classes.

3.2.4 Comparison

These three papers use different cohorts, stratification methods, encoding of their input data, and tests to evaluate clinical validity of their results. Quantitative comparison is therefore not relevant, and we focus on a qualitative comparison. The *Reddy* paper is slightly different from the other two in that it aims to combine subtypes to produce a novel prognostic factor called the Genomic Risk Model (GRM). As such, they do not define clusters of different genetic features, but rather clusters of different *sets of* genetic features to find larger groups of patients. To illustrate this, we'll make a concrete comparison between one of the clusters defined by *Schmitz* and one by *Reddy*. *Schmitz* defines a "BN2" cluster for patients that have a combination of *BCL6* gene fusions and *NOTCH2* gene mutations, whereas *Reddy* defines a GRM level (cluster) of 1 for all patients that are associated with (the majority of) 15 different types of gene mutation combinations.

We summarise the three papers in Table 3.1, comparing their decision choices regarding the challenges we outlined in Section 3.1 in the hopes that it will make our methodological choice clearer.

Design Consideration	*Reddy*	*Chapuy*	*Schmitz*
Pre-processing & dealing with noise	Filtered initial 9,500 gene signatures to 1,228 using correlations with gene expression data	NMF feature extraction in clustering algorithm	MICE imputation at an earlier stage
Clustering Algorithm	Affinity propagation clustering	Bayesian NMF consensus clustering with ARD	GenClass iterative clustering
Clustering input/output	1,228 gene signatures/31 gene signature clusters	Patient-mutation matrix/Five (+one irrelevant) patient clusters	Patient-mutation data/Four patient clusters
Ensuring robustness	GRM: use multiple feature combinations to define a subgroup	Ensemble method (consensus NMF) + regularisation (ARD)	Ensemble method (random forest) on generated subtypes
Extracting info from clusters	Find gene mutations correlated with gene sets	One-sided Fisher's exact test to find key gene mutations	Permutation testing
Evaluation	Enumerate feature combinations and feed into supervised *Coxnet*	Survival analysis	Permutation testing and survival analysis

Table 3.1: Qualitative comparison of the three studied papers (*Reddy*, *Schmitz*, *Chapuy*) for the study designs and evaluation of biological clustering results.

3.2.5 Our Method

Our main objective is patient stratification through gene expression data clustering. Since none of the three studied papers use gene expression as the input for their classifier, we first constrain the problem to satisfy our specific requirements (dealing with the "large p, small n" problem) and then assess whether any of the three algorithms fit our constraint. In particular, we focus on methods that incorporate dimensionality reduction to deal with our high-dimensional gene expression data.

The GenClass iterative algorithm is not strictly a clustering algorithm and requires initial preparation to work out the required seed class features. It is not applicable in our case. Between affinity propagation clustering and NMF consensus clustering, the robustness of the NMF method makes it a more suitable choice. NMF has been used frequently in bioinformatics and has also shown less sensitivity to the initial selection of genes [22].

We looked more broadly at other dimensionality reduction methods like PCA but settled on NMF because it was more suited to our problem description. Brunet et al. [22] were some of the first to explore NMF in gene expression clustering and observed that with each NMF update iteration, the profiles of the metagenes become more sparse and localised. This allows for greater distinction between class boundaries and so easier interpretation for clustering. On the other hand, PCA produces orthogonal profiles that are denser and globally supported making class distinction more difficult. In order to produce the best possible clustering results within our framework, we decided to compare different **NMF consensus clustering** methods. We focus on probabilistic methods since they are more focussed on dealing with noise. The probabilistic NMF methods are outlined in the next section.

NMF consensus clustering is essentially NMF feature extraction followed by a clustering method. As mentioned in the challenges section, biological data often represents an underlying structure. We choose **hierarchical clustering** for this purpose, and because of its easy interpretability. We also manage to get around the problem of deciding the optimal cut for hierarchical clustering by using NMF beforehand. NMF naturally produces clusters [23] based on an input parameter called the rank. The rank corresponds to the number of clusters and so we know the "right" number of clusters from the start. Centroid and average linkage are the most common linkage criteria in biology; Brunet et al. [22] found average linkage to perform better and so we use **average linkage** in our work. We perform NMF clustering by optimising a cost function based on the Frobenius norm which is analogous to the Euclidean norm. For this reason, we choose to use the **Euclidean distance metric** throughout our project to maintain consistent results.

In terms of pre-processing, the only method that has a comparable number of initial elements to our 13,128 genes is the *Reddy* paper which filters 9,500 gene sets. They use correlations between gene sets and gene expression data to find the most relevant gene sets. Although we have mutation data, it is harder to quantify correlations between the binary mutation data and real-value gene expression levels unless we include clinical data which would go against the unsupervised nature of the exercise. Thus, we turn to two feature selection methods: **variance thresholding** as a simple initial method and **Laplacian score (LS) filtering** [24] for final gene selection because of its ability to preserve local structures of the data. With the knowledge that we are using NMF algorithms later which produces parts-based (localised) representations, we felt that LS filtering would prioritise the features we needed for meaningful results.

We do not include clinical data in the clustering so that we can focus on the biological relationships. In terms of evaluation, both *Chapuy* and *Schmitz* use the well-known COO classification to help ascertain the validity of their results. For this reason, we display the COO classifications in our clustermaps in case they give a clear indication of intra-group homogeneity. Beyond GADD45B and NF-κB correlation, we look at survival analysis and employ a similar Cox PH model to *Reddy* in order to identify key genes in each subgroup.

3.3 NMF Consensus Clustering

In this section, we detail the different NMF consensus clustering (NMFCC) algorithms that we will compare in Section 5.2.3. We first introduce the standard method in Section 3.3.1. Each of the following methods differs on how the matrix decomposition is performed. Without loss of generality, we use patient clustering examples for the rest of this section.

3.3.1 Standard NMFCC

NMF is widely utilised because of its inherent clustering property. That is, for any given rank k (the rank of the matrix decomposition), NMF groups the data into k clusters. The result of a single NMF run can be described by a connectivity matrix. On each run, the algorithm produces a square *connectivity matrix* $\mathbf{C}$ of size $n \times n$ (n is the number of patients) with an entry $C_{ij} = 1$ if patients i and j are in the same cluster, and $C_{ij} = 0$ otherwise.

To determine whether the clusters are reliable, We can use *consensus clustering*. Consensus clustering is one type of ensemble clustering methods commonly used to produce robust results for algorithms with some stochasticity (e.g. random initialisation). NMF consensus clustering [22] exploits the random initialisation of the NMF procedure using consensus clustering. Since each NMF run produces a connectivity matrix, consensus NMF averages over the connectivity matrices to produce a square *consensus matrix* $\bar{\mathbf{C}}$ of size $n \times n$. Since each connectivity matrix contains entries 0 or 1, the consensus matrix contains entries in the range $[0, 1]$. With a sufficient number of runs, $\bar{C}_{ij}$ converges toward the probability that the patients i and j cluster together and thus gives us a more reliable set of clusters than a single NMF run.

The consensus matrix can be regarded as a similarity matrix which can be fed into a clustering algorithm to identify groups of patients that cluster together. Following its notable usage in the literature, we focus on hierarchical clustering. We show the clustermap of a consensus matrix in Fig 3.1. The darker colours on the map indicate a higher frequency of patients clustering together. Note that the matrix is symmetric and so the column and row dendrograms are the same. For this reason, we will leave out one of the dendrograms in the final results.

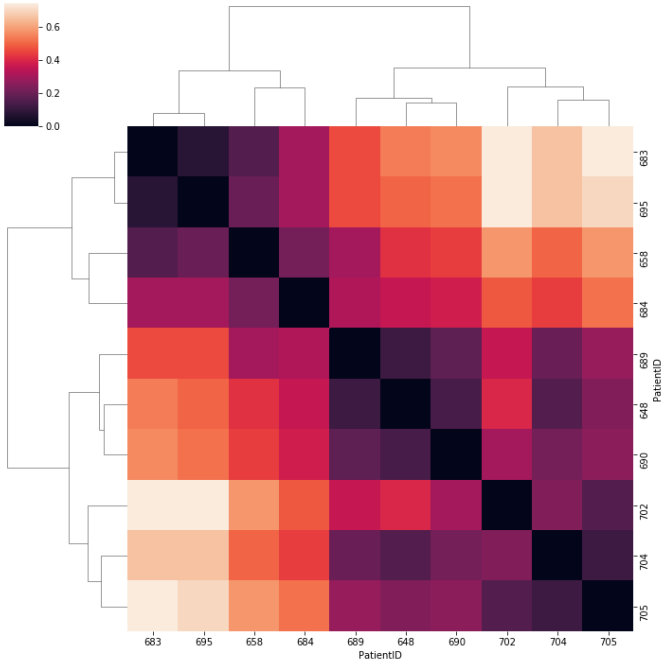


Figure 3.1: Clustermap of a NMF consensus matrix of 10 patients.

3.3.2 Probabilistic NMFCC

Probabilistic NMFCC uses a probabilistic NMF (PNMF) algorithm for the feature extraction stage. PNMF tries to infer the distribution over $\mathbf{W}$ and $\mathbf{H}$ after observing the dataset D. It finds a reasonable approximation by using an expectation-maximisation algorithm to iteratively update each distribution. In other words, we are updating $p(\mathbf{W}|D)$ and $p(\mathbf{H}|D)$ to approximate $p(\mathbf{X}|D)$.

3.3.3 Bayesian NMFCC

Bayesian NMFCC uses Bayesian NMF methods for the feature extraction stage. Bayesian NMF is similar to probabilistic NMF in that both methods try to infer the distribution over $\mathbf{W}$ and $\mathbf{H}$ after observing the dataset D. However, Bayesian NMF also places priors over $\mathbf{W}$ and $\mathbf{H}$, which leads us to find a posterior. We outline two algorithms that perform approximate posterior inference: the Gibbs sampler approach and the iterated conditional modes approach.

Gibbs Sampler

We want to figure out the posterior of the parameters given the data: $p(\boldsymbol{\theta}|D)$. This distribution is often computationally intractable and Gibbs sampling can be used to draw values from it and approximate the posterior [25].

Gibbs sampling draws a set of samples from the conditional posteriors of each of the model parameters ($\mathbf{W}$, $\mathbf{H}$, and noise variance σ^2) which, when combined, describe a sample from the joint posterior. In this way, we can build up samples from the joint posterior to get a reasonable approximation of its true form.

Using the priors outlined in the original paper [26], the conditional posteriors of $\mathbf{W}$ and $\mathbf{H}$ are proportional to a normal distribution multiplied by an exponential distribution resulting in a rectified normal distribution (a normal distribution where values less than zero are set to zero). The conditional posterior of σ^2 is an inverse Gamma distribution.

The algorithm is as follows:

1. Initialise the basis $\mathbf{W}$ and coefficients $\mathbf{H}$ with exponential priors.
2. Sample from a rectified Gaussian for each column of $\mathbf{W}$.
3. Sample from a rectified Gaussian for each row of $\mathbf{H}$.
4. Sample from the inverse Gamma for σ^2.
5. Repeat steps (1-3) until a convergence condition is met.

Iterated Conditional Modes

The iterated conditional modes (ICM) algorithm is similar to the Gibbs sampler, but takes the mode of the conditional posteriors of each parameter at each iteration rather than sampling from the conditional posterior. So where we sample from rectified Gaussian distributions and the inverse Gamma distribution in the Gibbs sampler, we simply take modes of these distributions when using ICM. This produces a maximum *a posteriori* (MAP) estimate as opposed to a full posterior distribution.

3.3.4 Implementation Details

Python and R are the two go-to languages for statistical programming and machine learning. The libraries recommended by my supervisor had more support on the Python side and so Python ended up as the chosen language. In particular, `NumPy` [27] and `pandas` [28] were used for all general data handling; the library `Nimfa` [29] was used to handle all NMF algorithms.

All experiments and analysis were run on a Lenovo IdeaPad 720S-13IKB laptop, with 3.5 GHz Intel Core i7 processor (dual-core), 8 GB RAM, and Intel HD 620 graphics card.

Chapter 4

Data & Preprocessing

4.1 Data Pre-processing Strategies

Some feature values may simply be in an unsuitable format for statistical analysis. Variables can broadly be split up into four categories: continuous, categorical, binary and ordinal. We want all the prognostic variables to be in an easy-to-analyse format (i.e. continuous or ordinal) so that they can be used in our clustering or survival models.

The data provided to us requires a reasonable amount of pre-processing and I have outlined my general strategies here. I then describe the datasets in more detail and discuss specific instances of where the strategies are applied.

4.1.1 Imputation & Substitution

Imputation is the process of replacing missing data with values that can be used for analysis. Missing values in the the data used for GADD45B and clinical information regression are represented by `NaN`s ("Not a Number"s). These `NaN`s often raise errors during regression. Samples with `NaN`s can be removed but since our cohort is small we preferred to replace the `NaN` values with reasonable values. *Substitution* re-encodes data with values that are more suitable for the task, such as ordinal to continuous encoding. The three strategies used have been outlined below:

- **Constant Value Replacement**: A simple imputation and substitution strategy that replaces values with a constant.
- **Gaussian Distribution Replacement (GDR)**: This is a substitution strategy to make ordinal values *that represent ranges* continuous. We replace values in a range via sampling from a Gaussian distribution with a mean at the centre of the class range. This assumes that the underlying distribution which the samples are drawn from is Gaussian. This is a strong assumption but preserves the initial ordinal distribution while adding continuity.
- **MICE**: Some feature values may be difficult to interpolate/extrapolate and more advanced methods may be required. In these cases, we employ the imputation technique used in *Schmitz*: Multivariate Imputation by Chained Equation (MICE). This method is implemented using the Python package `statsmodels` [30]. The MICE algorithm imputes each incomplete variable by a separate model and works on the three types of data described above that appear in our data.

4.1.2 Feature Selection

In the absence of labels to evaluate the importance of features, unsupervised feature selection methods use alternative criteria to define the relevance of features [31]. There are three types of feature selection methods:

- Filter methods: feature selection is independent of the learning algorithm.
- Wrapper methods: iterative improvement of the quality of the selected features using a learning algorithm.
- Embedded methods: feature selection is directly integrated into the learning algorithm.

The clustering we perform after feature selection is very computationally intensive and so we choose to focus on filter feature selection methods since they are independent of the clustering algorithm. These methods naturally favour features that are easier to cluster and thus a limitation is that they may remove features which are biologically meaningful but not suited to clustering algorithms. We aim to mediate this problem by including enough genes and performing consensus clustering later on.

We use two feature selection filter methods on the gene expression data: variance thresholding and Laplacian score filtering.

Variance Thresholding

Variance thresholding is a simple feature selection algorithm, removing features with variance below a certain threshold. The assumption when using this algorithm is that features that do not vary a lot across patients carry little predictive power and so can be discarded. The only parameter associated with this algorithm is the variance threshold.

Laplacian Score Filtering

Laplacian Score (LS) filtering is based on the assumption that if two data points are close, they are related to the same underlying property of the data. In this way, LS filtering prioritises local structure over global structure. To model the local structure, it uses a nearest neighbour graph to determine which features fit well. The smaller the Laplacian score, the more important the feature is. The algorithm used in the original paper [24] to construct the Laplacian scores is as follows:

1. Construct a k-nearest neighbour's graph $\mathbf{G}$ with n nodes where n is the number of samples. Each node i corresponds to a sample x_i. We put an edge between two nodes i and j if the two corresponding samples x_i and x_j are close i.e. x_i is in the k nearest neighbours of x_j or vice versa.
2. Create a weight matrix $\mathbf{S}$ where $S_{ij} = e^{\frac{-dist(x_i,x_j)}{t}}$ if nodes i and j are connected in $\mathbf{G}$, else set $S_{ij} = 0$. t is a hyperparameter to set, $dist(\cdot,\cdot)$ function to be chosen.
3. For each of the r features f_p, $p = 1, .., r$, we form a graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{S}$; $\mathbf{D} = diag(S\mathbf{1})$.
4. Let $\widehat{f}_p = f_p = \frac{f_p^T \mathbf{D}\mathbf{1}}{\mathbf{1}^T \mathbf{D}\mathbf{1}}\mathbf{1}$. Compute a Laplacian score L_p for each feature using the equation $L_p = \frac{\widehat{f}_p^T \mathbf{L}\widehat{f}_p}{\widehat{f}_p^T \mathbf{D}\widehat{f}_p}$.

4.2 Reddy Dataset

This project is designed around the dataset from the *Reddy* paper. This dataset has a number of patients for whom we have **clinical data** (e.g. response to initial therapy, overall survival years, etc.), **gene expression data** (levels of genes expressed in each patient's tumor - essentially a measure of gene activity in a sample), and **mutation data**. We also have different **NF-κB gene signatures** (as NF-κB can manifest through different sets of genes) to aid in future evaluation. The dataset was provided by the group of Prof. Guido Franzoso at Imperial College, Hammersmith Campus.

Where useful, feature distributions are plotted and then the best strategy for imputing the values is described. Of course, pre-processing is an integral part of machine learning and can greatly affect the results - in cases where pre-processing is applied, an explanation is provided.

4.2.1 Clinical Data

Details & Pre-processing

For the Reddy dataset, we have clinical data for 1,001 patients. Since we only have the gene expression data for 775 patients, we combine these two records based on the Sample ID of the patients to end up with 775 patients with both gene expression and clinical data. We now describe the variables encoded in the provided data:

- **Sample ID**: identification number for the patient. This is used to merge different records that are all indexed by sample IDs.
- **International Prognostic Index (IPI)** [ref: A predictive model for aggressive non-Hodgkin's lymphoma]: this is a risk assessment that has been used to predict survival outcomes for patients with aggressive non-Hodgkin's lymphoma (which DLBCL falls under). The IPI score stratifies patients into four prognostic groups based on a set of clinical factors: low risk: (0, 1); low-intermediate-risk: 2; high-intermediate risk: 3; high risk: (4, 5). We use the numerical representation of IPI in this project.
- **Genomic Risk Model (GRM)** [ref reddy]: this is a risk assessment proposed by *Reddy* to help stratify patients more accurately than with the IPI. The GRM takes categorical values: low risk, medium risk, high risk. Survival analysis in *Reddy* has shown that the IPI is better at predicting early mortality whereas the GRM is better in predicting both early and late mortality and so we include both as potential prognostic factors.
- **Age at diagnosis**: the age of a patient when they were first diagnosed with DLBCL.
- **Response to initial therapy**: percentage decrease in cancer cell count (i.e. tumor decrease) following first treatment. It takes three categorical values: no response, partial response, complete response. In cancer terminology, a 'complete response' does not mean that you are cured - it is simply the best starting point for a cure. In a similar way, 'no response' means that the cancer has not decreased after treatment but does not mean that you are incurable. We have assumed, since the data is not more descriptive, that 'no response' suggests that the extent of a cancer does not increase after treatment.
- **ABCGCB**: corresponds to the most widely-studied and documented stratification of DLBCL based on the COO classification. There are three subtypes represented by categorical values for: activated B-cell like (ABC), germinal center B-cell like (GCB), and Type-III (unclassified) subtypes.
- **Overall survival years**: time from first recorded treatment until death from any cause. This is the time variable used in Kaplan-Meier and Cox PH plots.
- **Censored**: as described in Section 2.2, this refers to whether the patient's death is censored or not. It takes binary values: 0 for censored data (no death observed) and 1 for uncensored data (death observed at time given).
- **Tumor purity**: refers to how much of a patient's sample contains cancerous cells i.e. is cancerous. The hypothesis from the clinicians is that lower purity samples affect the accuracy of the results and so part of this project is investigating this claim.

