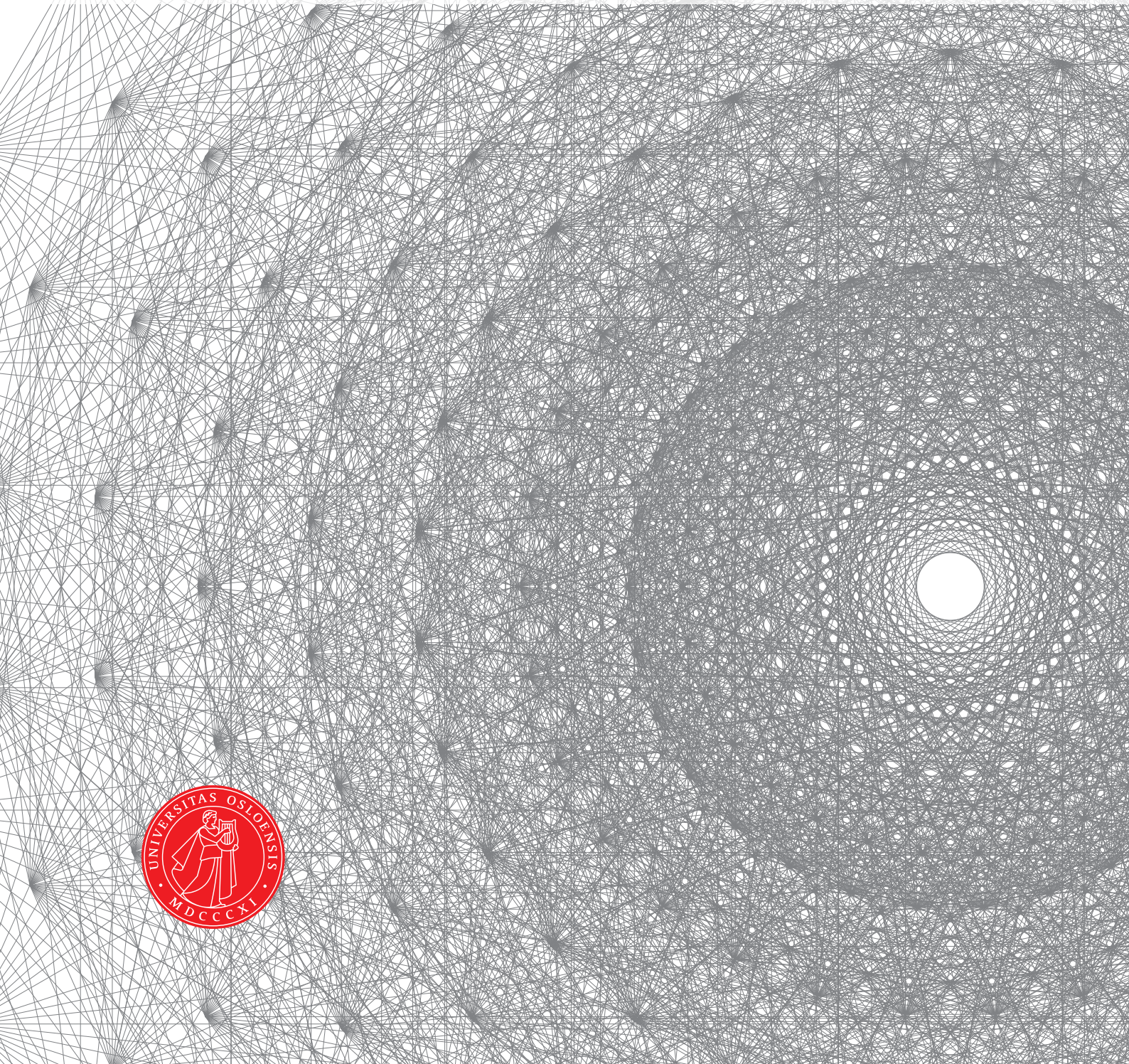
UiO : Department of Mathematics
University of Oslo

Integrating Biological Domain Knowledge in Machine Learning Models for Cancer Precision Medicine

Anders Kielland

Master's Thesis, Spring 2023



This master's thesis is submitted under the master's programme *Data Science*, with programme option *Data Science*, at the Department of Mathematics, University of Oslo. The scope of the thesis is 60 credits.

The front page depicts a section of the root system of the exceptional Lie group E_8, projected into the plane. Lie groups were invented by the Norwegian mathematician Sophus Lie (1842–1899) to express symmetries in differential equations and today they play a central role in various parts of mathematics.

Acknowledgements

I would like to express my profound gratitude to my supervisors Alvaro Kohn Luque, Leonard Schmiester and Ingrid Kristine Glad. Alvaro, my primary supervisor, deserves special thanks for allowing me to pursue my thesis in his group and for creating an inclusive and stimulating working environment. Leonard has been instrumental in assisting me with everything from minor practical issues to complex scientific problems, especially in relation to the mechanistic model. Ingrid, I am grateful for her close guidance throughout the entire process, as well as for her thorough discussions and valuable suggestions on the scientific aspects. Her positive and warm demeanor has made the thesis work a more enjoyable experience.

I would also like to extend special thanks to George Zhi Zhao, who, although not my official supervisor, generously took the time to help me with writing code and understanding statistical problems. I particularly appreciate his contributions to the development of a new approach for group interaction. This thesis would have been significantly different without George's involvement.

Finally, I would like to offer my heartfelt thanks to Chrysolula Kielland for her unwavering support throughout my entire study period and for taking exceptional care of our children. Her love and dedication have always meant, and will continue to mean, a great deal to me.

Contents

Abstract

Cancer is an incredibly complex and diverse disease. Therefore, medical treatment preferentially should be tailored at the level of individual patients. There exists a vast amount of knowledge related to cancer biology, diagnosis, and treatment, and an extensive amount of measurements can easily be performed on each patient. A key challenge is to utilize such large amounts of information to design the most precise treatments.

This thesis addresses this problem by analyzing data from a clinical trial on breast cancer treatment. The trial investigated a combination of hormone therapy with a targeted drug that specifically inhibits CDK4/6, a protein involved in estrogen-stimulated cell proliferation. The trial included 49 patients, with measurements of 771 gene expression levels. The outcomes were two continuous scores which aimed to quantify cancer cell proliferation and long-term prognosis.

We have compared various machine learning models, both alone and in combination with domain biological knowledge, to assess their predictive power for cancer treatment outcomes. Furthermore, we evaluated the integration of machine learning models with a mechanistic mathematical model characterizing the mechanisms of action of the targeted drug. Finally, we explored the use of domain knowledge in a novel model approach.

Among the standard model classes - ridge regression, lasso, elastic net, and boosting with stumps as base learners - ridge demonstrated the best predictive performance. Feature selection revealed high overlap between lasso and elastic net, while boosting showed an overlap of approximately half with the two linear models. The integration of mechanistic and machine learning models did not improve upon the standard models.

To leverage biological knowledge, the gene set was divided into smaller subsets based on each gene's involvement in different aspects of breast cancer biology, such as regulation of cell proliferation, estrogen signaling, immune system activity, and DNA repair mechanisms. The smaller gene subsets underwent feature engineering through principal component analysis, and the resulting components were used as covariates in the standard machine learning models. This led to a slight improvement in predictive power and offered some insights into the importance of different aspects of breast cancer biology. We also included interaction terms between principal components from different gene sets, which further improved predictive performance.

In a second attempt to utilize biological knowledge, we employed a stacking-like approach by first training models on the gene sets individually, and then

using the predictions of these models, each representing a gene set, as input features for a new machine learning model. This method did not outperform the best standard model.

Lastly, inspired by the potential of modeling interactions between functional units of cancer biology, we attempted a novel iterative approach focusing on these interactions. This method showed promising results on simulated data with more observations than features but faced challenges when the number of observation became too small.