Since we only have a small number of samples, an effort was made to preserve as much data as possible. The missing values are imputed per feature with the three imputation strategies detailed above.

Table 4.1: Percentage of missing values in *Reddy* cohort [n=775] (1 d.p.)

IPI	Genomic Risk Model	Overall survival years	Censored
19.6%	2.8%	2.5%	0.9%
ABCGCB	**Response to initial therapy**	**Tumor Purity**	**Age at diagnosis**
0.0%	6.6%	0.0%	0.0%

Firstly, we compare age distribution in our cohort to similar distributions from the literature. From Fig 4.1, we can see that the distributions are very similar and so we have a sample that is representative of the true distribution. We note that age has not shown to be a prognostic factor for DLBCL diagnosis although DLBCL incidence does increase with age.

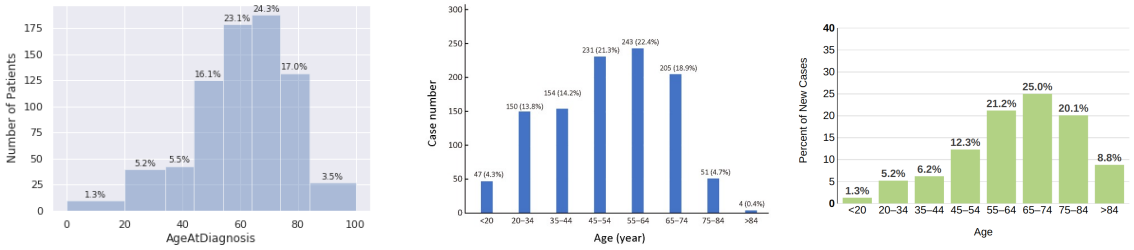


Figure 4.1: Histograms of age at diagnosis of the *Reddy* cohort [n=775] (left) against a large cohort [n=1,085] from a Chinese study[1] (middle) and new cases [exact cohort size not given] in the USA from 2013-17[2] (right).

Next, we look at the data for 'Response to initial therapy'. Originally, this variable had values as 'No response', 'Partial response', and 'Complete response', along with `null` for the missing values. These values were not found to be very useful for our analysis since 'partial response' could take values over the whole range of responses and so we used the *GDR* strategy here.

We decided on a 5% buffer for the 'no response' and 'complete response' values with the *GDR* strategy. We replaced the 'no response' category with continuous values sampled from a $Normal(2.5, 1.5)$ distribution in the range [0, 5]; the 'complete response' category with continuous values sampled from a $Normal(97.5, 1.5)$ distribution in the range [95, 100]; the 'partial response' category with continuous values sampled from a $Normal(55, 15)$ distribution in the range [5, 95]. Here we pushed the mean above the standard mean of 50 given that 77.2% of the patients showed a 'complete response'. The null values were imputed with a simpler constant replacement method so they take the value of 80.

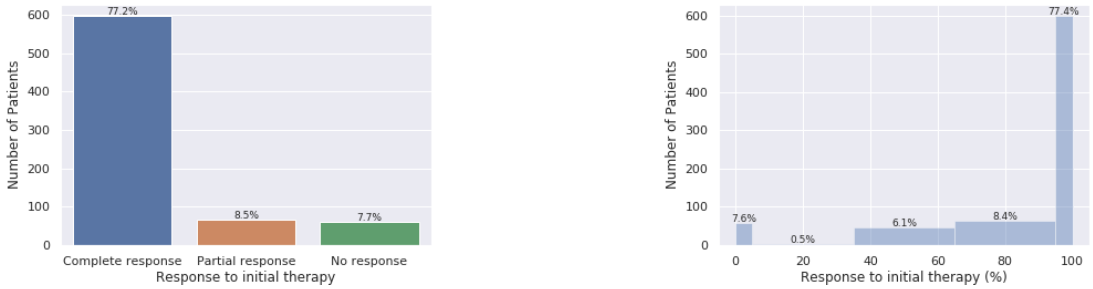


Figure 4.2: Bar charts of response to initial therapy distributions before (left) and after (right) pre-processing.

[1] `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6433587/`
[2] `https://seer.cancer.gov/statfacts/html/dlbcl.html`

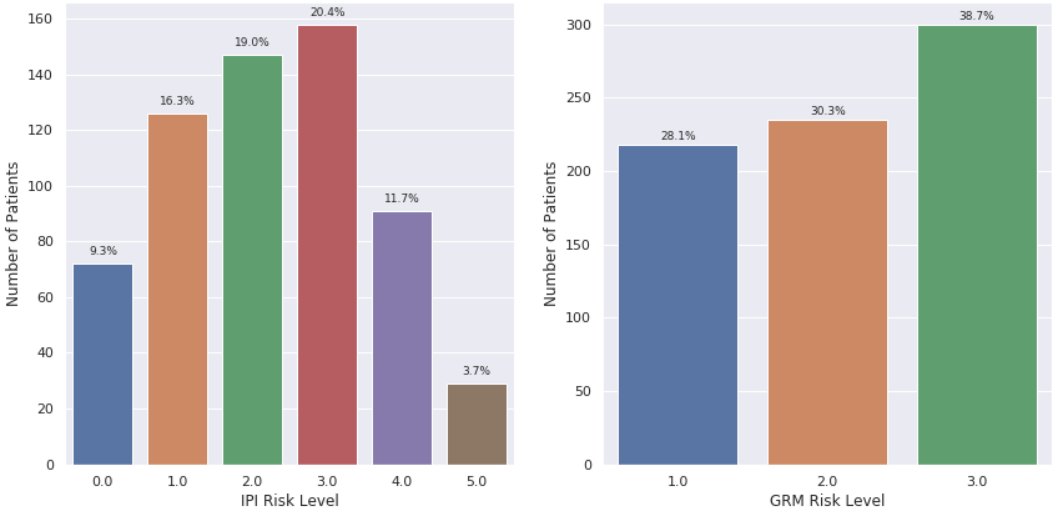


Figure 4.3: Bar charts for IPI and GRM distributions after pre-processing

Both IPI and GRM are risk assessments indicative of how severe a DLBCL case is. The GRM was represented as categorical strings but, like the IPI, encodes ordinal values and so we encoded low risk=1; medium risk=2; high risk=3. These numerical values make future analysis easier as they can be better integrated into our models. As can be seen from Table 4.1, IPI had the most missing values by a large margin - we used MICE to first impute GRM values and then used MICE with the imputed GRM values to impute IPI missing values since they both encode similar information.

It is interesting to note, however, that the distributions of the IPI and GRM are not exactly the same as shown by their joint distribution in Fig 4.4. While the IPI vaguely resembles a normal distribution with the majority of patients falling in the middle categories, the GRM is trimodal and with a slight skew towards the upper risk level. The upper risk level in GRM overlaps mostly with IPI level 3 values indicated by the darker blue color.

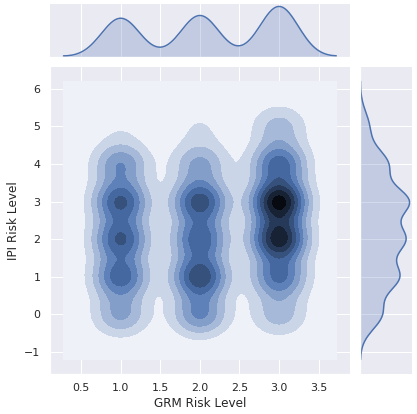


Figure 4.4: Joint plot of IPI and GRM

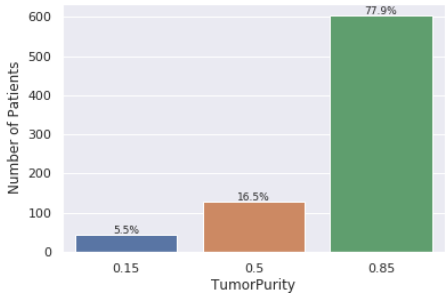


Figure 4.5: Bar chart showing distribution of tumor purity values after pre-processing

The tumor purity had no missing values and takes ordinal values encoding 3 ranges: 0-30%, 30-70%, and 70-100%. We encoded these three categories as taking the following values: 0.15, 0.5, and 0.85. The *GDR* substitution method was also considered but, after discussing with the clinicians, we decided to leave it as an ordinal variable.

MICE was used to impute the missing values in overall survival years while the censored values were all set to 0 since the exact death times were not observed.

Since we are evaluating GADD45B gene's role, we add it to our clinical features from the expression data that we describe in the next section. No pre-processing is required. Table 4.2 summarises our final clinical data for the 775 patients (each patient has a `SampleID` for identification):

Variable Name	Description	Data Type	Data Range
International Prognostic Index	A risk assessment to predict survival outcomes for DLBCL patients.	Ordinal (n=6)	0|1|2|3|4|5
Genomic Risk Model (GRM)	A novel risk assessment from *Reddy* to predict survival outcomes for DLBCL patients.	Ordinal (n=3)	1|2|3
Age at diagnosis	Age of patient at diagnosis.	Continuous (1 d.p.)	3.1-93.4
Response to initial therapy	Percentage *decrease* in patient's tumor following treatment.	Continuous	0-100
ABCGCB	COO classification of patient's DLBCL.	Categorical (n=3)	ABC|GCB| Unclassified
Overall survival years	Time in years from first recorded treatment until death from any cause.	Continuous (2 d.p.)	0.01-12.61
Censored	A survival time is censored if the exact death time is not observed.	Binary	0 (censored) | 1 (uncensored)
Tumor purity	Percentage (as decimal) of patient sample that is cancerous.	Ordinal (n=3)	0.15|0.5|0.85
GADD45B	The GADD45B gene expression level of a patient.	Continuous (15 d.p.)	0.3-11.5 (1 d.p.)

Table 4.2: Final Clinical Features

4.2.2 Gene Expression Data

Details & Feature Selection

The gene expression data contains the mRNA levels of 13,128 genes for the 775 patients, indexed by the patient ID. We removed 6 genes which had null columns and then GADD45B since it is used as a clinical variable, leaving us with 13,121 genes. The remaining gene expression levels range from 0.321 - 16.7 (3 s.f.). Crucially, the data includes genes present in the different signatures of the NF-κB pathway which we investigate in the next section. We now apply our two feature selection methods to extract a smaller set of most informative genes.

One of our objectives is to keep the analysis as data-driven as possible and not include any prior biological information. Keeping with this aim, we purposefully do not remove any well-known *housekeeping genes*. These are genes that appear frequently in omics data because they are involved in many standard biological processes. Instead, we first use one of our feature selection methods - variance thresholding - to remove genes that do not vary a lot across patients.

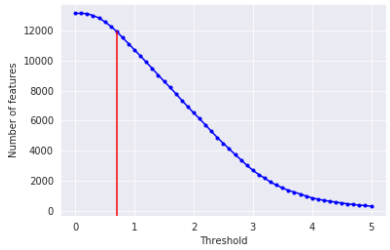


Figure 4.6: Number of selected genes versus variance threshold (for 13,128 genes).

From Fig 4.6, we can see that the number of selected genes starts decreasing linearly at a threshold of 0.70 as indicated by the red line. We make this our threshold to get rid of the minimally-variable features. This preserves 11,928 out of 13,128 genes. We now turn to Laplacian Score (LS) filtering to get a smaller set of genes that best preserves the structure of the data.

We use the `skfeature` package [31] to perform LS filtering using the Euclidean distance metric. Recall that there are two hyperparameters t and k that we set. At $t > 50$, the overall trend of the Laplacian scores becomes clearer and increasing k only highlights this trend more clearly. The results for different t and k can be seen in the appendix. In Fig 4.7, we display the LS graph with $t = 500$ and $k = 20$.

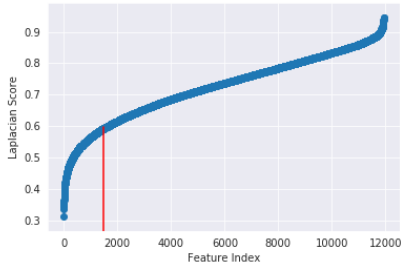


Figure 4.7: Laplacian score (LS) on 11,928 genes (sorted in ascending order for the Laplacian score).

The LS starts to increase linearly when the number of best genes increases beyond 1,500, indicated by the red line on the graph. This means that 1,500 genes are sufficient for preserving the key dynamic of our gene expression data's structure. **These 1,500 genes will be used in the NMF clustering algorithms**.

4.3 Initial Data Analysis

This project investigates the ability of using GADD45B gene expression as a prognostic factor and its ability to stratify patients. For this reason, most of our statistical analysis focuses on GADD45B gene expression and the existing COO stratification. Occasionally in Cox PH models, we use other relevant clinical variables for comparison of prognostic factors (e.g. GADD45B gene's effect on survival compared to GRM). We perform our analyses on patients groups with **low purities** (<70% tumor purity), **high purities** (>70% tumor purity), and **all purities** to see if tumor purity affects the results.

Firstly, we compare the relative GADD45B gene expression levels at a significance level of 0.025. We use the independent two-sample Student's (ST) t-test and Welch's t-test for pairwise comparisons which check if the means from two cohorts are different. The ST t-test assumes that the two populations being compared have normal distributions with equal variances. Welch's t-test is less strict: it assumes normality but does not assume equal variances. The true population variances are unknown and so we use both tests here. To compare COO categories, we use the ANOVA one-way test with null hypothesis: samples are drawn from populations with the same mean. Results are reported in Table 4.3 with significant values highlighted in red and values near to our significance level highlighted in orange. The 'all purities' boxplot is displayed in Fig 4.8 while the 'low' and 'high' purity boxplots are in the appendix.

COO Cohort Combination	Low Purities	High Purities	All Purities
ABC vs GCB	ST: 0.279 Welch's: 0.303	ST: 0.040 Welch's: 0.041	ST: 0.02 Welch's: 0.019
ABC vs Unc	ST: 0.509 Welch's: 0.484	ST: 0.992 Welch's: 0.992	ST: 0.747 Welch's: 0.754
GCB vs Unc	ST: 0.203 Welch's: 0.140	ST: 0.129 Welch's: 0.164	ST: 0.042 Welch's: 0.043
ABC vs GCB vs Unc	ANOVA: 0.310	ANOVA: 0.093	ANOVA: 0.028

Table 4.3: P-values of similarity tests for GADD45B gene expression between COO categories.

These tests show that GADD45B gene can stratify between the main COO categories ABC and GCB but not convincingly for any other cohort pairs. For the ABC/GCB comparison, the results support the earlier analysis of low purity data producing opposite results when compared to high and all purities samples- we see that significant differences in the 'all purities' and 'high purities' dataset is not observed at 'low purities' samples. The three-way comparison yields a potential indication of GADD45B gene expression difference between the three categories, but, again, nothing conclusive. To summarise, we see that ABC patients express GADD45B at slightly higher (but statistically not significantly different) levels than GCB.

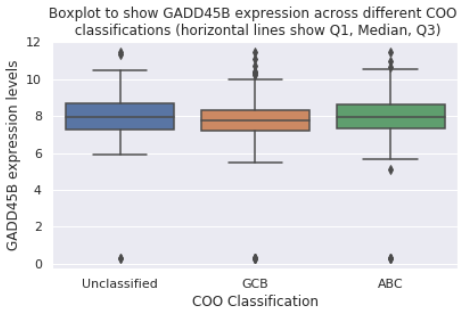


Figure 4.8: Boxplot of GADD45B levels for different COO classifications at all tumor purities.

The next step is to look at correlations between activation of the NF-κB pathway and GADD45B gene expression to check more rigorously whether the cancer utilises this mechanism to proliferate. Biological pathways are mechanisms associated with a set of genes known as a signature. We were provided with eight main NF-κB gene signatures associated with GADD45B gene, three of which have been highlighted by the clinicians for biological relevance: the *Staudt* (69 genes), *Annunziata* (8 genes), and *Broyl* (42 genes) signatures.

All the genes for the *Annunziata* signature are present in our cohort for gene expression data, and we selected this signature for analysis here (Fig 4.9). Plots for the other two signatures can be found in the appendix and their results are briefly discussed here. We report the mean gene expression value per signature and study its relationship to GADD45B gene expression in each patient using scatter plots. Relationships are measured through Pearson's correlation coefficients (r), reported on each graph.

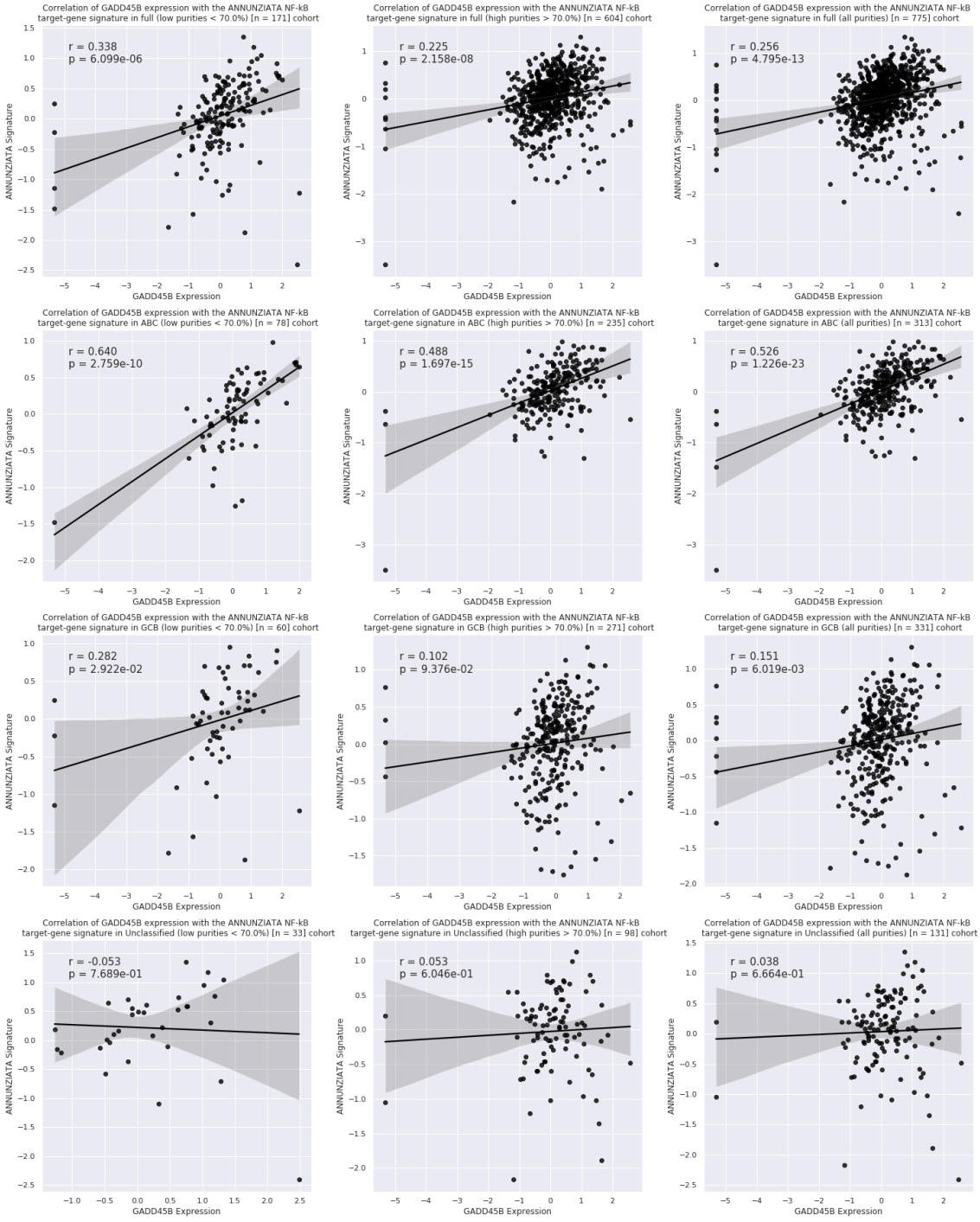


Figure 4.9: Correlations between *Annunziata* gene signature and GADD45B gene expression level at different purity levels and for different COO categories. Columns (left to right)=low purities, high purities and all purities; Rows (top to bottom) = all, ABC, GCB, Unclassified categories.

Looking at purity effect in the all, ABC, and GCB cohorts (top three rows), it seems to have a minor effect on correlation values. In the unclassified category however, the three purity levels agree with each others and all return very low correlation values. This suggests that taking into account samples with all purity levels is a safe option to preserve sample size.

Looking at 'all purities' (right column), the correlation r is $+0.526$ for the ABC cohort and $+0.151$ for the GCB cohort and both of these are different to the full cohort which has $+0.256$ correlation. The boxplots recapitulate an underlying difference between the ABC and GCB cohorts (middle rows) based on GADD45B gene expression levels and this can be inferred in the differences between their NF-κB correlation plots too. As expected, the Unclassified cohort (bottom row) shows very little correlation with the gene signature. Very similar patterns are seen in the *Staudt* signature. Surprisingly, category effects are seen in the low purity level for the *Broyl* signature but not as clear or as in the high or all purity cohorts. Consequently, we conclude that there is sufficient evidence to believe a large positive association exists between the GADD45B gene expression and NF-κB pathway in the data, indicating that GADD45B gene expression is tied to a cancer mechanism rather than just a byproduct of other biological processes.

Finally, we investigate the effect of GADD45B gene expression and other clinical variables on survival. We visualise Cox PH models for the full cohorts in Fig 4.10. As described in the 2.2.2, we plot the hazard ratios of prognostic factors side-by-side with the population Cox survival plots when varying one variable - here GADD45B gene expression level. In the high and all purities cohorts, we see that the hazard ratios for IPI and GADD45B gene expression are greater than one and so both decrease chances of survival as they increase. Strangely, GRM increases chances of survival as it increases - which is counter-intuitive to what was originally thought about IPI and GRM encoding the same risk information.

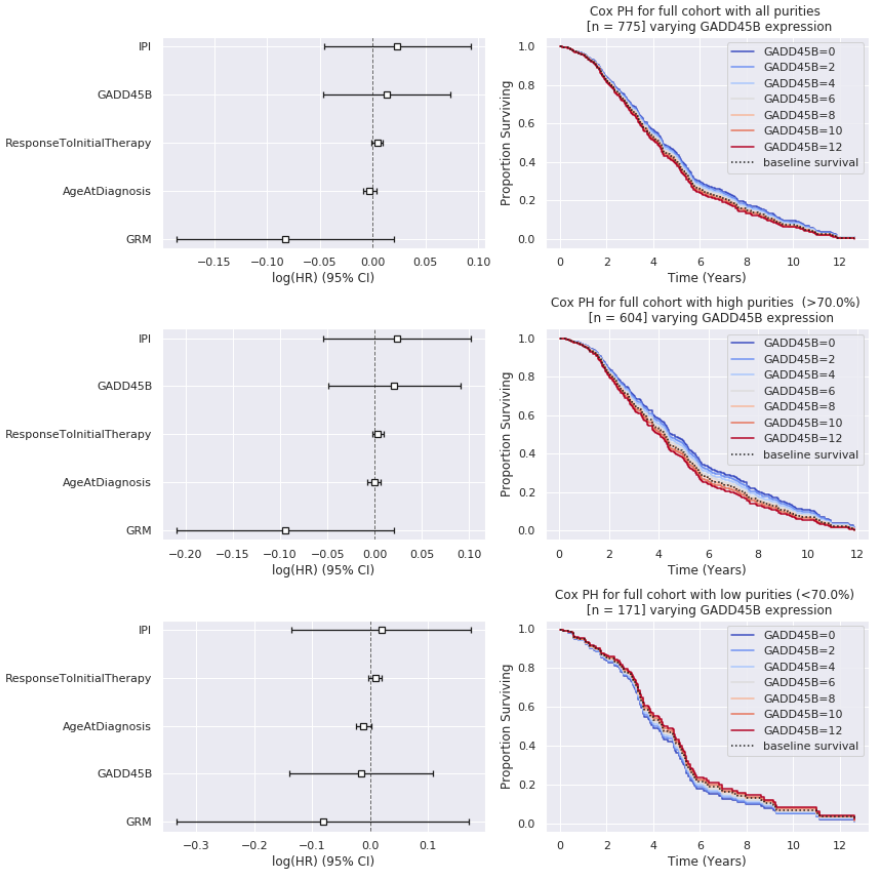


Figure 4.10: Hazard ratios and Cox PH plots versus GADD45B gene expression on the full cohort at different purity levels.

Turning our attention to purity, we see that the low purity cohort shows that increasing GADD45B gene expression level has an opposite effect on patients and increases their chance of survival. This is contrary to the well-researched biological description of GADD45B gene expression and our observations on high purity samples. This can be visualised clearly by the switch of colour progression in the high and all purity cohorts compared to the low purity one. In other words, the low purity data produces an inverse signal to what is seen overall, giving some credential to low purity samples being less indicative of the true relationships underlying the biological data.

We performed the same analysis on the COO cohorts. The plots for these can be found in the appendix but we describe the main results here. The GCB cohort shows a similar pattern to the full cohort with the low purity samples showing an inverse signal for GADD45B gene expression. The ABC cohort, as expected, showed that increasing GADD45B gene expression correlates to a lower chance of survival - which is backed up by the signature correlation results which tell us that the NF-κB cancer pathway is more active with higher GADD45B gene expression levels in our cohort. The Unclassified cohort was the exception with increased GADD45B gene expression level leading to a higher chance of survival. Summarising, we find that high GADD45B gene expression level and high IPI levels generally indicate a lower chance of survival, while increased GRM indicates a higher chance. Response to initial therapy and age at diagnosis were not found to be of prognostic value for survival.

We want to identify a more significant stratification of patients based on GADD45B gene expression level. The inconsistent results shown for the COO classification motivate us to explore different patient (and gene) stratifications - as such, we now turn our attention to clustering and creating our own subgroups of patients. Following our inconsistent results with lower purities, we also focus solely on the "high purities" and "all purities" subgroups.

Chapter 5

Evaluation

5.1 Evaluation Method

The aim of the project is to keep the initial analysis as data-driven as possible without relying on biological information for subgroup creation. While there is no fundamental right-or-wrong answer, we use the evaluation strategies from *Reddy*, *Schmitz*, and *Chapuy*, to try and create the most clinically and biologically relevant clusters. We describe our evaluation pipeline in three stages:

1. The first stage is a statistical evaluation which looks at properties of the clusters produced by the different NMF clustering algorithms and helps us select the best rank (number of clusters) for each one. We call this stage *Intrinsic Evaluation*.
2. The second stage, *Clinical Analysis*, has two parts. The first part evaluates the GADD45B and NF-κB stratification in subgroups generated by the different algorithms. The most distinct GADD45B stratification is taken forward into the second part by evaluating the subgroups from a survival standpoint.
3. The third and final stage, *Biological Analysis*, helps determine biological networks present in each subgroup by performing gene clustering.

5.2 Intrinsic Evaluation

In this section we detail our chosen metrics and present our results. We evaluate each algorithm on a set of decomposition rank values (i.e. different number of clusters) to see which rank produces the most stable and reproducible clusters while maintaining an accurate description of the original data.

5.2.1 Metrics

It is important to have consistent comparison across the different tests. Here, we use four metrics that measure important qualities we want in our generated clusters: stability, reproducibility, and accurate matrix factorisation. As a reminder, the consensus matrix $\bar{\mathbf{C}}$ is the average of connectivity matrices produced over multiple NMF runs.

Cophenetic Correlation Index

The cophenetic correlation index (CCI) proposed by Brunet et al. [22] is computed as the Pearson correlation between two distance matrices and measures how well hierarchical clustering on a consensus matrix preserves the original pairwise distances between the samples. The metric is defined as follows:

$$CCI = \frac{\sum_{i<j}(D(i,j) - \bar{d})(T(i,j) - \bar{t})}{\sqrt{(\sum_{i<j}(D(i,j) - \bar{d})^2)(\sum_{i<j}(T(i,j) - \bar{t})^2)}} \tag{5.1}$$

where:

- $D = 1 - \bar{\mathbf{C}}$ is the symmetric distance matrix such that $D(i,j) = 0$ if patients i and j belong to the same cluster else $D(i,j) = 1$.
- T is the symmetric matrix which measures the cophenetic distance between samples. As such, $T(i,j)$ is the cophenetic (dendorgrammatic) distance between samples.

- $\bar{t}$ is the mean of the upper diagonal elements of T.
- $\bar{d}$ is the mean of the upper diagonal elements of D.

The CCI metric takes values in the range $[-1, 1]$ and measures the stability of the produced clusters from a hierarchical clustering perspective. We can visualise this stability more clearly with a heatmap where higher stability corresponds to more distinct boundaries between squares (clusters). We illustrate this idea in Fig 5.1.

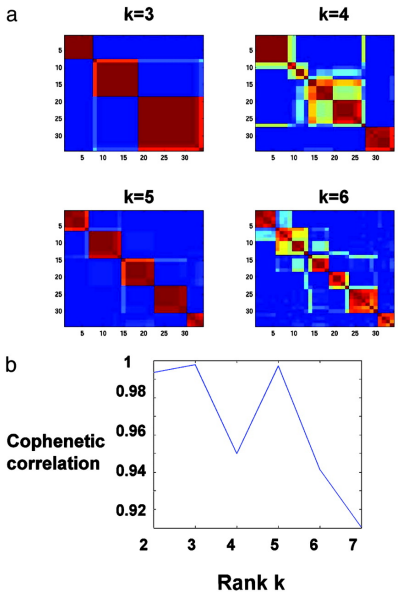


Figure 5.1: Heatmap encoding the cophenetic correlation metric between distance matrices [22]. (a) Distance matrices running consensus NMF with different rank values. (b) Cophenetic correlation values versus rank.

Dispersion Index

The dispersion index (DI) proposed by Kim et al. [32] is also based on the consensus matrix. It measures the reproducibility of the obtained clusters by measuring the deviation of the consensus matrix from 0.5 in each element (i.e. random chance of two elements being in the same cluster). The DI measure is defined as:

$$DI = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} 4(C_{ij} - 0.5)^2 \tag{5.2}$$

The DI measure takes values in the range $[0, 1]$. In essence, DI measures the similarity between connectivity matrices produced in each NMF run. In terms of heatmap visualisation, DI is similar to CCI. If the DI is closer to one, it means that the consensus matrix clusters have more defined boundaries, whereas if the DI is closer to zero, it means that the cluster boundaries are less reproducible.

Residual Sum of Squares & Rank Balance

The residual sum of squares (RSS) measures the difference between our factorisation **WH** and the original matrix **X**. Naturally, as we increase the rank, the latent components will have higher dimensionality and thus be able to recreate the original **WH** matrix better. There is a trade-off between increasing the dimensionality too much to the point of overfitting and decreasing it to the point where the factorisation doesn't accurately summarise the data. For this reason, we employ an additional metric to take the rank into account and balance the RSS which we call the rank balance (RB).

Combined Metric

We use a combined metric (CM) to make comparison between algorithms easier and prioritise the more important metrics.

CCI and DI measure similar properties. We choose to include both because since subtle differences between them allow us to balance two properties of our method. The DI measures the stability as a process, looking at the similarity of each connectivity matrix - this is what we define as reproducibility. The CCI, on the other hand, measures stability at the end of the process by viewing it through a hierarchical clustering lens. These are the most important metrics for us and we give them equal weights to enforce the importance of clustering stability across the whole process.

Next, we consider the RSS and RB. Although high-fidelity with respect to the original data matrix is important, we see (later on) in Fig 5.2 that the RSS does not vary too much across different ranks using the standard NMF algorithm and so we give it less importance. As mentioned before, the RB is used to balance out the RSS and so we give both metrics the same weights. We also transform the RSS and RB so that they take values in similar ranges to the CCI and DI. The RB is calculated for each rank divided by the max rank. So for ranks in [4, 5, .., 10], the RB values are in [0.4, 0.5, .., 1.0]. We do a similar rescaling for the RSS measure, dividing by the max RSS value.

Our final metric is defined as:

$$CM = 0.5(CCI + DI) - 0.1(RSS + RB) \tag{5.3}$$

Overall, the higher the CM metric, the better the quality of the consensus NMF decomposition for the purpose of clustering.

5.2.2 Hyperparameter Tuning

Choosing Number of NMF Decomposition Runs

Consensus clustering averages the results across a number of runs. To set up the experiment, we had to determine how many runs were necessary for stable results so this could be kept constant across all algorithms. We were not concerned about cluster quality, instead focusing on ensuring accurate factorisation of the data matrix. Hence, we only evaluate the RSS here, disregarding CCI, DI, or RB. We ran the standard NMF consensus clustering algorithm for a number of runs in $[5, 10, .., 50]$ and rank values in $[4, 5, .., 10]$, averaging RSS values over rank values for our final plot in Fig 5.2.

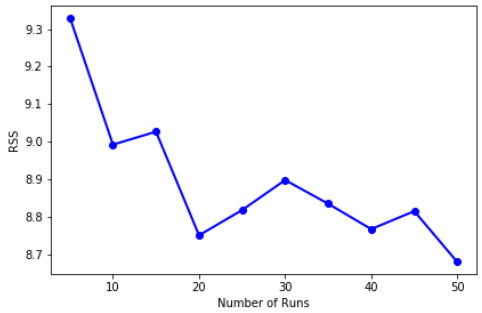


Figure 5.2: Average RSS measures versus number of runs, using the NMF decomposition algorithm on 775 patients (using all genes), averaging over rank values in 4-10.

Surprisingly, the average RSS across all ranks are reasonably similar. The RSS doesn't vary too much across the whole range but we start to see values stabilise after 20 runs. We settled on a standard number of 40 runs for each algorithm to allow for adequate stabilisation leeway.

Algorithm-Specific Hyperparameters

There are no other hyperparameters in the consensus NMF or PNMF algorithms. Both the GS and ICM algorithms have two sparsity-based parameters that allow to not sample from specific columns or rows when updating the priors (similar to drop-out in neural networks) to avoid overfitting. We chose not to employ the sparsity-based parameters since we had already filtered down our genes sufficiently with variance thresholding and Laplacian score filtering to tackle noise.

5.2.3 Results

We first display the individual metric scores, then find the best rank values for each algorithm using our combined metric, and finally qualitatively evaluate clustermaps. We refer to NMF and PNMF as non-Bayesian algorithms (NBAs), and to GS and ICM as Bayesian algorithms (BAs).

Individual Metrics

Metric values versus rank for CCI, DI and RSS are shown in Fig 5.3 for the four consensus NMF decompositions being tested.