CHAPTER 1

Introduction

The primary aim of this thesis was to explore the potential of combining machine learning and domain knowledge to develop predictive tools for cancer treatment, using data from a specific clinical trial. A secondary goal was to evaluate the predictive performance achieved by integrating an established mechanistic model with machine learning models.

1.1 Cancer

Cancer is a severe global health problem. Estimates suggest that 19.3 million new cancer cases and almost 10 million cancer-related deaths occurred worldwide in 2020 (Bray et al. 2018). Female breast cancer was the most commonly diagnosed cancer, with an estimated 2.3 million new cases. This makes cancer the second leading cause of death globally, after cardiovascular disease. In some developed countries, cancer has become the leading cause of death in recent years. Prevention and treatment of cancer is in general more challenging than for cardiovascular diseases as cancer is more complex, less dependent on lifestyle dependent risk factors, and cancer is most often diagnosed later in disease progression (Sung et al. 2021).

Although cancer is primarily considered to be a genetic disease it is a highly diverse and heterogeneous condition, encompassing numerous types, subtypes, and developmental stages. As a result, a variety of different treatment approaches is required. For instance, breast cancer cells are sometimes sensitive to estrogen, which determines whether hormone therapy should be included as part of the treatment plan (Waks and Winer 2019).

In addition to the effect of treatments on the tumor, side-effects must be considered. Side-effects of cancer treatment can be severe and persist, affecting patients for the rest of their lives. Moreover, there are multiple patient specific factors that also should influence the choice of treatment, including cancer stage, overall health, patient demographics and genetics. Another challenge is that cancer cells typically change during the progression of the diseases. Both mutations of their genome and changes in their metabolism lead to transformation of the cells. This causes diversification of the cancer cells and may lead to development of resistance to ongoing treatment. Consequently, it becomes difficult to effectively eliminate all cancer cells and prevent the disease from recurring. Economic factors can have a significant impact on availability of treatment. Cancer treatment can be very expensive and in national health systems the cost of a treatment is weighted against the clinical

benefit. Considering all these factors makes determining the ideal cancer treatment for an individual patient challenging. Therefore, tailoring cancer treatment to the individual patients remains one of the most pressing global health issues today and in the foreseeable future.

1.2 Precision medicine - a high dimensional problem

An approach addressing the tailoring of disease treatment and a goal in modern healthcare systems is the development of precision medicine. Precision medicine represents a shift in how medical care is provided where the aim is to move away from a one-size-fits-all approach towards a more tailored and personalized approach that takes into account the unique needs and characteristics of each patient. In cancer treatment, precision medicine typically involves analyzing the genetic compositions of a patient's cancer cells and in combination with demographic factors tailor treatment to specifically target the characteristics of the present cancer cells.

A major challenge in developing the precision approaches is the high dimensionality of the available research data. High-dimensional data refers to situations where the number of features, or dimensions, is close to or larger than the number of observations (Hastie, Tibshirani and J. Friedman 2009). A typical human cell expresses around 5000 genes and although not all of these genes are necessarily relevant to cancer biology a substantial number of them have the potential to be of importance as a diagnostic marker. In addition, other diagnostic and demographic variables will be part of the feature space. Since clinical studies often are limited to a couple of hundred patients, or less, and often distributed across various treatments strategies, the development of precision based cancer treatment will typically give a high dimensional problem.

1.3 Importance of machine learning in cancer medicine

Artificial intelligence refers to the idea of using computer-based systems to carry out highly complex and advanced analytical and decision-making processes. Machine learning is among the most successful part of artificial intelligence. Machine learning can be dived in two categories: traditional machine learning and deep learning, the latter involves the use of artificial neural networks.

In recent years, deep learning has proven to be highly successful in various fields and in many problems surpassing traditional machine learning, particularly in prediction tasks. However, deep learning requires a large amount of training data, thus traditional machine learning models are still useful, particularly for data-limited or tabular data problems. In this thesis, the available data is insufficient for deep learning, making machine learning the method of choice. Traditional machine learning includes a variety of algorithms which primarily are designed to analyse tabular data. These algorithms are construct to solve unsupervised and supervised learning with continuous and classification outcome, often involving numerous features. Prediction and feature selection are common objectives in traditional machine learning applications.

In recent years, advancements in processing the biomolecular composition of tumor samples have generated large datasets, which give opportunities for improved molecular cancer diagnosis, prognosis, and treatment (Hanahan 2022).

However, the often small sample size is then becoming more challenging by the high-dimensional data structure. Machine learning models have played an important role in utilizing these datasets for precision medicine, uncovering patterns, predicting outcomes, and identifying important features, ultimately enabling the development of more personalized and effective treatment strategies (Azuaje 2019; Swanson et al. 2023). While blood sample-based methods are being developed for diagnostics, molecular analysis of the tumor samples provides the ultimate information to characterize cancer prognosis. Through tumor monitoring, machine learning have shown great promise for selecting cancer treatments and predicting responses. The standard in current treatment selection is determined by clinical guidelines and trials that typically use a few clinical features. In contrast, molecular profiles of cancers generate a much larger number of features that can inform cancer treatments. For instance, Sammut et al. (2022) predict chemotherapy response by incorporating clinical, genomic, transcriptomic, pathology and treatment information into an ensemble model that averages the predictions of logistic regression, support vector machine and random forest models.

These machine learning algorithms reflect significant advances in the research landscape. However, before the algorithms can be used to treat patients, they generally require regulatory approval, which involves more rigorous clinical trials and validation than what is usually presented in academic work. Consequently, only a small proportion of the algorithms end up being used in the clinic (Wu et al. 2021). In conclusion, although machine learning is increasingly important in cancer detection, prognosis, and treatment planning, it is likely that machine learning algorithms have far from reached their full potential.

1.4 Tumor ecosystems - impact on cancer treatment

A tumor is not just a mass of cancer cells. It is regarded as a complex ecosystem that consists of both cancerous and non-cancerous cells, as well as a network of blood vessels, immune cells, and many other components (Marusyk, Janiszewska and Polyak 2020). The development and growth of a cancer is a complex process that involves not only the cancer cells but also the surrounding environment. The interactions between the cancer cells and the environment, and furthermore, the interactions between different parts of the environment can play critical roles in the responses to different types of treatments. Despite the importance of the tumor ecosystems, few efforts to predict treatment response have taken these factors into account (Sammut et al. 2022). Therefore, there is likely a large potential for developing precision medical approaches by integrating data that represents different parts of the tumor ecosystem in order to accomplish more accurate predictions and optimal treatment decisions. In this thesis, we have incorporated domain expertise of the tumor ecosystem into machine learning models.

1.5 Main findings

In this thesis, data from a clinical trial study using targeted drug therapy against breast cancer was analyzed. The primary aim was to evaluate the potential of predictive models as tools for selecting patients who would benefit

from the tested drug combination. The predictors included 771 gene expression measurements, and 49 patients received the particular drugs.

Among the standard machine learning model classes, ridge regression demonstrated the best predictive performance. We managed to enhance the performance by introducing domain knowledge, particularly when incorporating interactions between smaller groups of genes based on each gene's involvement in different parts of the cancer biological ecosystem. In a novel model approach proposed in this thesis to leverage domain knowledge, we achieved success with high-sample-size simulated data but faced challenges when the number of observation became too small. As a result, the model proved to be ineffective when applied to the dataset from the clinical trial.

The three genes LEFTY2, GATA3, and HDAC2 were consistently selected in various model scenarios. LEFTY2 is known to be involved in the regulation of cell growth, and its dysregulation has been implicated in tumor development and progression (Saito et al. 2013). GATA3 is a transcription factor that plays a role in cell differentiation. Abnormal expression of GATA3 has been associated with tumor progression and poor prognosis in breast cancer patients (Yoon et al. 2010). HDAC2, a histone deacetylase enzyme, has been implicated in the regulation of gene expression, cell cycle progression, and cellular differentiation, with its altered expression linked to various cancers (Li, Tian and Zhu 2020). To the best of our knowledge, no association of these genes with response to the drugs used in the clinical trial has been previously described.

All code scripts used throughout this thesis can be downloaded from a GitHub repository (https://github.com/akielland/Cancer).

1.6 Outline of the thesis

The remainder of this thesis is organized as follows. Chapter 2 offers an overview of the biological and clinical background of the treatment regime used in the clinical trial and the study design of the trial. In this chapter, we also review relevant details of the trial's dataset, including the division of genes into smaller gene sets based on cancer biology domain knowledge.

Chapter 3 introduces standard statistical terminology and covers the standard machine learning models ridge regression, lasso, elastic net and boosting. In Chapter 4, we conduct an analysis comparing the performance of these standard models on the trial data.

In Chapter 5, we examine the integration of a mechanistic model with machine learning models to enhance their predictive capabilities.

Chapter 6 investigates the benefits of incorporating cancer biology domain knowledge into statistical methods with the goal of guiding the application of principal component regression, stacking, and interactions.

Finally, chapter 7 presents a novel modeling approach that incorporates group interactions, discussing its characteristics, algorithm, and performance on simulated data and the clinical trial dataset. The thesis concludes with a summary of the findings, implications, and suggestions for future research in the final chapter 8.

CHAPTER 2

The patient data of the CORALLEEN trial

2.1 Breast cancer

Breast cancer is the most common cancer type in women and the second cause of cancer-related mortality (Sung 2020). The breast cancer subtype addressed in the thesis is luminal B by the so-called PAM50 classification (Perou et al. 2000; Wallden et al. 2015). The other four subtypes are luminal A, HER2-enriched, basal-like and normal-like. Luminal B tumor cells are characterized by the expression of the estrogen receptors, being HER2-negative and showing high levels of proliferation markers. The latter refers to genes that are stimulating cell division. Increased expression of these proliferation genes in tumor cells is associated with poor prognosis in cancer patients. The prevalence of luminal B is approximately 15% of all breast cancer cases and while prognosis varies substantially between individual patients luminal B is generally accepted to have a middle prognostic outcome.

2.2 The targeted drug

The clinical trial utilized a combination of the drugs letrozole and ribociclib for treatment. Letrozole is a well-established hormone therapy for breast cancer that has been in use for a long time. It functions by blocking estrogen production in the body, which in turn inhibits the growth of hormone-sensitive breast cancer cells. Ribociclib, the primary drug of interest in the trial, is a targeted drug. A targeted drug is generally defined as a pharmacological treatment designed to specifically interfere with a distinct molecule to regulate its role in cellular functions. In the last few years, ribociclib, along with two other similarly acting drugs (palbociclib and abemaciclib), has been approved for concurrent use with hormone therapy in hormone-receptor-positive/HER2-negative breast cancer, which account for 65-70% of the breast cancer cases. This development have considerable changed the clinical practise for this type of cancer (Burstein et al. 2021).

Here, we give a simplified overview of the biological mechanisms underlying the therapeutic effect of ribociclib (for thorough description see Goel, Bergholz and Zhao 2022 and Fassl, Geng and Sicinski 2022). The target molecule of ribociclib is cyclin-dependent kinase 4 and 6 (CDK4/6), which play crucial

roles in regulating the cell cycle and, consequently, cell proliferation. The cell cycle progresses through different phases, and the transition between these phases is strictly regulated. A hallmark of cancer is the loss of control over this regulation. CDK4/6 regulates the transition from the growth 1 phase to the synthesis phase, where DNA replication occurs. Cyclin D, a protein, binds to and activates CDK4/6. This complex then modifies (i.e., phosphorylates) another protein called retinoblastoma protein 1 (RB1). In its unmodified state, RB1 blocks a transcription-regulating factor (E2F) that, when active, stimulates the expression of the genes necessary for DNA replication. When RB1 is modified by the CDK4/6-cyclin D complexes, it releases the transcription factor, which subsequently promotes the expression of the genes required for transitioning the cell into the synthesis phase of the cell (a schematic representation is presented in Figure 2.1).

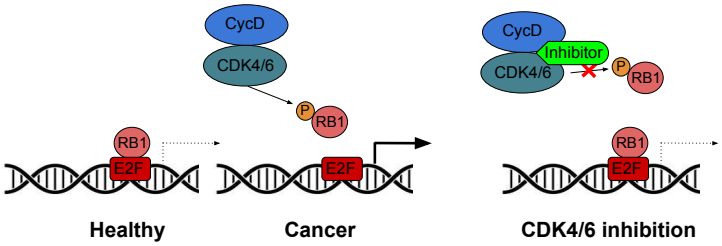


Figure 2.1: Schematic representation of the molecular mechanisms of the targeted drug CDK4/6 inhibitor.

In cancer cells, this signaling pathway can be overactive, leading to uncontrolled cell proliferation. The mechanism of ribociclib is not fully understood, but there are indications that it blocks the activity of CDK4/6 either directly by binding to CDK4/6 or through other more indirect ways (Goel, Bergholz and Zhao 2022; Fassl, Geng and Sicinski 2022). However, the downstream effect appears to be the prevention of CDK4/6 from activating RB1, thereby down-regulating signaling that promotes the transition to the synthesis phase.

A significant challenge in the therapeutic use of CDK4/6 inhibitors is the considerable variability in patient responses to the treatment. This can be observed by examining the markedly divergent response frequency in the clinical trial analysis in this thesis (Figure 2.2). It is known that patients who initially respond to treatment may develop drug resistance. Furthermore, tumors may display cancer cells with preexisting, intrinsic resistance to CDK4/6 inhibitors. One of the primary objectives moving forward is to prescreen patients to identify those who are likely to respond well to these inhibitors. Investigating whether machine learning can be a useful tool for this task is a goal of this thesis.

2.3 Measuring gene expression

The most common and simplest approach to assess gene activity in the cells of an organism is to quantify the levels of messenger ribonucleic acid (mRNA) molecules. mRNA is synthesized in the cell nucleus through a process called transcription, where the genetic information encoded in the DNA is copied into

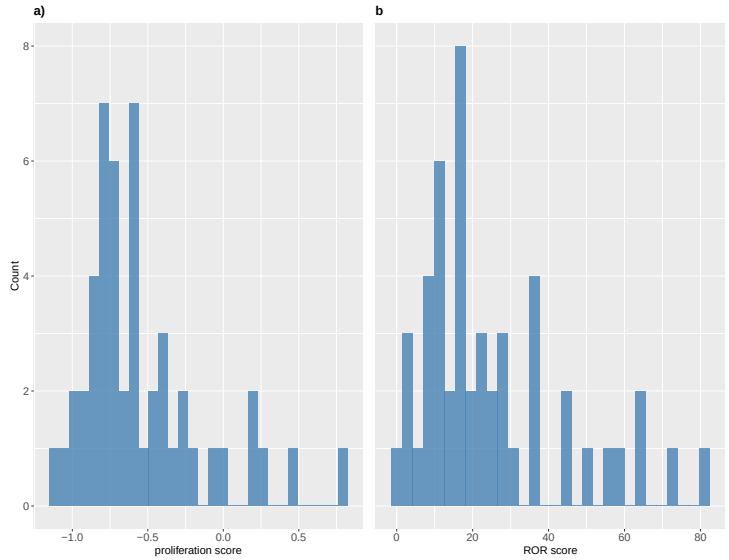


Figure 2.2: Response variables in the clinical trial with 49 patients. a) shows the proliferation score, while b) shows the ROR score.

the mRNA molecules. Following transcription, mRNA molecules migrate out of the nucleus into the cytoplasmic area (i.e. the material within a cell besides the nucleus), where they provide the genetic information to the protein building machinery of the cells. Proteins determine the morphology and functionality of a cell. Consequently, the cellular identity and functionality are dependent on the expression of a particular set of proteins. Therefore, measuring protein levels provide more accurate information about a cell's identity then measuring mRNA. However, it is considerably more difficult to measure protein than mRNA. A challenge in using mRNA levels as a measure of cellular identity is the transient nature of the mRNA molecules. While proteins directly represent the identity and functionality of the cells, the role of mRNA is to transfer genetic information from DNA. Once their role is accomplished, mRNA molecules are degraded. Therefore, mRNA levels fluctuate over time and will not accurately capture the snapshot of the actual cellular identity. Nonetheless, assessing mRNA levels remains a widely used and valuable approach for studying gene expression.