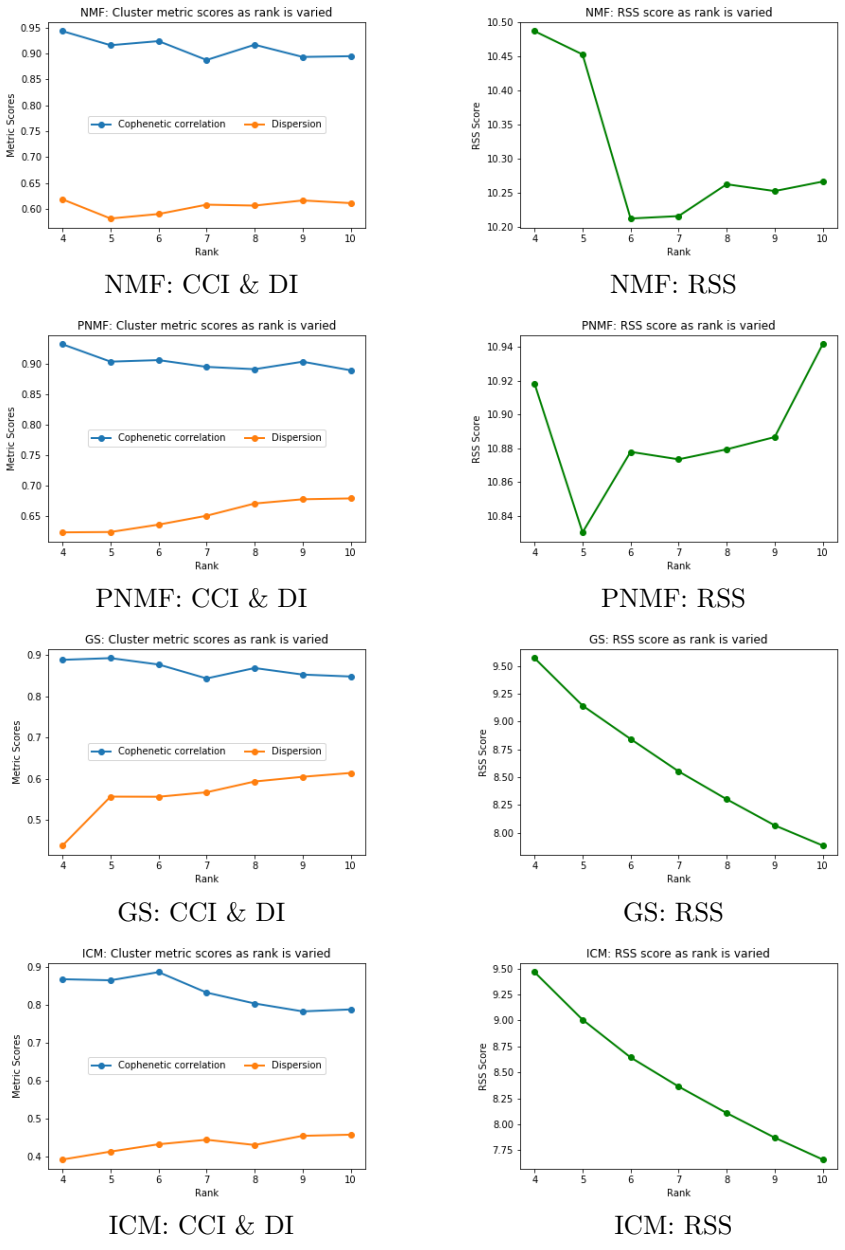


Figure 5.3: Quality of consensus NMF decomposition versus rank values using 3 individual metrics (CCI, DI, RSS) and 4 different NMFCC algorithms on 775 patients with 1,500 genes.

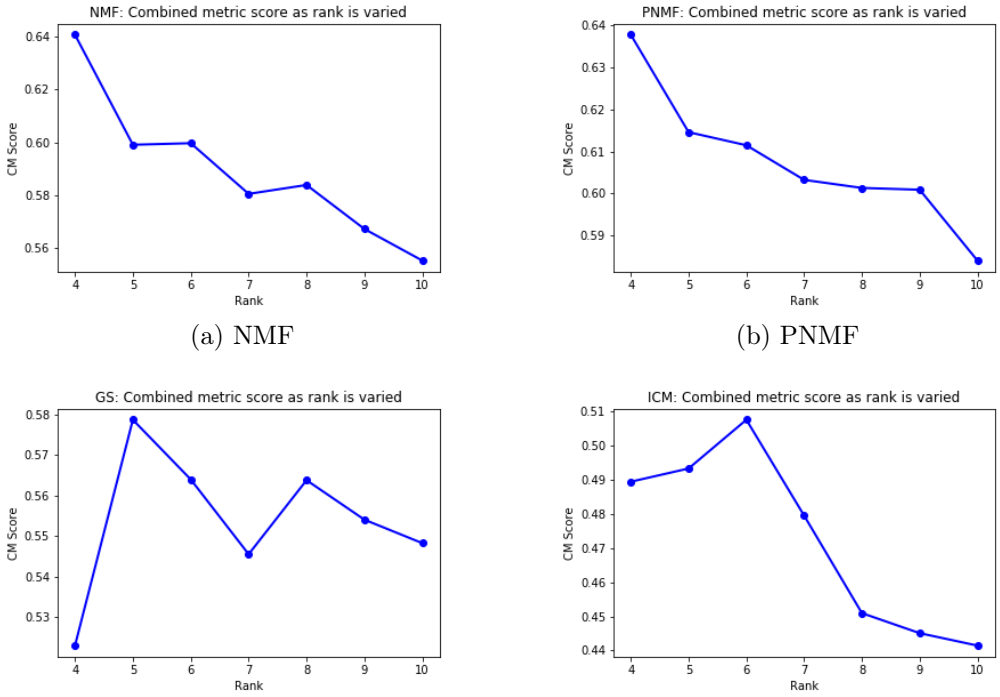
With the exception of the DI with the GS algorithm, we see that the CCI and DI generally vary minimally across ranks. There are, however, two subtle trends. Across all algorithms as the rank

increases, the CCI worsens and the DI improves. This is surprising as we would expect stability and reproducibility to follow the same trends. It also suggests that reproducibility (measured by the DI) may improve at higher ranks with smaller average cluster sizes - an unlikely result since we would expect noise to play a bigger role in smaller clusters and reduce reproducibility. Looking at the y-axes of all the graphs, we observe that the PNMF algorithms produce the best results in terms of DI, and NMF performs the best on the CCI measure.

For the BAs, the RSS decreases as rank increases as expected. For the NBAs, the RSS decreases and then jumps up again. If we look at the y-axis for these graphs, the range of RSS values is very small - this is most likely an indication that after 40 individual runs, the RSS of both algorithms plateaus between 10 and 11. Overall, the BAs outperform the NBAs using this metric.

Combined Metrics

We display the combined metric scores for our 4 tested consensus NMF decomposition algorithms across rank values in Fig 5.4.

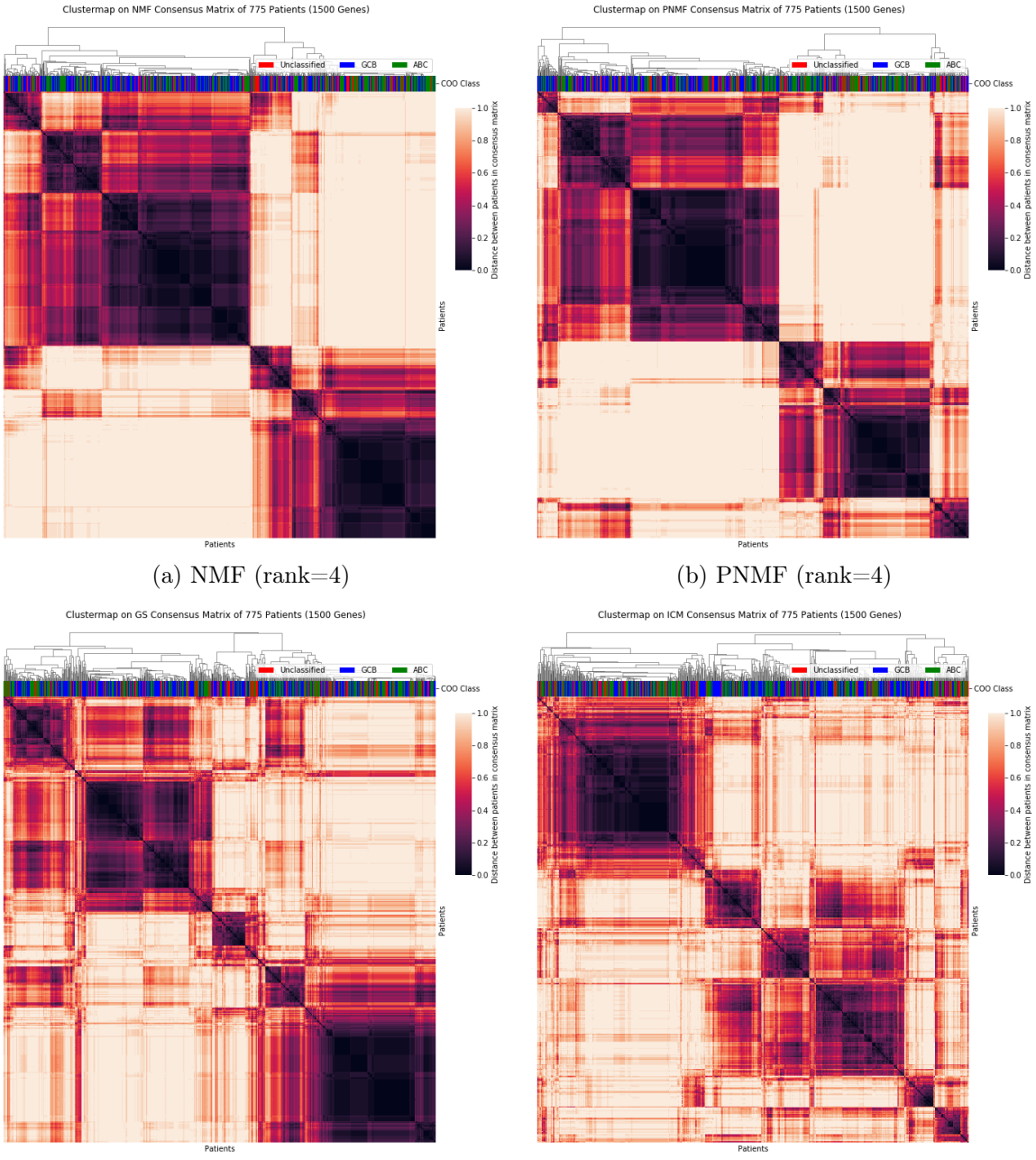


(a) NMF (b) PNMF

(c) GS (d) ICM

Figure 5.4: Our proposed 'combined metric' (CM) score versus rank values for the 4 tested NMFCC algorithms on 775 patients with 1,500 genes.

Looking at the CM score values (y-axis) for each algorithm, best scores are in the range [0.508-0.641]. The best performing algorithm is the standard NMFCC at rank 4 with a score of 0.641. We see that smaller ranks return the best scores across all algorithms with the best results generally having a rank less than or equal to 6. As we saw in the RSS graphs, the non-Bayesian methods differ in trends from the Bayesian methods. The non-Bayesian algorithms follow very similar curves, peaking at rank 4 and following almost identical trajectories (in shape and magnitude) downwards as rank increases. The two Bayesian approaches also show similar trends: they start lower, peak at a rank in the middle, and then decrease to a point where they start to stabilise. However, the ICM seems to fare poorly at higher ranks as we see a drastic drop of 0.059 from rank 6 to 8. We finish the intrinsic evaluation by looking at the clustermaps for each algorithm at its best rank according to the CM score in Fig 5.5.



(a) NMF (rank=4)

(b) PNMF (rank=4)

(c) GS (rank=5)

(d) ICM (rank=6)

Figure 5.5: Clustermaps from the NMF consensus matrices generated by the 4 tested algorithms using best rank according to the CM. Colour coding is provided for the patients' COO classification.

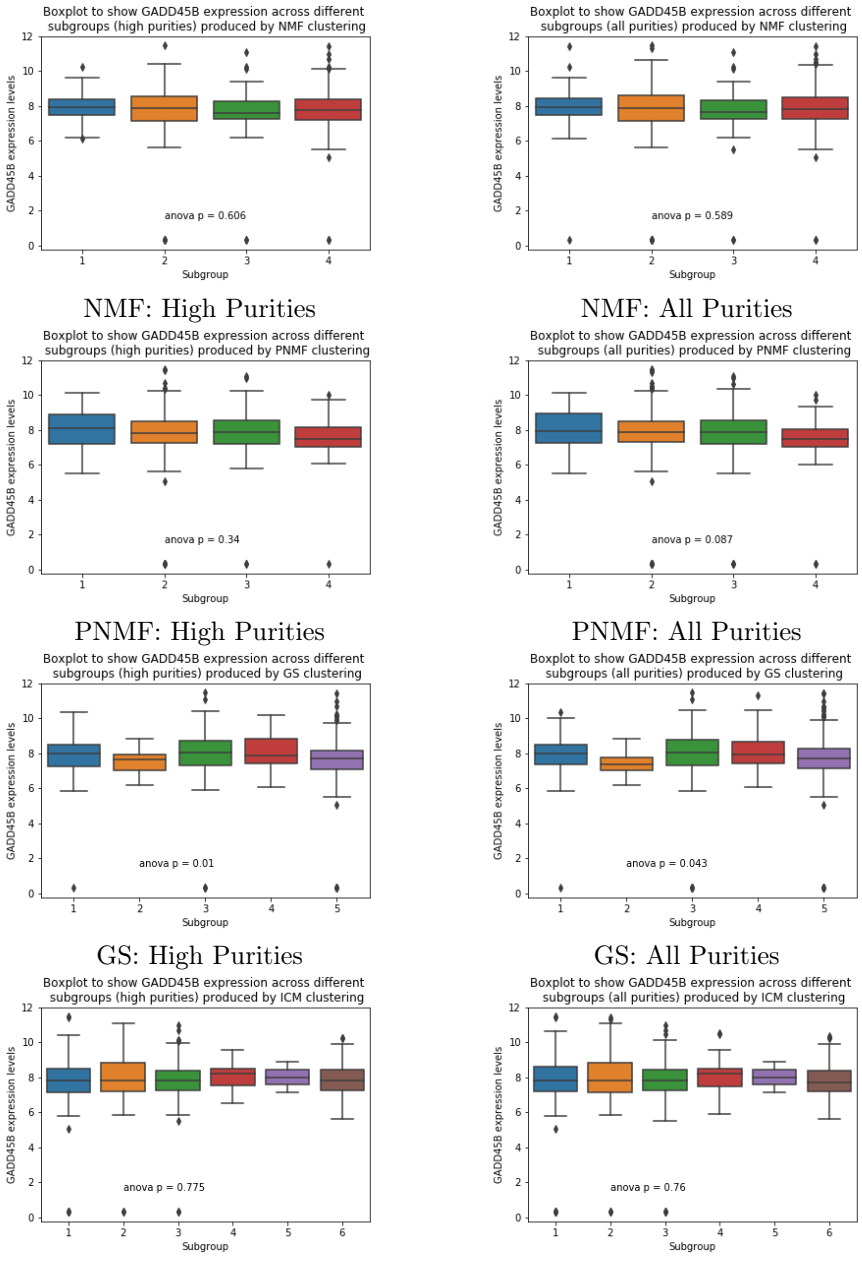
The NMF and PNMF clustermaps produce clearer distinctions and so increased stability between *larger* clusters than the GS and ICM clustermaps. However, while the NMF and PNMF distinguish more clearly between two major subgroups, their clustermaps are slightly misleading since they don't distinguish as well *within* these subgroups. For example, we can see there are two large purple squares (clusters) in the top left and bottom right of the NMF clustermap, but the rank is 4 and so we should have four distinct clusters. Contrasting this to the GS clustermap with rank 5, even though there is more noise in the overall graph, visual depiction of five clusters along the diagonal is easier. Looking at the colour bars above the heatmaps, we can see that no clusters have a distinct COO (ABC/GCB) type - this tells us that we most likely deal with different underlying biology than the well-studied COO classification.

5.3 Clinical Analysis

Having evaluated our 4 algorithms at their best rank and found a set of patient subgroups for each one, we turn to clinical analysis to find the best stratification (and algorithm). We take the best stratification forward for further clinical analysis.

5.3.1 GADD45B Stratification

The boxplots of GADD45B gene expression levels within each subgroup shown in Fig 5.6 are used to check if there is a significant difference between groups. The left column shows results on high-purity samples only (>70%) [n=604] and the right column shows results on all purities (i.e all samples) [n=775].



NMF: High Purities | NMF: All Purities

PNMF: High Purities | PNMF: All Purities

GS: High Purities | GS: All Purities

ICM: High Purities | ICM: All Purities

Figure 5.6: Boxplots of GADD45B gene expression levels within subgroups generated by the 4 tested NMF clustering algorithm ran with their 'optimal' rank values.

Included Purities	**NMF**	**PNMF**	**GS**	**ICM**
High Purities	0.606	0.34	0.01	0.775
All Purities	0.589	0.087	0.043	0.76

Table 5.1: ANOVA p-values measuring whether clusters show different GADD45B gene expression levels at a significance level of 0.025.

As for COO classification, most identified clusters do not show clear differences in average GADD45B gene expression levels using the ANOVA test (Table 5.1). If we are able to stratify on GADD45B gene expression level, this gives a strong indication that the generated patient subgroups have not been created by random chance or because they "cluster well", but actually have clinical relevance. From our results, the only algorithm that has this quality is the GS. Referring back to the clustermaps (Fig 5.5), we made the point that the NMF and PNMF methods were better in terms of cluster stability and reproducibility, but lacked the ability to distinguish more finely between groups. We can see this more clearly in terms of the GADD45B gene expression level stratification.

Unfortunately, it is the nature of this problem that we have to consider just one gene as a clinical variable, irrespective of the other 13,120 that are originally present. This means that even if just the GADD45B gene has been corrupted by biological or experimental noise, our evaluation is greatly affected. Thus, to get a better idea of whether our GS clusters are clinically relevant, we also investigate correlation of GADD45B gene expression level in each subgroup with their *Annunziata* signature to see if there are noticeable differences. Again, we include the comparison of high and all purities. We display the correlation plots in Fig 5.7 and the Pearson correlation values in Table 5.2.

Included Purities	**SG1**	**SG2**	**SG3**	**SG4**	**SG5**
High Purities	0.316 [n=97]	-0.062 [n=10]	0.179 [n=184]	0.126 [n=65]	0.235 [n=248]
All Purities	0.362 [n=128]	0.023 [n=12]	0.283 [n=235]	0.021 [n=82]	0.255 [n=318]

Table 5.2: Pearson correlation coefficient values between GADD45B gene expression level and NF-κB expression across the different clusters (subgroups) generated by the GS algorithm.

Correlation p-values are significant for all subgroups except subgroup 2, most likely due to its smaller size, and so we ignore it for now. We see that there are correlation values of 0.126, 0.179, 0.235, and 0.316 for subgroups 4, 3, 5, and 1. There are clear differences of at least 0.05 between the subgroups which suggests a different level of NF-κB activation in each subgroup. These results are promising and so we continue the clinical analysis further, looking more broadly at survival.