The study of the complete set of mRNA transcripts in a cell population (or single cell) is called transcriptomics, and there are numerous techniques available for carrying out such analyses. In this thesis, a variant of the microarray technology called Nanostring is employed to analyze the dataset. Briefly, Nanostring utilizes uniquely designed molecular probes to specifically bind to each target mRNA of interest. These molecular probes consist of a pair for each mRNA: a capture probe and a reporter probe. The capture probe immobilizes the mRNA onto a surface, while the reporter probe carries a colour-coded barcode that allows for the identification of the specific mRNA molecules. An mRNA sample from a specific cell population (e.g., a sample from a tumor) is mixed with these probes, and the target mRNA molecules bind to their

respective probe pairs. Following binding unbound probes are removed. The immobilized mRNA-probe complex are counted.

2.4 Study design

The dataset used in this thesis is from a clinical trial, named CORALLEEN, which compared the response of combining a target drug with hormonal therapy against a standard chemotherapy treatment for breast cancer (Prat et al. 2020). Briefly, this is a randomized, multicenter study where the patients are postmenopausal women with the breast cancer subtype luminal B. The cancer subtype diagnosis in this study was primarily based on a well established weighted score of mRNA expression of 50 genes (PAM50 classification) (Wallden et al. 2015; Sørlie et al. 2001). Further inclusion criteria in the study was that the disease was in developmental stage I–IIIA and the tumors were confirmed to be operable. Stage I-IIIA refers to a maximum size of the tumors and whether the cancer has not spread beyond the breast (distant metastasis) or the nearby lymph nodes (draining lymph nodes). Randomisation was stratified to stage I-II or stage IIIA using permuted blocks of 25 with allocation ratio of 1:1.

The tested drug combination, ribociclib and letrozole, is in biological studies characterized to target the intracellular signaling pathway between the estrogen receptor and regulation of the cell cycle (see section 2.2). The control group received chemotherapy consisting of doxorubicin, cyclophosphamide and paclitaxel. The duration of the therapy was 24 weeks and thereafter, within two weeks, the patients went through surgical removal of cancerous tissue. Tissue samples of the tumors were taken at screening, two weeks after the start of treatment and at surgery. mRNA was extracted from the tissue in order to measure changes in gene expression in response to the treatments.

2.5 Study Outcome

The clinical study operated with one primary outcome and multiple secondary outcomes. Here, I present the two outcomes of interest to this thesis. The primary outcome was a three level categorical variable. This were based on an integer score ranging from 0-100, aiming to predict risk of relapse (ROR) (Wallden et al. 2015). This score is reported as a secondary outcome, but we have used this score as it increases the statistical power to detect a relation between the features and the response variable (Altman and Royston 2006). The score aims to predict a risk of less than 10 % of developing distant metastasis at 10 years if treated with local therapy and 5 years of endocrine therapy and without chemotherapy. The score is based on measurements of mRNA expression level of a distinct gene set (the PAM50 genes) and tumor sizes. We also analysed the an outcome named proliferation score. This score was calculated using the mRNA expression level of a subset of the 50 genes that are associated with the cell cycle. From the frequency distribution, we observe that the ROR score is dense around score of 20 while there are few observation above a score of 40 (see Figure 2.2). This compactness of the distribution can potential create a challenge for predictive models. The proliferation score have similar distribution.

2.6 The dataset

In the clinical trial, chemotherapy and targeted drug therapy were compared, however, in this thesis we have only utilized the data from the targeted drug therapy group. In the trial 51 patients receiving chemotherapy and 49 patients receiving the targeted drug therapy had mRNA samples at start and end of the study of high enough quality for adequate analysis. However, 10 patients in the chemotherapy group and 8 in the targeted drug therapy group did not receive the full treatment. In the dataset provided for this thesis that information was not given. Therefore, all patients were processed equally.

2.7 Major findings in clinical trial

At surgery, 24 (46 · 1% [95% CI 32 · 9–61 · 5]) of 52 patients in the chemotherapy group and 23 (46 · 9% [95% CI 32 · 5–61 · 7]) of 49 patients in the ribociclib plus letrozole group showed low-ROR (Prat et al. 2020). Thus, the current effect measurement on cancer outcome is not significantly different between the two treatments. However, the side effect appeared to be lower in the ribociclib plus letrozole group compared to the chemotherapy group.

2.8 Features in the dataset - the signature gene sets

The dataset contains demographics, clinical parameters and mRNA expression of 771 genes in the tumors of the patients at the above mentioned timepoints. In this work the analysis is concentrated on the gene expression data. The genes were pre-selected based on domain knowledge suggesting their involvement in cellular processes relevant for the tumor ecosystem of breast cancer biology (nCounter® Breast Cancer 360™ V2 Panel). More specifically, the total gene set is composed of subsets of genes, where each of these subsets take part in specific cellular functionality such as intracellular signaling, immune activity, regulation of cell division and cell deaths, generation of blood vessels and tumor metabolism. See table 2.1 for a full overview of the 25 gene sets and the number of genes within each of them. From a medical perspective this option for dissecting the role of the different parts of the cancer ecosystem as described in the introduction. We have primarily included this biomedical domain knowledge into the statistical models to reveal its predictive power. However, as some of the methods conduct feature selection, insight into the cancer biology of the tumor ecosystem is also feasible. The sub gene sets are called signature gene sets and they are referred to by the name of the process they are involved in or the name of the particular cells that express the genes. We have studied the gene sets angiogenesis, antigen presentation, apoptosis, cell migration, cytokine and chemokine signaling, DNA damage repair, estrogen receptor signaling, immune infiltration, proliferation and tumor metabolism.

Table 2.1: The signature gene sets

Name	Number of genes
Cell Migration	83
Angiogenesis	34
Antigen Presentation	21
Apoptosis	9
Cytokine and Chemokine Signaling	50
DNA Damage Repair	143
EMT	85
Estrogen Receptor Signaling	27
Epigenetic Regulation	18
Hedgehog	20
Immune Infiltration	34
Internal Reference Gene	18
JAK-STAT	47
MAPK	100
Notch	22
PI3K	96
Proliferation	144
Stromal Markers	6
Subtypes	70
TGF-beta	57
Transcriptional Misregulation	63
Triple Negative Biology	50
Tumor Metabolism	15
Wnt	51
Internal Reference Gene	18

CHAPTER 3

Methods and theory of standard statistical terminology and machine learning models

In this chapter, we introduces standard statistical terminology and present the linear regression model and its extensions ridge regression, lasso and elastic-net. These models extend the standard linear model by incorporating different penalty terms to control the model's complexity, thereby improving its generalization performance (Hastie, Tibshirani and J. Friedman 2009). Furthermore, we outline the general notation, some standard definitions and expressions used in this thesis.

We also review the non-linear ensemble model, boosting, with decision stumps as the base learner. Collectively, ridge, lasso, elastic net, and boosting are considered standard initial approaches for statistical analysis of high-dimensional data (Hastie, Tibshirani and J. Friedman 2009).

Finally, the evaluation strategies we employed to compare the performance of the models are described.

3.1 The linear regression model

Throughout this thesis, we will consider datasets containing a scalar outcome variable y_i and a vector of predictor variables $\boldsymbol{x_i} = (x_{i1}, \cdots x_{ij}, \cdots, x_{ip})$, where $i = 1, \ldots, n$ corresponds to individual patients. Given a dataset of observations $(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_n}, y_n)$, the multiple linear regression model can be defined as

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_j x_{ij} + \cdots + \beta_p x_{ip} + \epsilon_i, \tag{3.1}$$

where y_i is a dependent variable and x_{ij} are the independent variables. The $p + 1$ coefficients β_j, $j = 0, ..., p$ have to be estimated, which is a major goal in this thesis. The last term, ε_i, is the error for the i-th observation. This is a random variable that accounts for variations the model cannot explain and is often placed under assumptions such as having a specific distribution and an expected value equal to zero. The term "multiple" refers to a situation with more than one independent variable. We will mostly use the terms features, predictors and genes when we refer to the independent variables and response or outcome variables when we refer to dependent variables. The matrix form of the multiple linear regression model is given by

$$\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{3.2}$$

where $\boldsymbol{y}$ is the response vector, X is the design matrix, $\boldsymbol{\beta}$ is the coefficient vector and $\boldsymbol{\epsilon}$ is the error vector. This equation can can be written using the notation above as

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1p} \\ 1 & x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}. \tag{3.3}$$

The linear model can be expanded by introducing interactions to account for non-additive relationships among the independent variables. In its simplest form this concept can be exemplified as the situation where the effect of one predictor on a response variable depends on the value of a second predictor. In order to represent the interaction effect a new variable, known as the interaction variable, is typically constructed as the product of the original variables. The interaction variable is then added as an additional term to the linear regression

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i1} x_{i2} + \epsilon_i, \tag{3.4}$$

where β_3 represents the coefficient of the interaction. This equation can be naturally expanded to cases with more than two explanatory variables of interest by constructing multiple interaction variables, with pairwise products representing pairwise-interactions and higher-order products representing higher-order interactions. In general, interaction terms can be challenging to understand and interpret, so they are rarely used with more than three original variables in interaction terms, and most often only two are used (Aiken, West and Reno 1991).

3.2 Basic definitions and terminology

In this section, we present and define general statistical terminology and concepts that are used throughout the thesis.

Mean Squared Error (MSE) is a metric we use to measure the differences between predicted and observed values, in order to assess the quality of the statistical models. MSE is the average squared distance between the predicted values and the observed values. Given a set of predicted values $\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n$ and corresponding observed values $y_1, y_2, \ldots, y_n$, the MSE can be calculated as

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2. \tag{3.5}$$

Loss functions are used to quantify discrepancy between the observed values and the predicted values of a model, serving as a performance measure. The goal in model training is to minimize the loss function by optimizing the model parameters for the given data. Loss functions vary depending on the model. For the ordinary linear regression model it is common to use the residual sum of squares (RSS) as loss function

$$\text{RSS} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \tag{3.6}$$

or in matrix form

$$\text{RSS} = (\boldsymbol{y} - \hat{\boldsymbol{y}})^\top(\boldsymbol{y} - \hat{\boldsymbol{y}}). \tag{3.7}$$

Finding estimates of the βs in the linear regression model, the $\hat{\beta}$s, then becomes the solution to this problem

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n}(y_i - \beta_0 - \boldsymbol{\beta}^\top \boldsymbol{x}_i)^2 \right\}, \tag{3.8}$$

or in matrix form

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \|\boldsymbol{y} - X\boldsymbol{\beta}\|_2^2 \right\}, \tag{3.9}$$

The closed form solution to the optimization problem is given by

$$\hat{\boldsymbol{\beta}} = (X^\top X)^{-1} X^\top y.$$

High-dimensional data refers to datasets which contain large number of feature variables relative to the number of observations. Mathematically it is defined in Hastie, Tibshirani and J. Friedman 2009 as the situation where $p \approx n$ or $p > n$.

Sparsity refers to the situation where only a small proportion of the covariates contain significant or non-zero information about a dependent variable, while the rest contribute with little to no information (Hastie, Tibshirani and J. Friedman 2009).

In the context of high-dimensional modeling, the phrase "**bet on sparsity**" refers to the assumption that the underlying true model is sparse. Assuming sparsity suggests that only a minor subset of the features contributes significantly to the prediction of the response variable and, thus, can be leveraged in model building to simplify models and improve interpretability.

L1 norm is a measurement of a vector magnitude, sometimes called the Manhattan distance or taxicab norm because it measures the distance between two points in a grid-like pattern, where one can only move horizontally or vertically, but not diagonally. It is defined as the sum of the absolute values of its elements. Given a vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, the L1 norm is defined

$$||\boldsymbol{x}||_1 = \sum_{i=1}^{n} |x_i|. \tag{3.10}$$

L2 norm, known as the Euclidean norm, is another measure of the size of a vector. It is defined as the square root of the sum of the squares of its elements

$$||\boldsymbol{x}||_2 = \sqrt{\sum_{i=1}^{n} x_i^2}. \tag{3.11}$$

Overfitting occurs when a model is too complex and fits the training data too close, capturing also noise instead of primarily focusing on the underlying

pattern in the data. As a result, the model performs poorly on new, unseen data. **Underfitting** is the opposite of overfitting and occurs when a model is too simple to accurately capture the underlying structure or patterns in the data, leading to bias.

Regularization is a technique that controls a model's complexity (encourage sparse feature space), through the size and number of parameters in order to prevent overfitting by incorporating a penalty term into the loss function. Two prevalent methods include L1 regularization and L2 regularization, which use the L1-norm and the squared L2-norm of the parameter vector as penalty terms, respectively. These examples, known as lasso and ridge, will be discussed below.

Stability refers to the model's sensitivity to small changes in the training data. Stable models maintain consistent performance and similar parameter estimates when conducted to minor perturbed training data. Stability can be improved by employing regularization, which reduces the variance in model estimates.

Reliability is the degree to which a model produces consistent and accurate results when applied to various datasets. A reliable model exhibits both stability and good generalization performance, making it suitable for use in practice.

Decision stump or just stump is the simplest form of a decision tree. A decision tree is a hierarchical machine learning model. A decision tree consists of nodes and branches, with each node representing a feature and each branch representing a decision based on that feature. A split is the process of dividing a node into two or more child nodes based on a threshold for a specific feature value. A stump, in this context, is the simplest decision tree with only one split for a single feature.

3.3 Ridge regression

In ridge regression, the objective is to minimize the RSS between the predicted values and the true values of the training data subjected to a regularization (Hoerl and Kennard 1970). The regularisation is defined by constraining on the square of the L2 norm of the coefficient vector $\boldsymbol{\beta}$. The constrain can be defined as $\|\boldsymbol{\beta}\|_2^2 < s$ for some s, but it is most common to present the ridge regression problem as the solution to

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} (y_i - \beta_0 - \boldsymbol{\beta}^\top \boldsymbol{x}_i)^2 + \lambda \|\boldsymbol{\beta}\|_2^2 \right\}, \tag{3.12}$$

or in matrix form

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \|\boldsymbol{y} - X\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_2^2 \right\}, \tag{3.13}$$

where $\lambda > 0$ is a hyperparameter controlling the strength of regularization. The penalty term has the effect of shrinking the magnitude of the coefficient estimates towards zero without setting them exactly to zero as in the case of the L1 penalty in lasso (see next section). As λ increases, the magnitude of the coefficients is shrunk towards zero. λ is typically tuned on a different dataset as the one used to learn the model.

The Ridge regression model can be fit using various optimization algorithms, such as gradient descent, but it also has the closed-form solution. The optimal

value of λ can be determined using techniques such as cross-validation combined with grid search. The closed-form solution to the optimization problem is given by

$$\hat{\boldsymbol{\beta}} = (X^\top X + \lambda I)^{-1} X^\top y,$$

where I is the identity matrix. However, it is not common to solve ridge by closed form calculations in algorithms due to the computational complexity and numeric instability of matrix inversion.