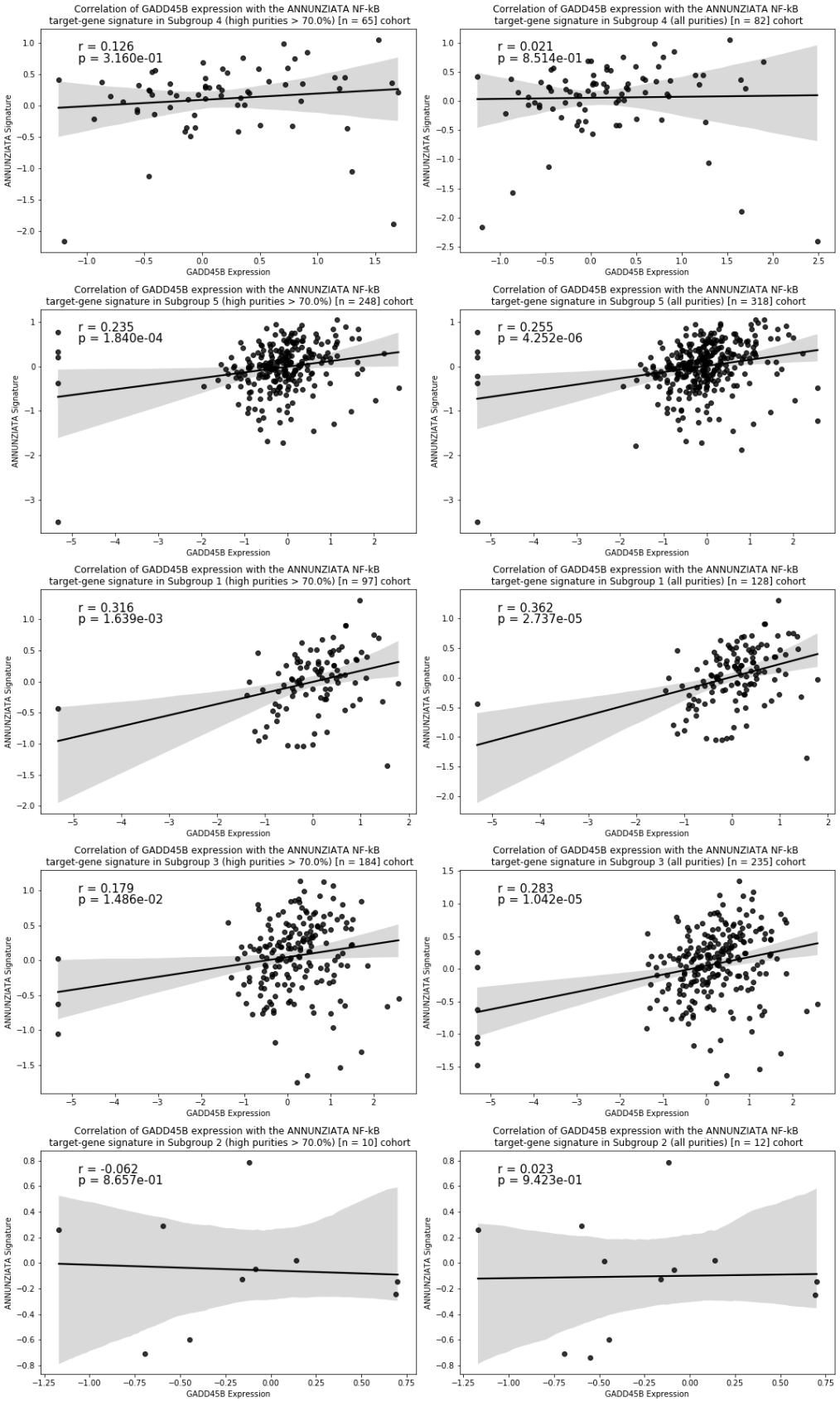
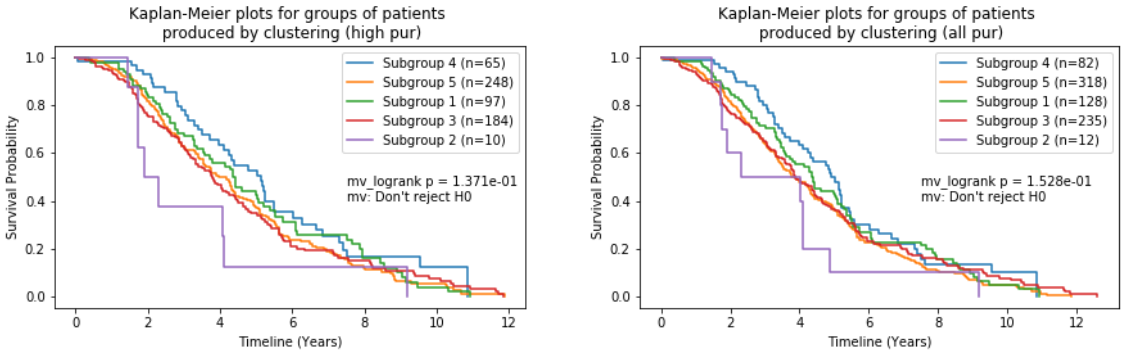


Figure 5.7: Correlation plots of GADD45B gene expression level with *Annunziata* in the subgroups derived from the GS algorithm for samples with high (left column) and all purities (right column).

5.3.2 Survival Analysis

We compare the subgroups generated by the GS algorithm at rank 5 based on their survival. Firstly, we look at Kaplan-Meier survival curves at high and all purities. The plots are shown in Fig 5.8.

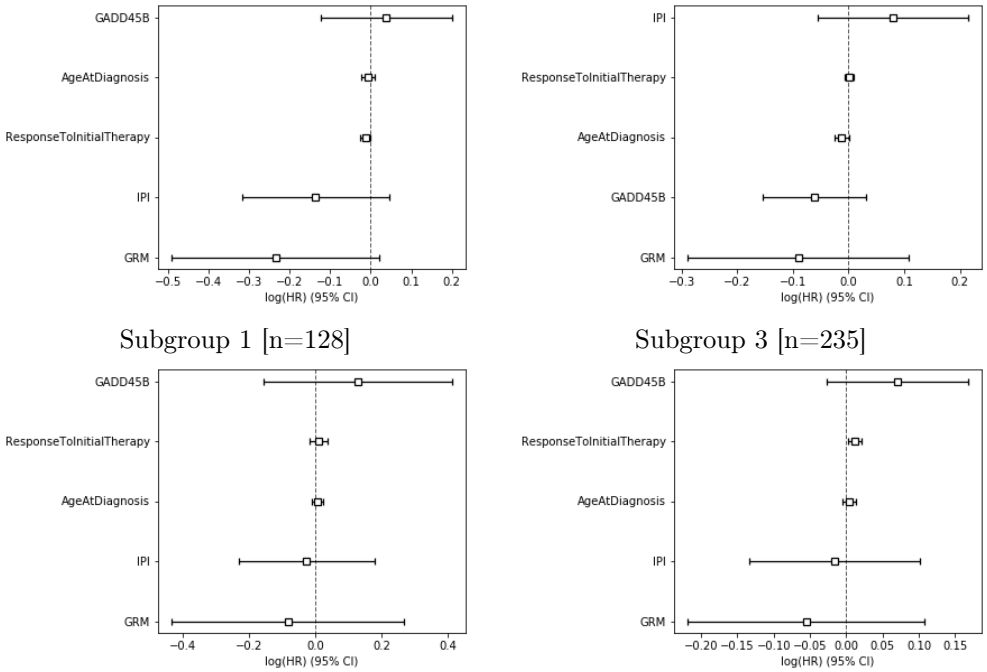


High Purities All Purities

Figure 5.8: Kaplan-Meier survival curves for each GS subgroup.

Under the null hypothesis that all the curves belong to the same distribution, the multivariate logrank test returns an insignificant p-value, indicating little difference in survival. The pairwise logrank test returns similar insignificant results with the notable exceptions of subgroups 3 versus 4, and subgroups 4 versus 5. These results being mostly inconclusive, we turn to the Cox PH model to see which clinical variables are playing important roles in survival.

Hazard ratios are very similar for all and high purities, and we choose to display results on all purities in Fig 5.9. We leave subgroup 2 out because of its small size. Hazard ratios for both purities with accompanying visualisations can be found in Appendix C.1.



Subgroup 1 [n=128] Subgroup 3 [n=235]

Subgroup 4 [n=82] Subgroup 5 [n=318]

Figure 5.9: Cox PH hazard ratios for each GS subgroup (all purities).

Looking at individual variables, GADD45B gene expression level and the GRM variables have strong prognostic power relative to the other variables. Focusing on GADD45B gene expression level, we see that it is a strong predictor (0.05<|coeff|) in subgroups 3, 4, and 5, and marginally in subgroup 1. We look at the specific coefficient values in Table 5.3.2.

Subgroup 1	Subgroup 3	Subgroup 4	Subgroup 5
0.04	-0.06	0.13	0.07

Table 5.3: Log(hazard ratio) values for GADD45B gene expression level in each GS subgroup (this is the value displayed in the hazard ratio plots).

Again, we see different levels of impact on survival from GADD45B gene expression level. The large differences between subgroups 3, 4, and 5 agree with pairwise survival curves comparison. Based on these results, we believe that our GADD45B gene expression level stratification may also have clinical significance.

5.4 Biological Analysis

The final part of the evaluation is to find the key gene networks underlying each GS subgroup. Unlike the *Chapuy* and *Schmitz* papers which use statistical tests to ascertain genetic differences between generated clusters, we take a broader approach looking at the overarching gene networks of subgroups *without* assuming any differences. We do this in the hope that common structures between subgroups can also be determined. Since our focus is on gene networks, we turn again to clustering. We look at clustering on our patient subgroups using the selected 1,500 genes. We use the same NMFCC algorithm as before: Bayesian NMFCC with Gibbs Sampling (GS).

5.4.1 Gene Scoring

We use this method to find the genes that contribute the most to each cluster (biological process) overall. Using the method outlined by Kim et al. [32], we define a gene score GI for the i-th gene as

$$GI(i) = 1 + \frac{1}{log_2(k)} \sum_{q=1}^{k} p(i,q) log_2(p(i,q)) \tag{5.4}$$

where

- k is the rank which corresponds to the number of biological processes in each subgroup. As such, q corresponds to each cluster.
- $p(i,n) = \mathbf{W}(i,n) / \sum_{q=1}^{k} \mathbf{W}(i,q)$ is the probability that the i-th gene contributes to cluster n.
- $\mathbf{W}$ is the final basis matrix produced by the NMF algorithm. $\mathbf{W}(i,n)$ corresponds to the value of the i-th gene in the n-th cluster.

The GI score lies within the range $[0,1]$. The higher the score, the more the gene impacts the different processes. Once scores have been calculated, they are ranked and chosen if they satsify two criteria:

- They are higher than $\mu + 3\sigma$ where μ and σ are the median and median absolute deviation of the gene scores.
- The maximum value in the rows of $\mathbf{W}$ influenced by the gene were larger than the median of all elements in $\mathbf{W}$.

5.4.2 Clustering on Subgroups

There are four steps to our strategy of finding key genes:

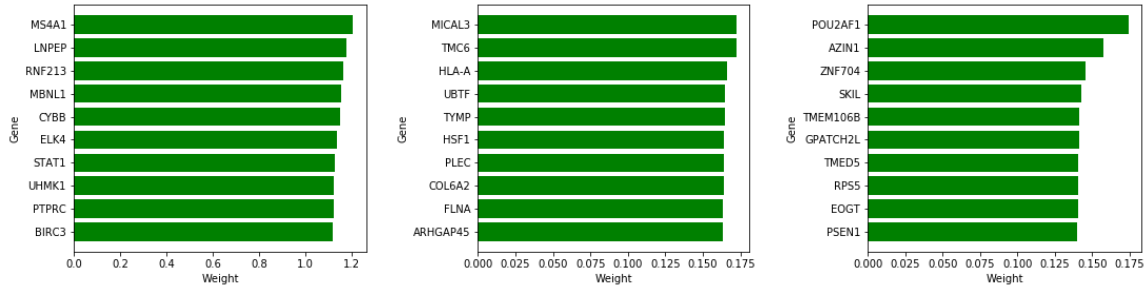
1. Identify number of processes in subgroup.
2. Identify key genes in each process.
3. Identify key genes across all processes using the gene scoring method detailed above.
4. Combine the key genes and use them in a *Coxnet* model (Cox PH with elastic net penalty, based on *Reddy*) to find the genes most associated with lower survival rates. These are the genes displayed in the hazard ratio plots.

We noticed that genes that ranked highly on the gene scoring metric were often not included in the top three genes associated with a specific biological process. For this reason, we specifically include the top 3 genes from each biological process in the initial list of genes used in the hazard ratios (note that this does not mean they will definitely show up in the final 10 key genes we have shown in the hazard ratio plots).

From the clinical analysis section, we found four large subgroups (subgroups 1, 3, 4, 5) for potential analysis. We do not consider subgroup 2 because of its size (n=12) and leave it as an "unclassified" group. We detail the full method for subgroup 1, present the results for the remaining subgroups, and then compare the results at the end.

Subgroup 1 [n=128]

We re-cluster on subgroup 1 and find the clusters are of highest quality at rank 3 using our combined metric (CM) score (Fig C.6). This corresponds to three biological processes underlying subgroup 1. We identify the metagenes associated with each process and plot them in Fig 5.10.



Process 1 Process 2 Process 3

Figure 5.10: 10 genes most associated with each process in subgroup 1.

Looking at the profiles of the top 10 genes we see that they are all reasonably similar. The lack of any distinct genes tells us there are no key genes in this subgroup, and instead, lots of genes play important roles in the underlying processes. This is confirmed by the fact that our gene scoring method returns 70 genes. We combine these 70 genes with the top 3 genes from each process and use these as the covariates in a Cox PH model. We see from the following hazard ratio plot (Fig 5.11) that the top two genes from process 3 and the second highest gene from process 1 are key prognostic factors.

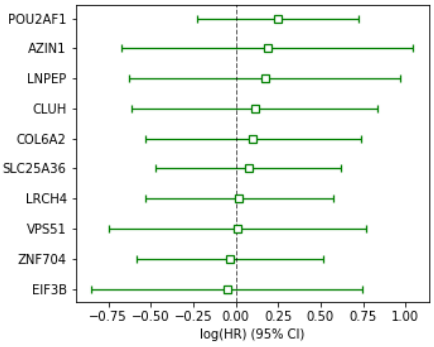
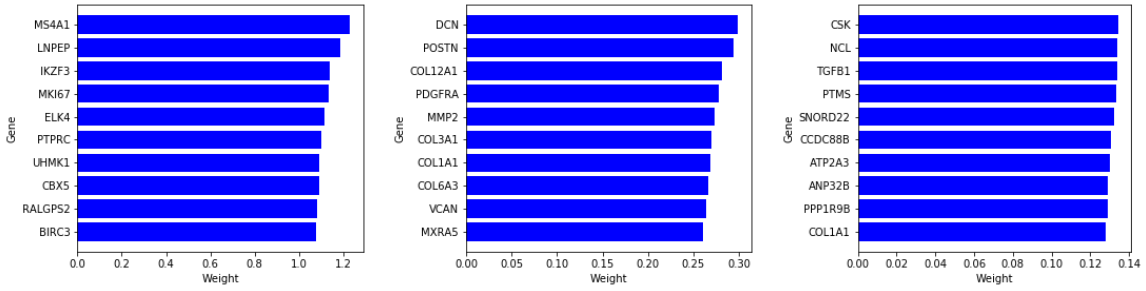


Figure 5.11: Hazard ratios for 10 genes that were most prognostic of death in subgroup 1.

Subgroup 3 [n=235]

The best quality rank was 3 (Fig C.7).



Process 1 Process 2 Process 3

Figure 5.12: 10 genes most associated with each process in subgroup 3.

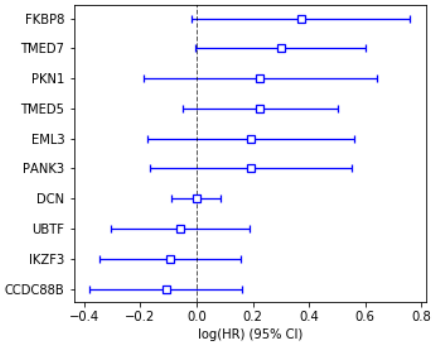
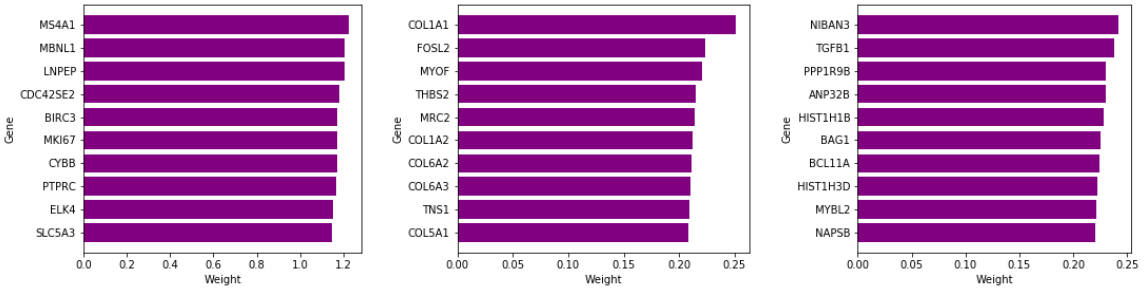


Figure 5.13: Hazard ratios for 10 genes that were most prognostic of death in subgroup 3.

Subgroup 4 [n=82]

The best quality rank was 3 (Fig C.8).



Process 1 Process 2 Process 3

Figure 5.14: 10 genes most associated with each process in subgroup 4.

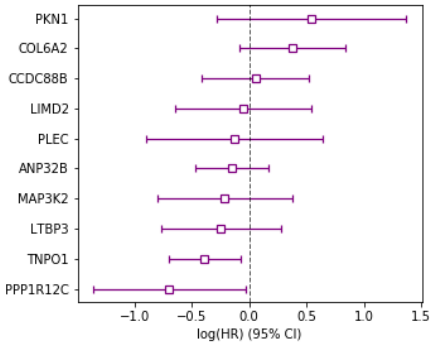
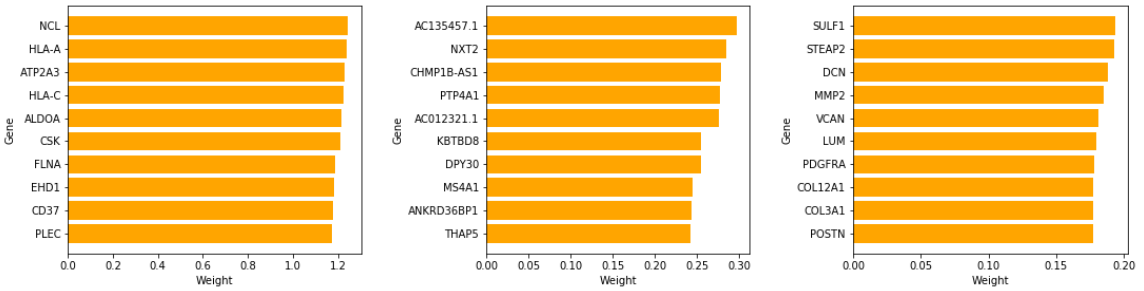


Figure 5.15: Hazard ratios for 10 genes that were most prognostic of death in subgroup 4.