3.4 Lasso regression

Least absolute shrinkage and selection operator (lasso) is a linear regression model that uses the L1 norm of the coefficient vector $\boldsymbol{\beta}$ as regularization (Tibshirani 1996). Interestingly, this will in addition to penalize the size of the β's also cause selection of the β's, which encourage sparsity in the feature space of the learned model (discussed further below). The lasso optimization problem is the solution to the problem

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \|\boldsymbol{y} - X\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_1 \right\}, \tag{3.14}$$

The equation is identical to ridge besides that the penalty terms is based on the L1 norm instead of the L2 norm. The λ parameter enables a desired level of sparsity and is typically tuned on a different dataset as the one used to learn the model. Lasso introduces a bias in the estimates of the coefficients in exchange for reducing the variance of the estimates. The strength of the regularization determines the trade-off between bias and variance.

3.5 Elastic net regression

The elastic net model combines the L1 and L2 penalties and is thus a blend between the lasso and ridge models (Zou and Hastie, 2005). It is often used in situations where there are many features where some are highly correlated. The elastic net problem has the following optimization problem

$$\hat{\boldsymbol{\beta}} = \arg\min_{\beta} \left\{ \|\boldsymbol{y} - X\boldsymbol{\beta}\|_2^2 + \lambda(\alpha \|\boldsymbol{\beta}\|_2^2 + (1-\alpha)\|\boldsymbol{\beta}\|_1) \right\}. \tag{3.15}$$

The first term in loss function is the well known RSS, the second term combines the L2 penalty and L1 penalty. $\alpha \in [0, 1]$ is a hyperparameter which controls the balance between the regularization terms. When $\alpha = 0$ elastic net reduces to lasso, and when $\alpha = 1$, it reduces to ridge. In general, a large value of alpha gives more weight to the L2 penalty, while a smal value gives more weight to the L1 penalty. α is often selected using cross-validation combined with grid search techniques, but sometimes it is just set, typically to 0.5.

3.6 Comparison of the ridge and lasso penalties

The choice between ridge and lasso penalty in linear regression analysis depends on the specific characteristics of the dataset and the goals of the analysis.

The L1 penalty encourages the model to use only a subset of the available features by setting some entries of the estimated β coefficients to exactly zero, thereby performing variable selection. To understand how the L1 penalty achieves this, we can visualize the loss function in a contour plot with two features (see Figure 3.1). In this case, the β coefficients are on the x-axis and y-axis and the contour lines represent regions with the same value. The RSS term will create elliptic contour lines with center at its lowest value, with gradually larger values as the ellipses increase in size. The L1 term will create diamond shaped contour lines with center at the origin. For the β values to be the same in the two terms the counter lines must intersect. Although the contour lines don't need to have the same loss value where they intersect, the objective of lasso is to minimize the sum of the contour values at the meeting point represented by the β values. It is now possible to see that in a large part of the β space the contour lines will intersect along one of the axes, giving zero value to the β coefficient represented by that axis. In contrast, the L2 penalty produces circular contour lines centered around the origin, with no corners protruding like those in the L1 penalty. Consequently, the likelihood for the two sets of contour lines to intersect along an axis is not higher than at any other location in the L2 penalty contour lines.

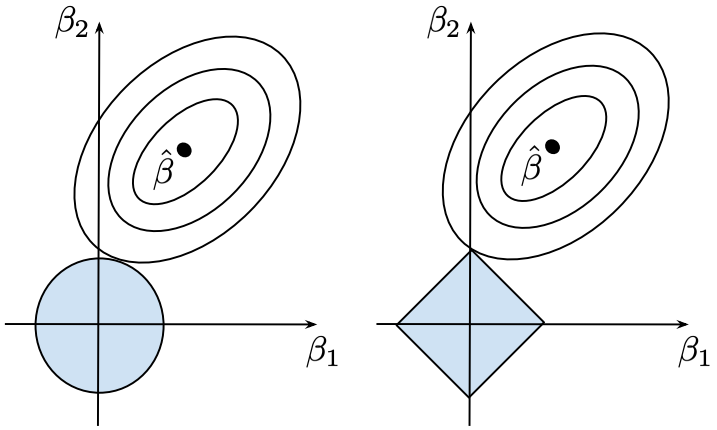


Figure 3.1: Figurative visualization of the differences between the L2 regularization (left) and the L1 regularization (right). Modified from Hastie, Tibshirani and J. Friedman 2009

Lasso regression is often preferred when the number of predictor variables is high, and many of them may be irrelevant or redundant (a "bet on sparsity" situation). In such cases, lasso's L1 regularization can effectively set the coefficients of irrelevant variables to zero, resulting in a simpler and more interpretable model. Moreover, when there is an absolute need to select a small subset of important predictor variables for the model lasso is the obvious choice. However, multicollinearity can be a challenge for lasso. Multicollinearity occurs when the independent variables in a regression model are highly correlated, making it challenging to distinguish the individual effect of each variable on the dependent variable. In such situations, a lasso model will typically select one among the correlated variables. This can lead to instability in model choice

and predictive power. In general, due to the encouragement of sparsity, lasso performs well when a small number of predictors have large coefficients and the remaining predictors have small coefficients. However, if all predictors have relatively large coefficients, the lasso penalty can be too severe and randomly selecting some predictors, which can lead to underfitting. In such cases a lasso model may not capture the underlying relationships between the predictors and the response variable adequately, resulting in poor generalization to new data.

On the other hand, ridge regression is often preferred when the goal is to obtain the best prediction accuracy. This is particularly true when in scenarios where all predictors have significant coefficients, but also even if some of them are not important or relevant. Ridge's L2 regularization shrinks the coefficients of all variables towards zero, but not exactly to zero, which can help to prevent both underfitting and overfitting and, thus, improve the stability of the model. Specifically, ridge regression can be more stable than lasso when the predictor variables are highly correlated. Ridge is suggested to be well adapted to handle the problem of multicollinearity in high dimensional data (Hoerl and Kennard 1970). Ridge regression reduces the variance in the estimated coefficients, which can lead to more stable predictions compared to other machine learning models.

What benefits do we get out of combining the two penalty terms in the elastic net is a relevant question. Obviously, when there is a need for both variable selection and accurate prediction, elastic net offers a balanced solution by incorporating both types of regularization. Elastic net can handle highly correlated predictors better than lasso as elastic net can shrinks the coefficients of correlated predictors together. Thus, with respect to selection between correlated features in elastic net the L2 term encourages highly correlated features to be averaged, while the L1 term encourages a sparse solution in the coefficients of these averaged predictors (Zou and Hastie 2005). Therefore, elastic net can in some situations lead to more stable and interpretable models than those produced by lasso or ridge alone.

In conclusion, statisticians commonly regard ridge regression as a safer choice for prediction purposes as it generally increase the chance of prediction accuracy. On the other hand, if the primary goal is feature selection and interpretability, lasso is the preferred choice. However, when both prediction and variable selection are of interest, elastic net emerges as a compelling and relevant solution. In practice, it is often a good idea to try all methods and compare their performance using appropriate evaluation metrics and cross-validation techniques.

3.7 The glmnet package used for ridge, lasso and elastic net

In the R-package glmnet (J. H. Friedman, Hastie and Tibshirani 2010; Tay, Narasimhan and Hastie 2023), the coordinate descent algorithm is used to solve a sequence of regression problems with different values of the regularization parameter λ. The algorithm starts with a high value of λ, where all coefficients are set to zero, and then gradually decreases λ, allowing the algorithm to introduce more and more input features into the model.

The optimization problem of lasso is a convex problem and even though the L1 penalty term is non-differentiable at zero it can be solved by gradient-based

methods such as subgradient descent or proximal gradient descent. However, non-gradient numerical optimization techniques also exist e.g. the LARS algorithm. In my thesis I have used the R packed glmnet which uses a highly optimized implementation of the coordinate descent algorithm for ridge and lasso that was developed by J. H. Friedman, Hastie and Tibshirani (2010). It is not strictly a gradient-based method, but it can be viewed as a generalization of gradient descent that operates on a single coordinate at a time. In traditional gradient descent, the algorithm updates the model parameters using the gradient of the loss function with respect to all the parameters at once. In contrast, in coordinate descent, the algorithm updates the model parameters one coordinate at a time while holding all other coordinates fixed. At each iteration, the algorithm identifies the coordinate that can be updated to achieve the largest reduction in the loss function, and then updates that coordinate by a specific amount. In the case of the lasso, where the L1 term is not differentiable at zero, still as the function is convex the gradient exists outside zero and it is a constant value of either -1 or 1, depending on the sign of the corresponding coefficient.

The glmnet algorithm also employs a rule called the "strong rule" to remove the coordinate from the active set of variables and thus avoid updating it in subsequent iterations (Tibshirani et al. 2010). The strong rule works by checking whether the absolute correlation between the response variable and the predictor is less than a certain threshold value. If the absolute correlation is less than the threshold, the coordinate is set to zero and removed from the active set, and subsequent iterations of the algorithm do not update that coordinate. The threshold value used in the strong rule can be chosen based on the numerical precision of the computation and the desired level of sparsity in the solution.

3.8 Boosting with stumps

Boosting is an ensemble model which combines multiple weak learners to form a strong model (Hastie, Tibshirani and J. Friedman 2009). An ensemble model is an aggregate of several different models. A weak learner in this setting is a model that perform just slightly better then a random guess. The weak learners are often called base learners, or base models and are typically simple statistical models.

The general idea underlying boosting is to iteratively train a set of such weak base learners and then add them together in a final model. Within each iteration higher weights are assigned to the data points that were less well learned in the previous iteration steps. In this way, the subsequent base learner focuses more on the data that were difficult to learn by the previous base learners. Thus, each new base learner does not disturb the set of learners created in the previous steps, but it transmits some more information from the data space into the total model in order to reduce the error. Therefore, as the different base learner focus on various aspects of the data the final model will capture various patterns in the data space.

The weak base learners are simple models which most often only use a subset of the feature space. In the simplest case there is one base learner for each feature. Many different models can be used as weak learners, e.g. it can be decision stumps or simple linear models, but also more flexible models as

splines are sometimes used. However, to maintain a week learning model these need to be defined with small degrees of freedom (Hofner et al. 2012). It is also an option to combine different base learners, which in principle means that an additive model is built iteratively. The final model is an average of the base learners.

Each of the base learners has high bias but low variance, which causes them to underfit the data. However, the combination of them reduces bias as they handle various aspects of the data space. Furthermore, as boosting takes the weighted average of many models the final model has lower variance than each of the base models since the random errors in the individual models typically are canceled out.

We have used regression stumps as base learners as this gives the opportunity to learn non-linear data and then becomes a supplement to the other models we have tested that focus on linearity.

In the algorithms of xgboost, the gradient of the loss function is used to train new base learners (Chen and Guestrin 2016). In the case of regression, where the loss function is the RSS, the gradient becomes the residual between model prediction and the response in the training data. The iterative process is outlined in Algorithm 1. To learn the set of base functions used in the model, xgboost minimize the following loss function

$$L = \text{RSS} + \sum_{m=1}^{M} \Omega(f^{(m)}), \tag{3.16}$$

where M is the number of trees in the ensemble. The term $\Omega(f^{(m)})$ represents the regularization for tree m. The penalty term in xgboost is formulated as

$$\Omega(f^{(m)}) = \gamma T + \frac{1}{2}\lambda ||w||_2^2, \tag{3.17}$$

where γ is a hyperparameter that controls the complexity of the tree structure through the number T of terminal nodes. λ is the L2 regularization hyperparameter and w is a vector of terminal node weights.

Algorithm 1 Component-wise gradient boosting with stumps

1. Initialize with offset value $f_0(x) = \frac{1}{n}\sum_{i=1}^{n} y_i$.
2. For $m = 1$ to M or early stopping criteria is met:
   a) Compute the negative gradients r_i of the loss function, L and evaluate it at the previous iteration step, $\hat{f}_i^{(m-1)}(x_i)$ (i.e. at the estimate of the previous iteration).
   $$r_i^{(m)} = -\frac{\partial L(y_i, \hat{f}_i^{(m-1)}(x_i))}{\partial \hat{f}_i^{(m-1)}(x_i)}.$$
   b) For each $j = 1, 2...., p$ features fit a stump $h_j^{(m)}$ using r_i as response variable (xgboost also uses a stochastic feature selection process here).
   c) Select the stump that improves the model most
   $$h_m = \arg\min_{h_j^{(m)}} \sum_{i=1}^{n} L(y_i, \hat{f}_i^{(m-1)}(x_i) + h_j^{(m)})$$
   d) Update the model by adding the new stump to the ensemble $\hat{f}(x_i)^{(m)} = \hat{f}(x)_i^{(m-1)} + \eta h^{(m)}(x_i)$, where $0 < \eta \leq 1$ is learning rate.
3. The final ensemble model becomes $f_0(x) + \sum_{m=1}^{M} \eta h_m(x)$.

3.9 Model comparison and assessments of predictive performance

Unfortunately, the dataset is so small that we found it unfeasible to split it into a training and a test set. To compare and evaluate the usefulness of the different models, we instead utilized two resampling techniques: bootstrapping and repeated cross-validation, which will be described in more detail below. Moreover, in order to assess the accuracy of the predicted values $\hat{y}_i$ in the test set of size n, we employed the Pearson correlation coefficient to linearly compare them with the true values y_i

$$r = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{\hat{y}})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(\hat{y}_i - \bar{\hat{y}})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}. \tag{3.18}$$

where $\bar{y}$ and $\bar{\hat{y}}$ are the mean of y_i and $\hat{y}_i$, respectably.

Additionally, we used MSE to measure of accuracy of the models

Bootstrapping

In non-parametric bootstrapping, multiple random samples are drawn with replacement from a dataset. Typically, each sample includes the same number

of observations as in the original dataset. These samples are often used to compute a statistic in order to acquire an empirical approximation of the sampling distribution of that statistic. We have utilized bootstrap sampling to generate multiple training samples for our models. Specifically, we learned 1000 fits of a model of interest by training on 1000 bootstrap samples from the original dataset. Each model was evaluated on the original data set, which means that the bootstrap samples acted as the training data and the original dataset acted as the test data. With this approach training and test data will have common observations, which typically can lead to overfitting. However, if assuming that the structure in the dataset is an adequate representation of the population this can be used to evaluate the relative differences in performance between models (Hastie, Tibshirani and J. Friedman 2009). While this bootstrapping approach can be useful in model selection it doesn't give satisfactory information about a model's performance on future prediction.

Repeated Cross-validation

In k-fold cross-validation the original dataset is first randomly split into k subsets, called folds, of roughly equal size. A separate model is trained on the data from k-1 folds, and tested on the remaining fold not used for the model training. The process is repeated k times, with a different fold serving as the test set each time. The overall performance measure is then calculated as the average performance across all the k iterations. When splitting a dataset into a learning and evaluation set, as in cross-validation, the results between different splits will typically vary significantly when the dataset contains few observations, as it becomes sensitive to the partition of the dataset. To address this instability problem we repeated the cross-validation 200 times, when using k=5.

CHAPTER 4

Naive analysis

In this section we present the results from utilizing a set of standard machine learning model classes often used to analyse high dimensional data. These are ridge, lasso, elastic net and boosting with stumps as base learners as described in chapter 3. All the 771 gene expression values in the dataset from the clinical trial were used as predictors, while both the proliferation score and the ROR score were used as response variable in separate analyses.