Subgroup 5 [n=318]

Again, the best rank was 3 (Fig C.9).



Process 1 Process 2 Process 3

Figure 5.16: Ten genes most associated with each process in subgroup 5.

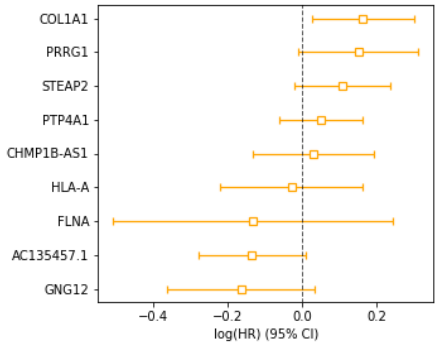


Figure 5.17: Hazard ratios for 10 genes that were most prognostic of death in subgroup 5.

Discussion

There are a few interesting observations between subgroups (SG = subgroup):

- Each of the subgroups had a best rank of 3 when using our combined metric. This points to three core biological processes in each group, regardless of size.
- The process 1 from SG1, SG3, and SG4 all share a lot of genes. All three subgroups share the ELK4, MS4A1, LNPEP, BIRC3, PTPRC genes; subgroup SG1 and SG3 share UHMK1; and subgroup SG1 and SG4 share MBNL1 and CYBB. This common process (or part of a common process) suggests that the cancer is utilising some sort of similar mechanism in all three subgroups. Strikingly, very few of these genes appear in any process in SG5.
- Looking at the weighting of the genes (x-axis), we see that process 1 has a much higher weighting than the other two processes and is likely the core process of each subgroup. This also supports the earlier point about process 1 being a key mechanism in subgroups SG1, SG3, and SG4.
- Process 2 in SG3 and SG4 share a lot of genes with the 'COL-' prefix. Genes with this prefix are part of the collagen gene family and so these two subgroups may be involved in a collagen-related pathway.
- Process 2 in SG3 and process 3 SG5 share a lot of common genes too: MMP2, VCAN, COL3A1, COL12A1, PDGFRA, and DCN.
- Finally, looking at the hazard ratios, there are clear differences between all groups. The only notable exceptions are SG1 and SG4 who share a bad prognostic gene in COL6A2, and SG1 and SG3 who share the bad prognostic gene PKN1.

Overall, our biological analysis supports the fact that there are a relatively small amount of genes (compared to the initial 13,121) that play a key factor in the tumors.

Chapter 6

Conclusion

6.1 Challenges

Throughout this project there were a few challenges. The most notable one was our difficulty in contacting the original authors of the *Reddy*, *Chapuy*, and *Schmitz* papers. Limited response from the corresponding authors of the three papers meant that we were unable to obtain data integration methods, clustering methods, and datasets that could have been useful for the project. In particular, having access to the random forest classifier used in *Schmitz* would have provided a nice comparison for the matrix factorisation algorithms used in our work.

6.2 Project Reflection

This project has performed exploratory data analysis on biomedical data from DLBCL patients using non-negative matrix factorisation methods. We have tried to identify patient subgroups that are relevant for further clinical study and targeted drug development. The project looked specifically at stratifications based on the GADD45B gene expression which is downstream product of the NF-κB pathway, a biological mechanism commonly utilised by the cancer. We have also looked more broadly at identifying biological networks present in the data and investigating the effect of tumor purity. One task which I did not complete was integrating the mutation and expression data together and clustering on both of them - in lieu of this, we briefly touch on multimodal methods in Future Work.

Keeping these objectives in mind, I think that the project has been carried out as intended and has shown potentially useful results. We were able to stratify patients effectively on gene expression data and GADD45B, and further analysis gives an indication of potential for therapeutic development. We also found some interesting properties of the underlying biology of our different patient subgroups. On the bioinformatics side, enrichment analysis is the next step to obtaining relevant results.

From our results in the initial data analysis, there is a strong indication that low purity samples produce inverse signals to the results achieved when using all the samples. It is important we mention that there are fewer low purity samples and so these adverse effects may get amplified, however, we believe there is sufficient evidence to focus on higher purity samples where possible.

6.3 Future Work

6.3.1 Short-Term

The first thing to do would be to find out whether our subgroups are functionally relevant by performing functional annotation. If we find that our gene sets are related to pathways involved in disease, we will have a solid foundation on which we can integrate other genes, aside from GADD45B, that are mediated by the NF-κB pathway. Specifically, XIAP, A20, and inhibitors of reactive oxygen species accumulation are also upregulated during NF-κB activity [33] and would be of interest. Beyond NF-κB, there are two other modules we would like to investigate in a similar way: STAT3, a pathway similar to NF-κB, and BCL-6, a transcription factor signature which is clinically relevant to lymphoma. The overarching idea is taking a systems-level approach by incorporating these different biological components into the analysis.

6.3.2 Long-Term

We made use of an important class of ensemble methods called consensus clustering throughout our project. These average out the runs from an algorithm with a stochastic property. However, ensemble methods are more powerful and can also be used to collate the results of different algorithms. This technique has been used in colorectal cancer [34] to combine subtypes produced by different groups, and would be of use to oncologists in other cancers like DLBCL too.

There are two types of machine learning paradigms that are seeing increasing popularity that we think would benefit our work: geometric learning and multimodal learning. One of the limitations of NMF is that it is a linear dimensionality reduction method and non-linear interactions between genetic components (e.g. negative feedback loops) are often observed in gene regulatory networks. Geometric learning uses graphs (which can represent non-linear relationships) which can model biological structures like NF-κB more accurately. Furthermore, as data becomes more standardised and collection technologies improve, incorporating various forms of omics and clinical data into the same algorithm will be essential. Multimodal learning could be the tool to understand this large volume of diverse data and has already shown some promise in oncology [35].

Bibliography

[1] R. Marcus, Diffuse large b-cell lymphoma, https://lymphoma-action.org.uk/types-lymphoma-non-hodgkin-lymphoma/diffuse-large-b-cell-lymphoma (Online; accessed 04-01-20).

[2] D. Pucheril, S. Sharma., The history and future of personalized medicine, https://www.managedcaremag.com/archives/2011/8/history-and-future-personalized-medicine (Online; accessed 09-01-20).

[3] C. B. F.R. Vogenberg, M. Pursel., Personalized medicine: part 1: evolution and development into theranostics, *Pharmacy and Therapeutics* 35 (10) (2010) 560.

[4] A. Reddy, J. Zhang, N. S. Davis, A. B. Moffitt, C. L. Love, A. Waldrop, S. Leppa, A. Pasanen, L. Meriranta, M.-L. Karjalainen-Lindsberg, et al., Genetic and functional drivers of diffuse large b cell lymphoma, Cell 171 (2) (2017) 481–494.

[5] B. Chapuy, C. Stewart, A. J. Dunford, J. Kim, A. Kamburov, R. A. Redd, M. S. Lawrence, M. G. Roemer, A. J. Li, M. Ziepert, et al., Molecular subtypes of diffuse large b cell lymphoma are associated with distinct pathogenic mechanisms and outcomes, *Nature Medicine* 24 (5) (2018) 679–690.

[6] R. Schmitz, G. W. Wright, D. W. Huang, C. A. Johnson, J. D. Phelan, J. Q. Wang, S. Roulland, M. Kasbekar, R. M. Young, A. L. Shaffer, et al., Genetics and pathogenesis of diffuse large b-cell lymphoma, *New England Journal of Medicine* 378 (15) (2018) 1396–1407.

[7] C. R. UK., How cancer starts, https://www.cancerresearchuk.org/about-cancer/what-is-cancer/how-cancer-starts (Online; accessed 20-01-20).

[8] A. Pietrangelo, E. Luo., Benign and malignant tumors: How do they differ?, https://www.healthline.com/health/cancer/difference-between-benign-and-malignant-tumors#key-differences (Online; accessed 01-02-20).

[9] I. Tomlinson, P. Sasieni, W. Bodmer, How many mutations in a cancer?, *The American Journal of Pathology* 160 (3) (2002) 755.

[10] I. Dagogo-Jack, A. T. Shaw, Tumour heterogeneity and resistance to cancer therapies, Nature reviews Clinical oncology 15 (2) (2018) 81.

[11] K. Strimbu, J. A. Tavel, What are biomarkers?, Current Opinion in HIV and AIDS 5 (6) (2010) 463.

[12] J. Bennett, D. Capece, F. Begalli, D. Verzella, D. D'Andrea, L. Tornatore, G. Franzoso, Nf-κb in the crosshairs: rethinking an old riddle, The international journal of biochemistry & cell biology 95 (2018) 108–112.

[13] E. De Smaele, F. Zazzeroni, S. Papa, D. U. Nguyen, R. Jin, J. Jones, R. Cong, G. Franzoso, Induction of gadd45 β by nf-κ b downregulates pro-apoptotic jnk signalling, Nature 414 (6861) (2001) 308–313.

[14] D. Capece, D. D'Andrea, D. Verzella, L. Tornatore, F. Begalli, J. Bennett, F. Zazzeroni, G. Franzoso, Turning an old gadd get into a troublemaker, Cell Death & Differentiation 25 (4) (2018) 642–644.

[15] R. Xu, Lecture notes: Estimating the survival function (April 2019).

[16] R. Xu, Lecture notes: The proportional hazards regression model (April 2019).

[17] D. D. Lee, H. S. Seung, Algorithms for non-negative matrix factorization (2001) 556–562.

[18] A. K. Jain, Data clustering: 50 years beyond k-means, Pattern recognition letters 31 (8) (2010) 651–666.

[19] T. Ronan, Z. Qi, K. M. Naegle, Avoiding common pitfalls when clustering biological data, Science signaling 9 (432) (2016) re6–re6.

[20] B. J. Frey, D. Dueck, Clustering by passing messages between data points, science 315 (5814) (2007) 972–976.

[21] V. Y. Tan, C. Févotte, Automatic relevance determination in nonnegative matrix factorization with the/spl beta/-divergence, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (7) (2012) 1592–1605.

[22] J.-P. Brunet, P. Tamayo, T. R. Golub, J. P. Mesirov, Metagenes and molecular pattern discovery using matrix factorization, Proceedings of the national academy of sciences 101 (12) (2004) 4164–4169.

[23] C. Ding, X. He, H. D. Simon, On the equivalence of nonnegative matrix factorization and spectral clustering (2005) 606–610.

[24] X. He, D. Cai, P. Niyogi, Laplacian score for feature selection (2006) 507–514.

[25] T. Brouwer, J. Frellsen, P. Lió, Comparative study of inference methods for bayesian nonnegative matrix factorisation (2017) 513–529.

[26] M. N. Schmidt, O. Winther, L. K. Hansen, Bayesian non-negative matrix factorization (2009) 540–547.

[27] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. Jarrod Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, İ. Polat, Y. Feng, E. W. Moore, J. Vand erPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, S. . . Contributors, SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, Nature Methods 17 (2020) 261–272. doi:https://doi.org/10.1038/s41592-019-0686-2.

[28] T. pandas development team, pandas-dev/pandas: Pandas (Feb. 2020). doi:10.5281/zenodo.3509134.

[29] M. Zitnik, B. Zupan, Nimfa: A python library for nonnegative matrix factorization, Journal of Machine Learning Research 13 (2012) 849–853.

[30] S. Seabold, J. Perktold, statsmodels: Econometric and statistical modeling with python (2010).

[31] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, H. Liu, Feature selection: A data perspective, ACM Computing Surveys (CSUR) 50 (6) (2018) 94.

[32] H. Kim, H. Park, Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis, Bioinformatics 23 (12) (2007) 1495–1502.

[33] D. Verzella, A. Pescatore, D. Capece, D. Vecchiotti, M. V. Ursini, G. Franzoso, E. Alesse, F. Zazzeroni, Life, death, and autophagy in cancer: Nf-κb turns up everywhere, Cell Death & Disease 11 (3) (2020) 1–14.

[34] J. Guinney, R. Dienstmann, X. Wang, A. De Reyniès, A. Schlicker, C. Soneson, L. Marisa, P. Roepman, G. Nyamundanda, P. Angelino, et al., The consensus molecular subtypes of colorectal cancer, Nature medicine 21 (11) (2015) 1350–1356.

[35] M. Liang, Z. Li, T. Chen, J. Zeng, Integrative data analysis of multi-platform cancer data with a multimodal deep learning approach, IEEE/ACM transactions on computational biology and bioinformatics 12 (4) (2014) 928–937.

Appendix A

Biological Glossary

Since readers may be unfamiliar with biology, I've included this glossary to make the reading process a bit easier.

Apoptosis: the death of cells which occurs as a normal and controlled part of an organism's growth or development.

B-cell: AKA B-lymphocytes are a type of white blood cell that helps fight disease.

Biomarkers: any measurable biological property or set of properties which can indicate a disease.

Cell-of-origin (COO) classification: most studied subclassification of DLBCL. Splits into two classes: activated B-cell (ABC) and germinal center B-cell (GCB). Cases that do not fit into either class are called unclassified.

Co-regulation: the fact that two (or more) transcripts are regulated by the same mechanism e.g. the same transcription factor. Therefore, they show co-expression, their expression is correlated.

Cohort: a group of the same species - refers to a group of patients in our project.

Diagnosis: identification of a disease.

Diffuse large B-cell lymphoma: a type of cancer that develops from abnormal growth of B-cells which are a type of white blood cell.

Driver gene: a gene whose mutation eventually causes an important biological event to occur - in the context of this project, the event is cancerous cell growth.

GADD45B: a gene that encodes the growth arrest and DNA-damage-inducible, beta (GADD45β) protein. GADD45B has an important connection with the NF-κB pathway in terms of cancer.

Gene: a sequence of DNA or RNA that codes for a molecule that has a function.

Gene expression: Gene expression is the process by which genes produce useful molecules. Gene expression data corresponds to the levels of these molecules produced in a particular tumor sample. As such, gene expression levels measure a gene's activity in a sample. Through gene expression data, we can work out the important genes in a biological process since these genes are expressed at higher levels.

Gene signatures: combinations of genes (i.e. gene sets, or if they have some structure, gene networks) that produce some sort of molecular activity.

Genetic heterogeneity: common feature of tumor genomes. Describes the observation that different tumor cells can show distinct genotypic and phenotypic profiles, causing differences in cancers from person to person.

Gene regulation: a pathway that turns genes on or off.

Genomics: the study of the genome (i.e. all of the genes in a person's body).

Growth arrest: refers to the phenomenon where a cell does not proceed through the normal cell cycle.

Inflammation: the process by which white blood cells enter your blood and tissues to help fight a foreign body.

Lymphoma: a cancer that develops when white blood cells grow at an abnormal rate.

Messenger RNA (mRNA): large family of RNA molecules that convey genetic information from DNA to the ribosome, where they specify the amino acid sequence of the protein products of gene expression.

Microarray: A DNA microarray is a collection of small DNA spots attached to a slide. A signal is collected from each spot which is then used to estimate a gene's expression level. There are thousands of spots, covering most genes in a genome.

Multiple myeloma: a type of bone marrow cancer.

Mutation: an alteration in the DNA sequence. A somatic mutation is a mutation that can occur in all cells except sperm and egg cells.

NF-κB: Short for "nuclear factor kappa-light-chain-enhancer of activated B cells". It is a transcription factor used in lots of important biological processes in the body which makes it a prime target for cancer.

Omics: study of different areas in biology whose names end in -omics. Examples include genomics (used in this project) and metabolomics.

Oncology: the study of cancer. Oncogenesis refers to cancer.

Pathway: a series of interactions among molecules in a cell that leads to a certain product or a change in a cell. A pathway can trigger the assembly of new molecules, such as a fat or protein.

Patient stratification: Stratification is the division of your potential patient group into subgroups, also referred to as 'strata' or 'blocks'. Each strata represents a particular section of your patient population.

Prognosis: prediction of how a disease or medical condition will develop.

Proliferate: the verb used to describe cells multiplying, often quickly or out-of-control.

Subtype: a subclass of a cancer defined by its biological characteristics.

Transcription: copying DNA sequence from a gene into RNA. This is the first step of gene expression. A transcription factor is a protein that controls the transcription rate.

Tumor: a swelling of a part of the body, generally without inflammation, caused by an abnormal growth of tissue, whether benign or malignant.

Tumor micro-environment (TME): the environment surrounding a tumour. The tumor and TME can interact with each other, affecting the growth of cancerous cells.

Sample purity: AKA tumor purity in cancer. The proportion of diseased (cancerous) cells in a sample.

Appendix B

Data & Preprocessing: Other Figures

B.1 Gene Expression Data

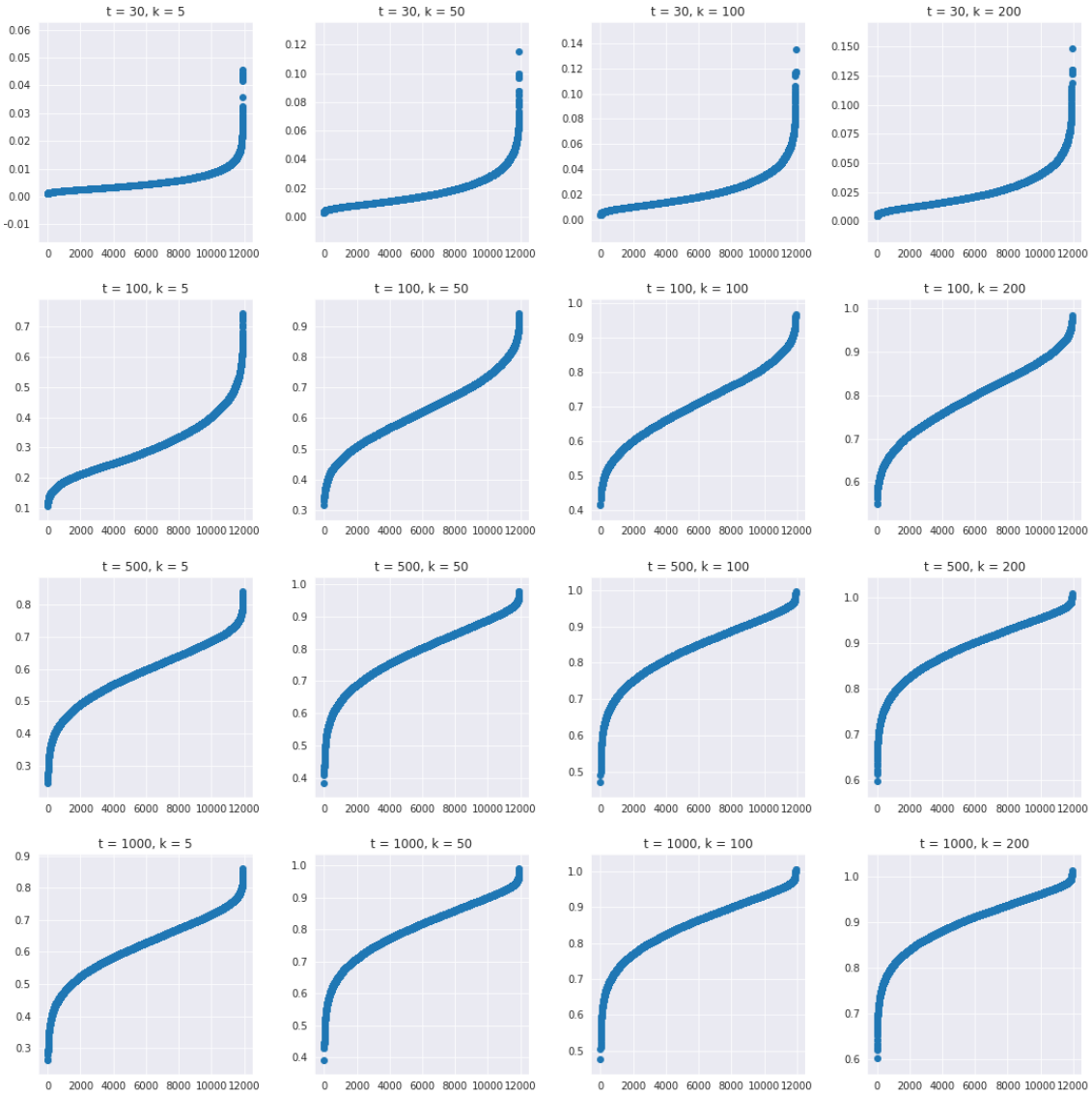


Figure B.1: Laplacian score plots for features after variance thresholding varying t and k

B.2 Initial Data Analysis

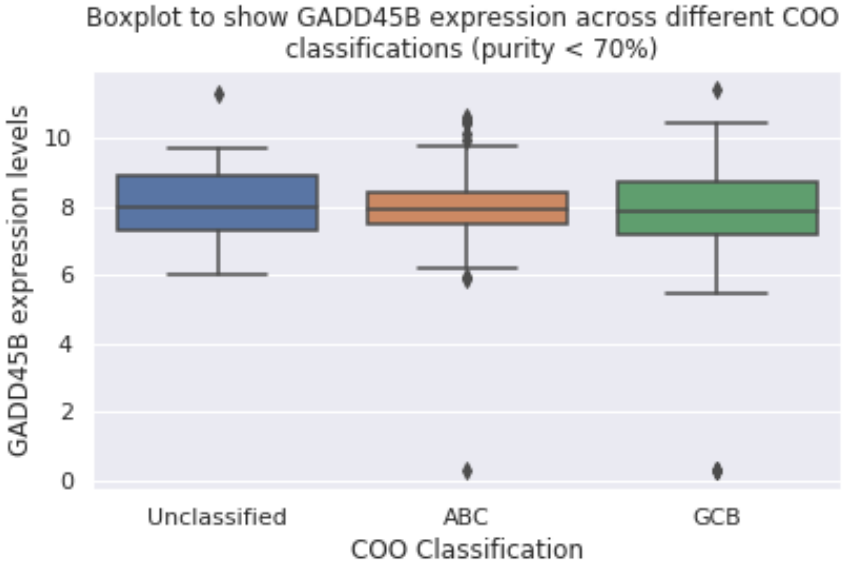


Figure B.2: Boxplot of GADD45B levels for different COO classifications at low purities

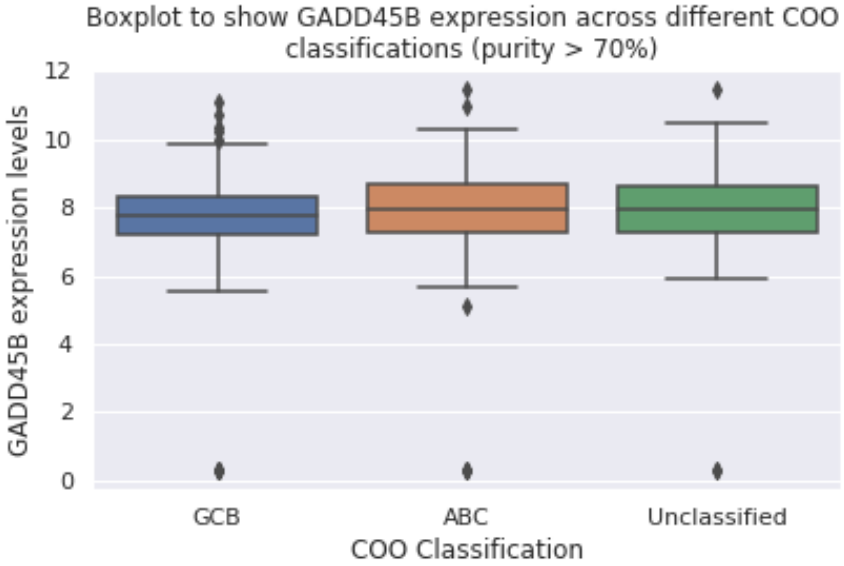


Figure B.3: Boxplot of GADD45B levels for different COO classifications at high purities

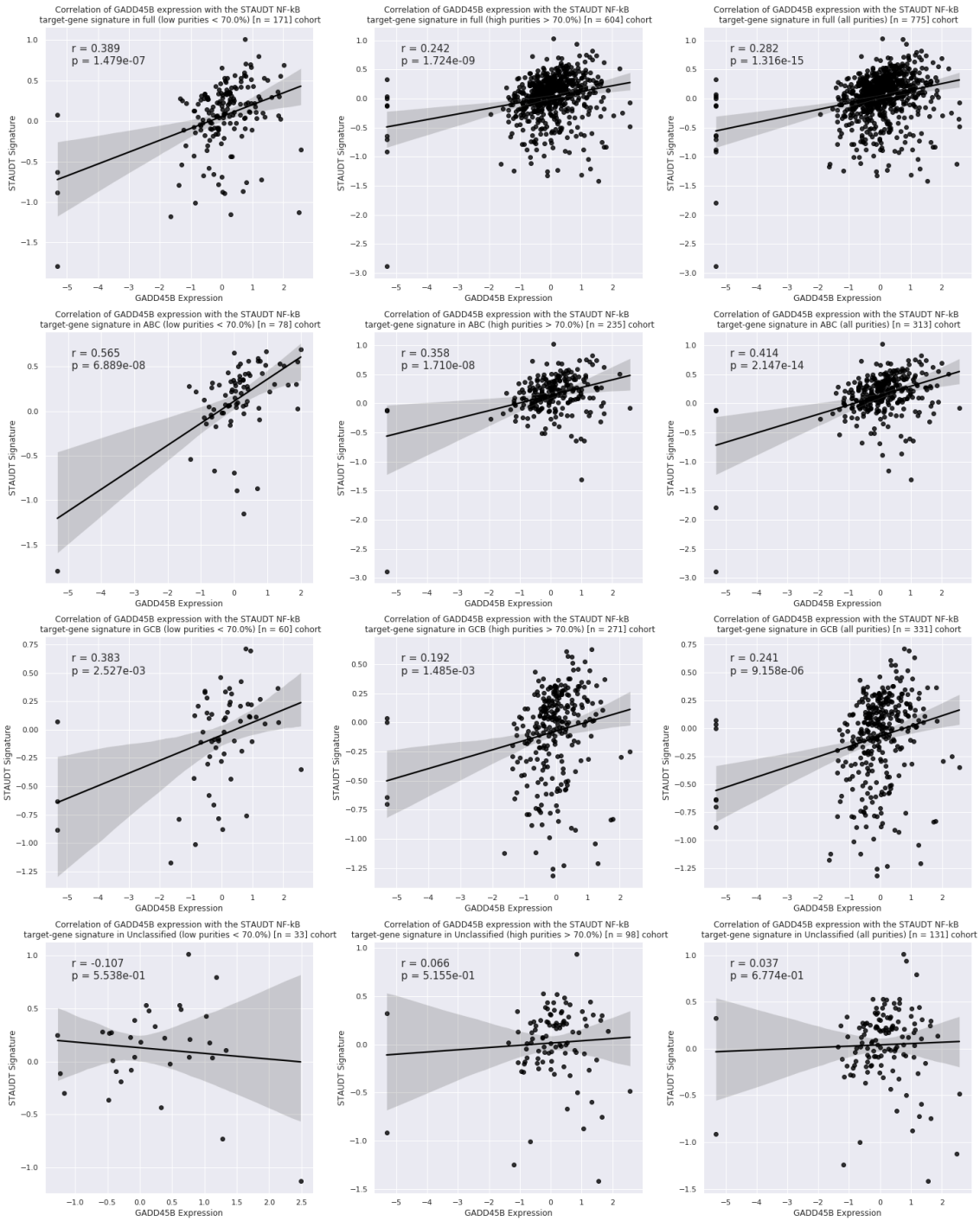


Figure B.4: Correlations between *Staudt* signature and GADD45B expression at different purities for different COO cohorts.

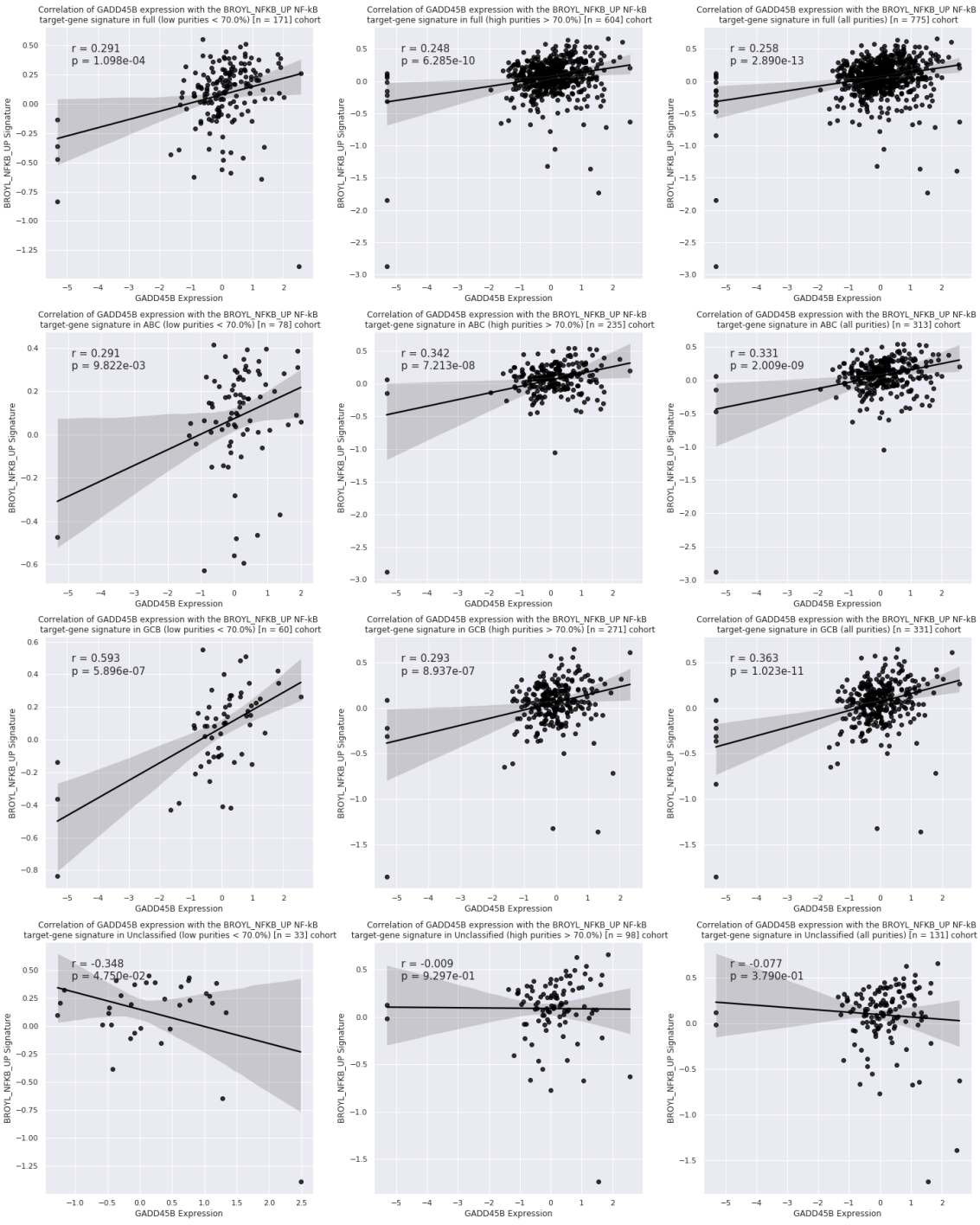


Figure B.5: Correlations between *Broyl* signature and GADD45B expression at different purities for different COO cohorts.

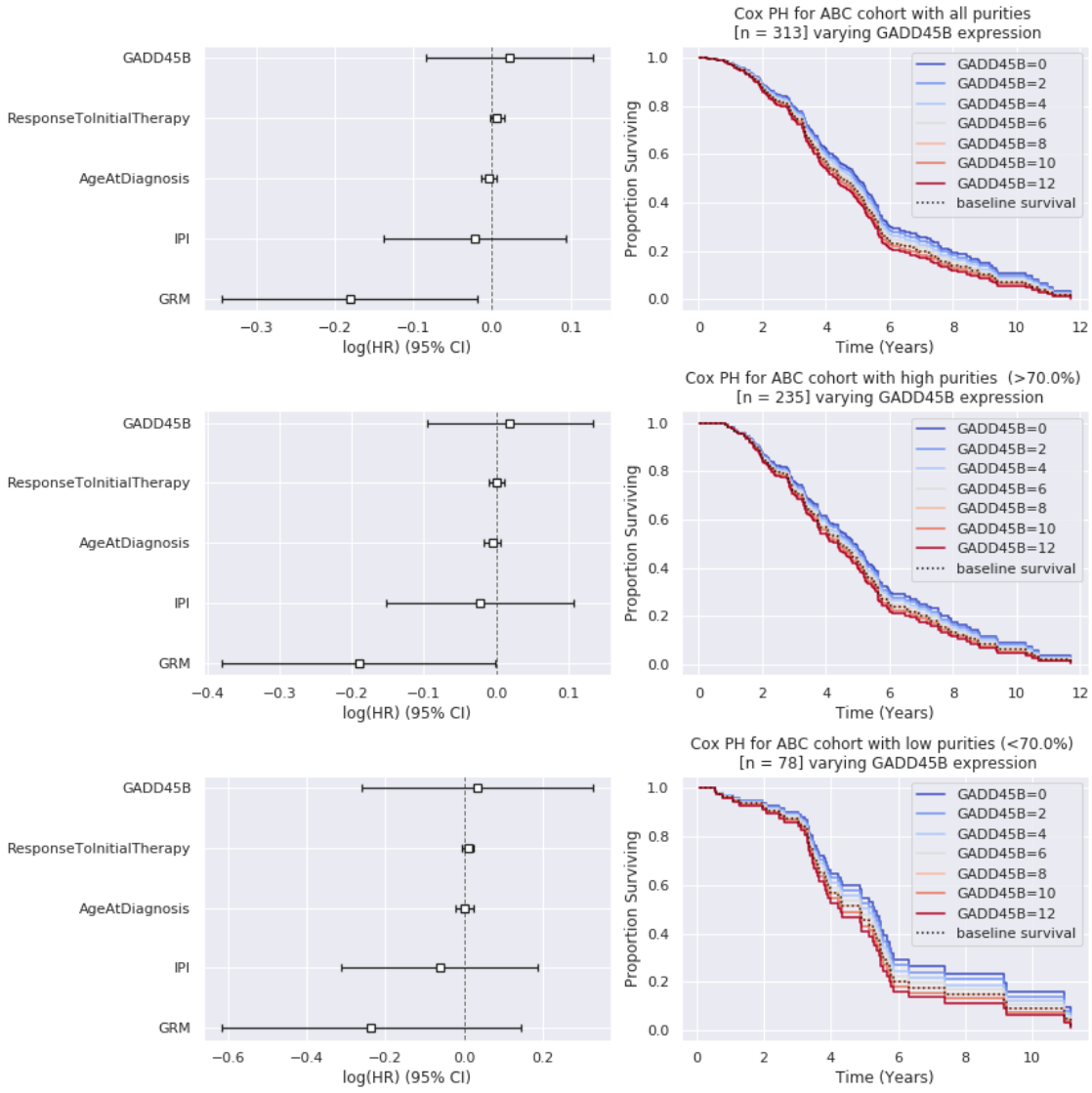


Figure B.6: Hazard ratio plot and Cox PH plot with varying GADD45B on ABC cohort at different purities.

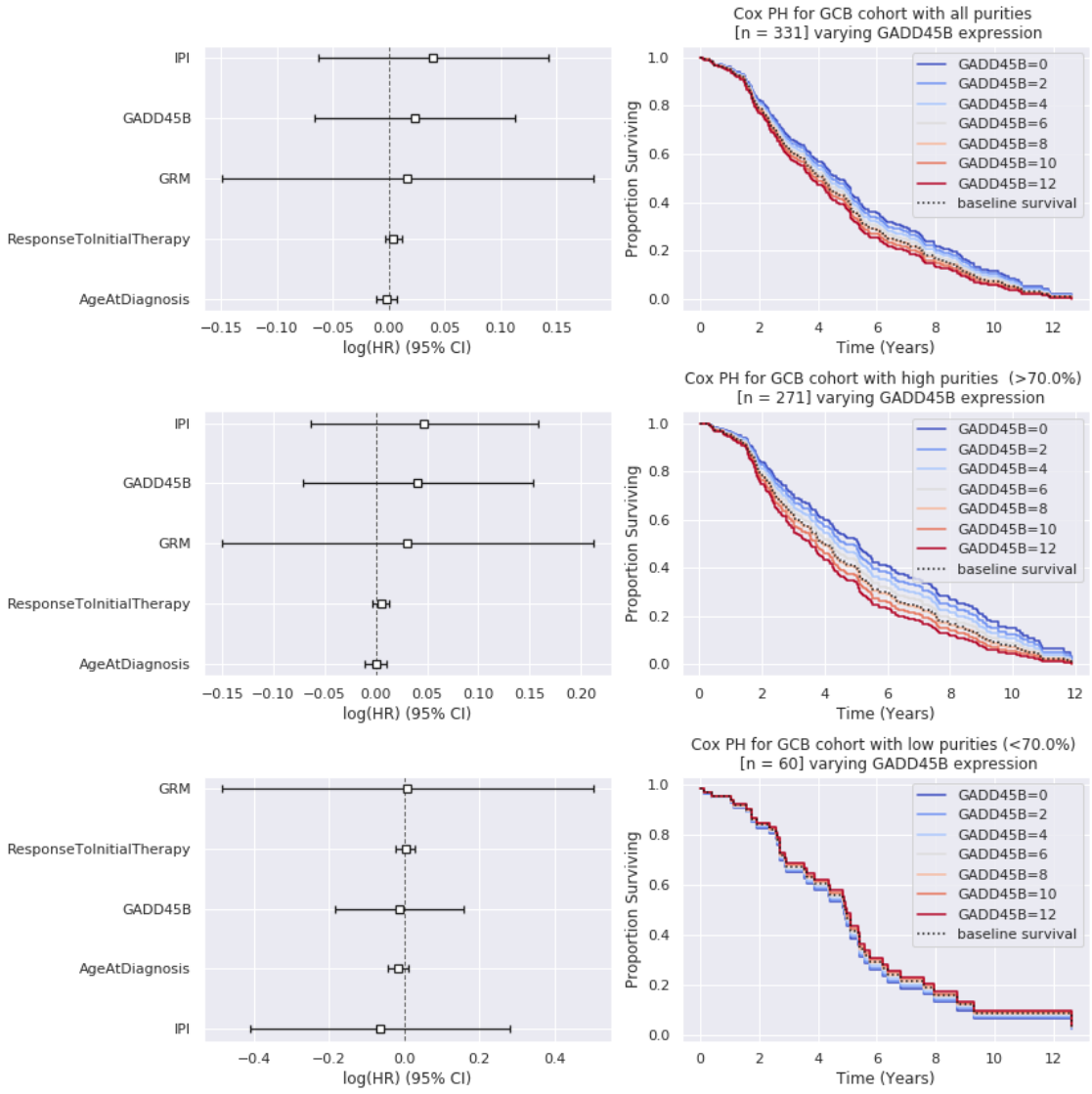


Figure B.7: Hazard ratio plot and Cox PH plot with varying GADD45B on GCB cohort at different purities.

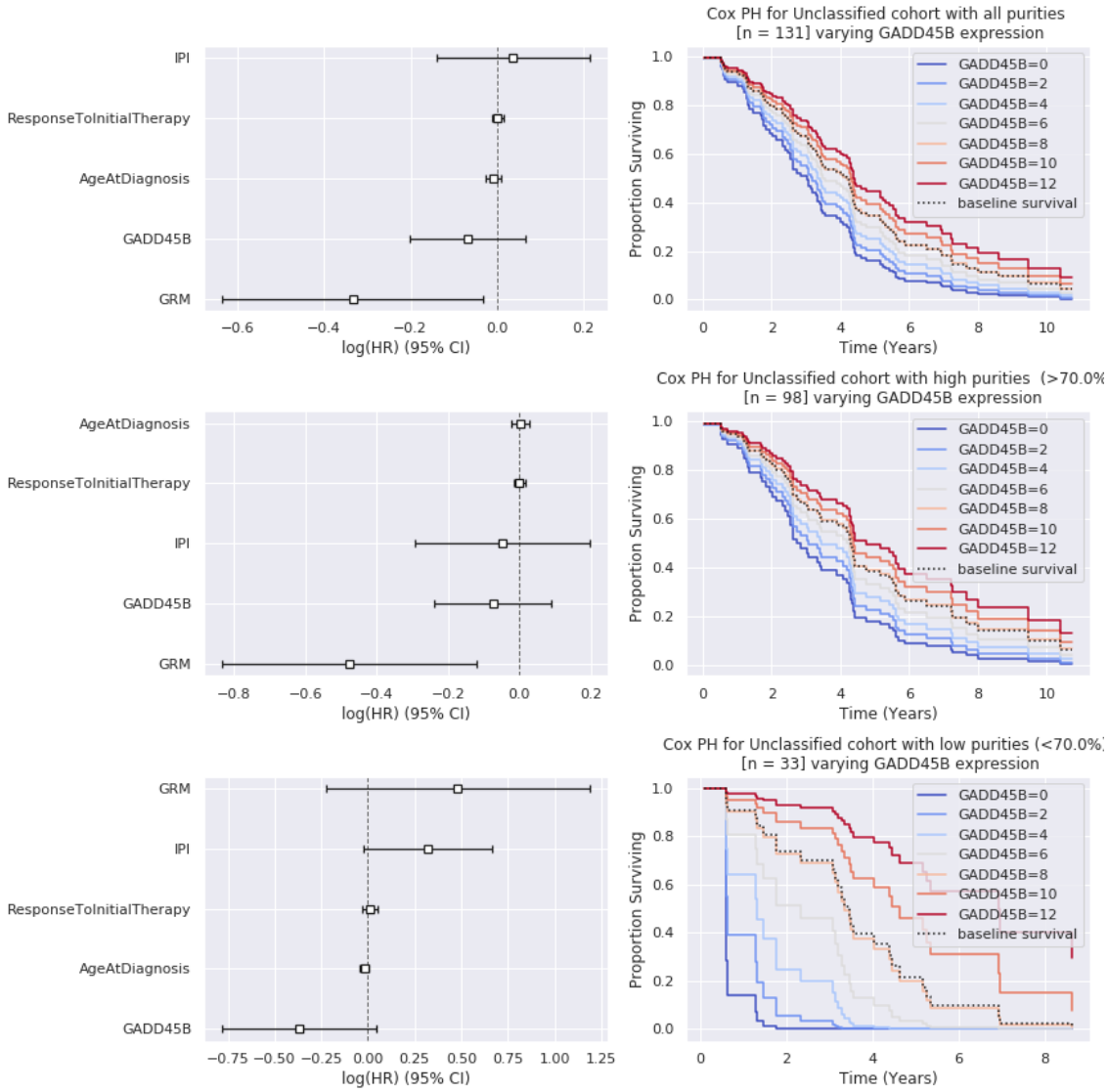


Figure B.8: Hazard ratio plot and Cox PH plot with varying GADD45B on Unclassified cohort at different purities.

Appendix C

Evaluation: Other Figures

C.1 Clinical Analysis

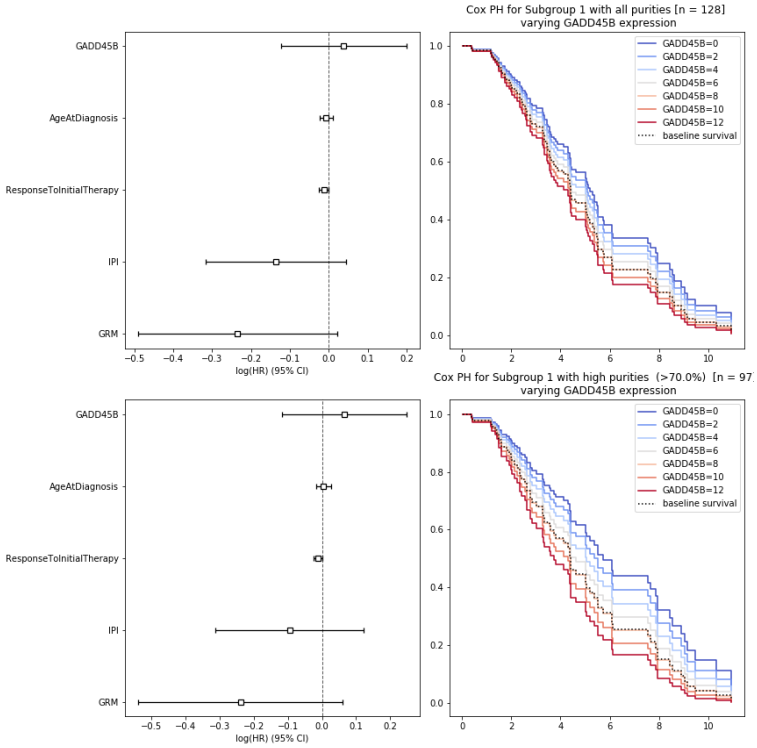


Figure C.1: Cox PH plots for high and all purities for subgroup 1 derived from the GS algorithm at rank 5.

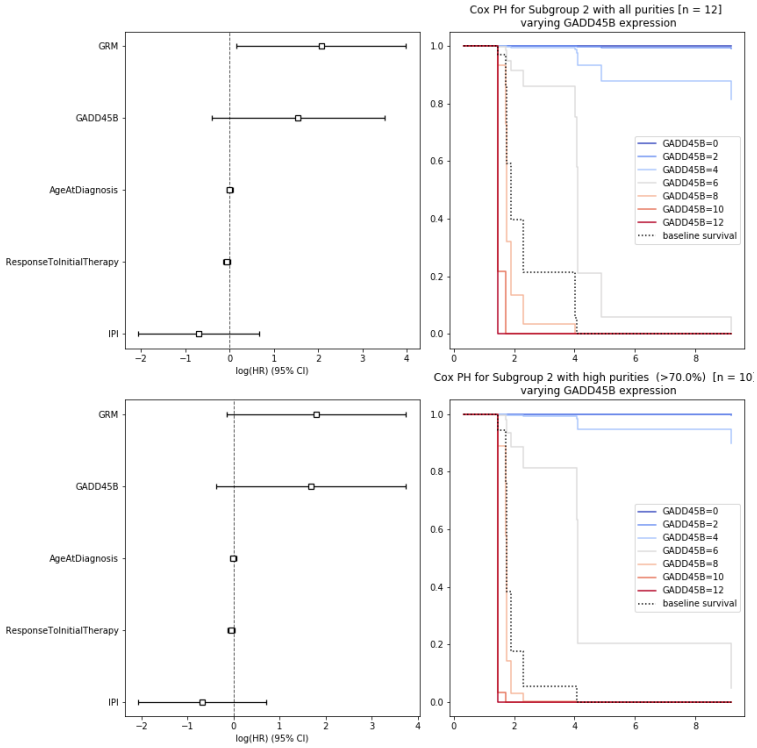


Figure C.2: Cox PH plots for high and all purities for subgroup 2 derived from the GS algorithm at rank 5.

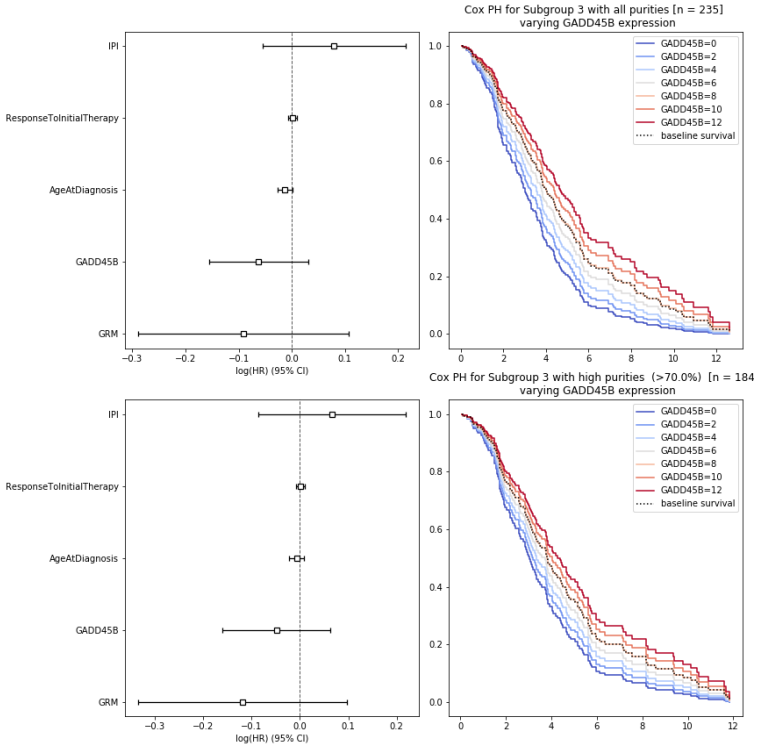


Figure C.3: Cox PH plots for high and all purities for subgroup 3 derived from the GS algorithm at rank 5.

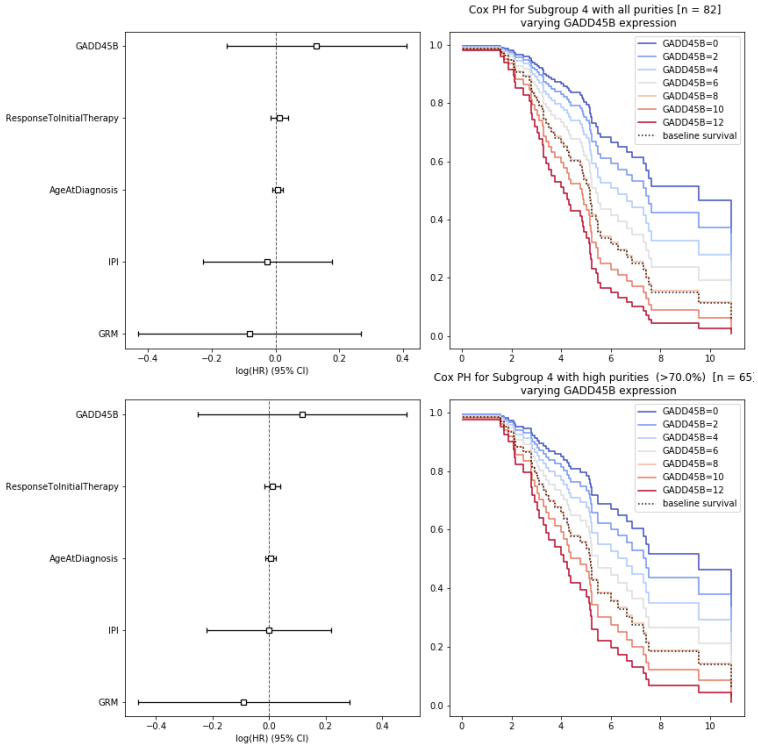


Figure C.4: Cox PH plots for high and all purities for subgroup 4 derived from the GS algorithm at rank 5.

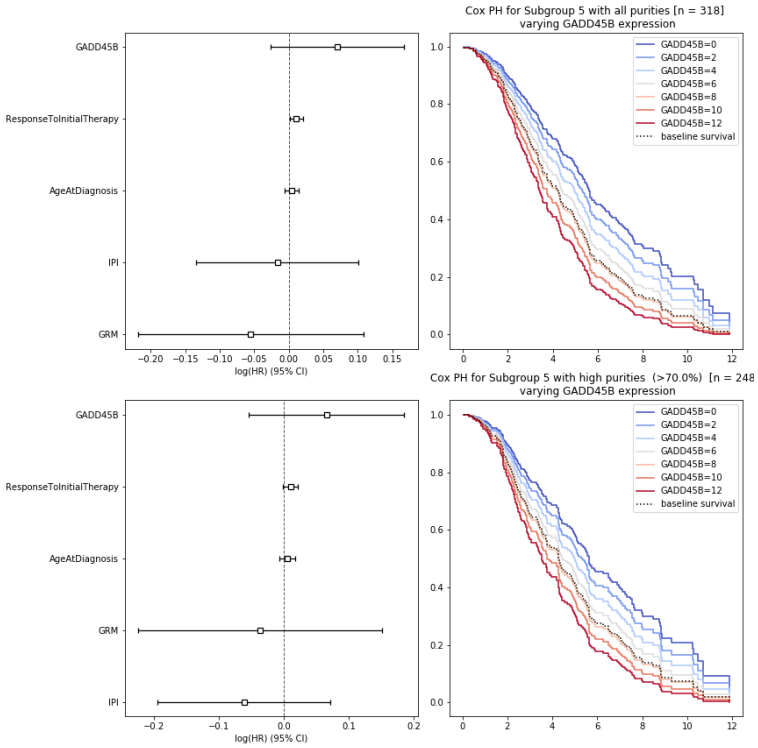


Figure C.5: Cox PH plots for high and all purities for subgroup 5 derived from the GS algorithm at rank 5.

C.2 Biological Analysis

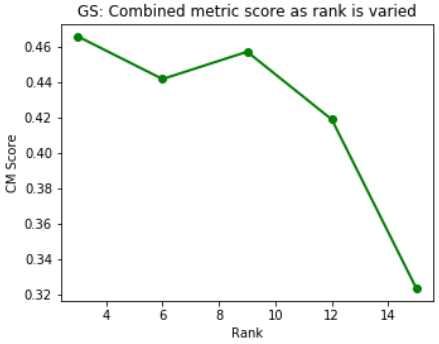


Figure C.6: Subgroup 1 CM score across different ranks using 1,500 genes.

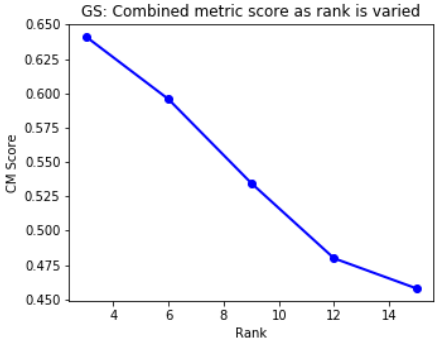


Figure C.7: Subgroup 3 CM score across different ranks using 1,500 genes.

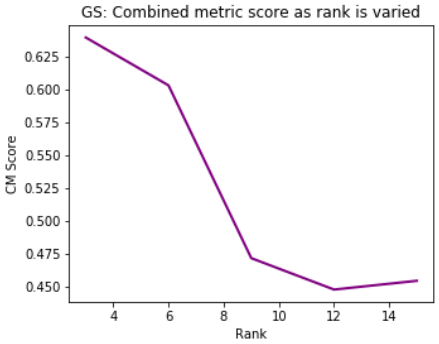


Figure C.8: Subgroup 4 CM score across different ranks using 1,500 genes.

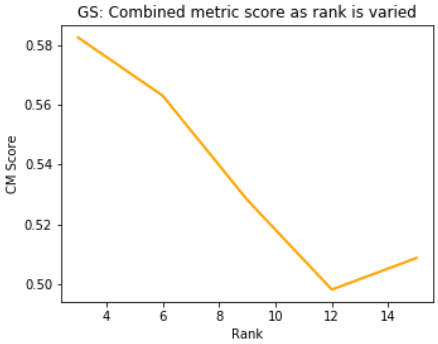


Figure C.9: Subgroup 5 CM score across different ranks using 1,500 genes.