The standard approach for evaluating machine learning models involves dividing a dataset into a training set and a testing set, fitting models on the training set, and then evaluating the model performance on the test set. However, in this thesis, we only have 49 patients in our dataset. With such a small sample size, the model fit and the accompanied performance evaluation is highly sensitive to the random split of data into training and test set. Therefore, we have employed two evaluation approaches typically used with small datasets (see Section 3.9).

The first approach involves fitting models on 1000 bootstrap samples and evaluating model performance using the full original dataset. The bootstrapping method employed in this thesis will, on average, give bootstrap samples that contain approximately 63.2% of the original dataset. Consequently, the models are fitted on an average of 31 patients ($49 \cdot 0.632$) and tested on all the 49 patients. In the second approach, we preformed five-fold cross-validation, repeated 200 times. This results in the generation of 1000 models, which are trained on 39 patients and tested on 10 patients. Thus, the bootstrap method in addition, to the overlap in training and test sets, providing a substantially larger number of observations in the test set than for the repeated cross-validation. Due to these factors, the repeated cross-validation method is a more conservative approach and in addition may give higher variability in comparison to the bootstrap method. The latter we did observe in our results.

During model training the hyper-parameters are selected with five-fold cross-validation, which is built into the algorithm of these machine learning classes. The models were chosen based on their specific hyper-parameters that corresponded to the lowest MSE obtained during the cross-validation process.

4.1 Ridge

Ridge regression was fitted using both the bootstrap and the repeated cross-validation methods. We utilized the proliferation score and the ROR score as

response variables, respectively. In this section, we first present the results related to model coefficient sizes, followed by the performance measures of the ridge models.

Coefficient sizes

To obtain an overview of the estimated coefficients' sizes, we plotted a histogram of the mean sizes over the model fits for all 771 coefficients (Figure 4.1). We only display the histograms for the bootstrap approach, but similar distribution were observed for repeated cross validation.

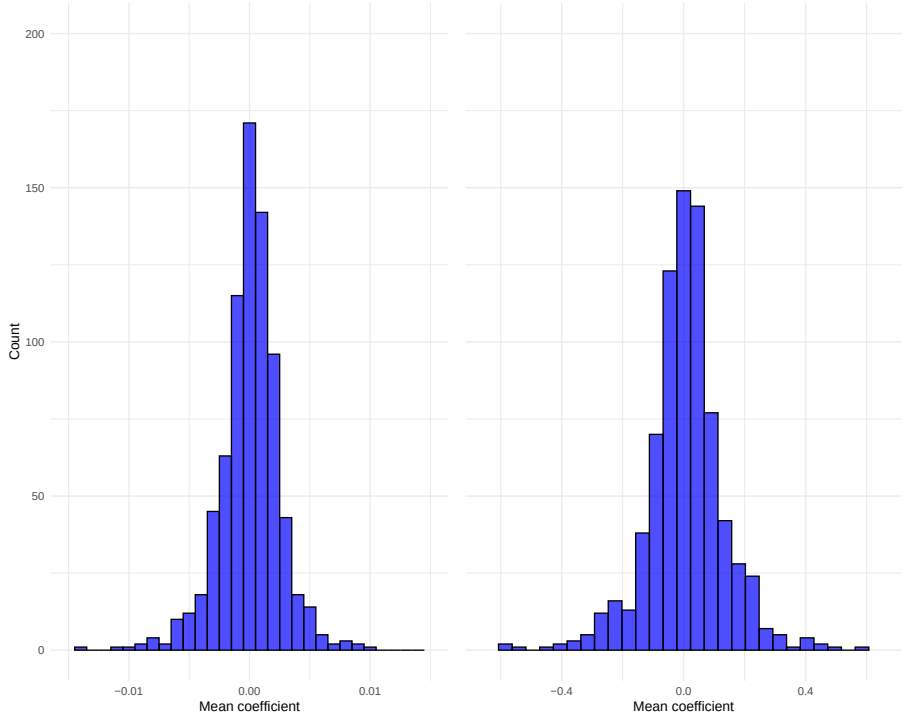


Figure 4.1: Histogram of the mean of each coefficient's size from 1000 bootstrap models of the ridge regression using the proliferation score (left histogram) and ROR score (right histogram) as response variables

Next, to gain information about the most important features estimated by the ridge regression, we selected the 20 coefficients with the largest absolute values. In Figure 4.2 a box plot of these coefficients for the bootstrap approach with the proliferation score as response are show. This provides some insights into the central tendency, variance and skewness of the estimated coefficient values for the most important predictor variables. The distribution of the coefficient values appears nearly symmetric for most genes, however, we observe some variability in the standard error of the coefficients. Notably, none of the genes show a substantially high coefficient value compared to the others, indicating that no individual gene stand out with a particularly strong influence

on the proliferation score, even though a couple of them seem to be significantly different from 0.

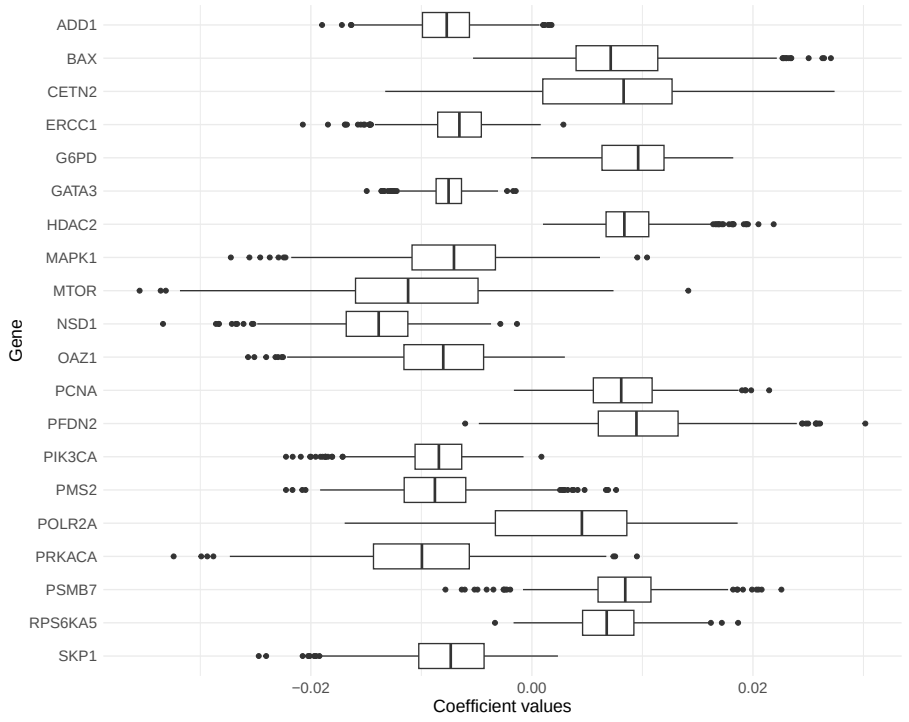


Figure 4.2: The distribution of the 20 largest mean absolute estimated coefficient sizes across 1000 ridge models. The models are fitted using 771 gene expressions values as features and the proliferation score as response variable. Bootstrapping was used to generate training samples and the full dataset was used to test the models. The horizontal axis contains the gene name abbreviations. The x-axis represents the coefficient values for the corresponding genes. The boxes represent the interquartile range, which contains the middle 50% of the data. The left most edge of the box is the 25th percentile, and the right most edge is the 75th percentile. The horizontal line inside the box is the median. The whiskers extend to the minimum and maximum data points within 1.5·interquartile range outside the box. Any data points outside the whiskers are plotted as individual points.

We also conducted box plots for the three other combinations of sampling approaches and response variable (Figures 4.3, 4.4, 4.5). We observed similar results as for bootstrap with proliferation score.

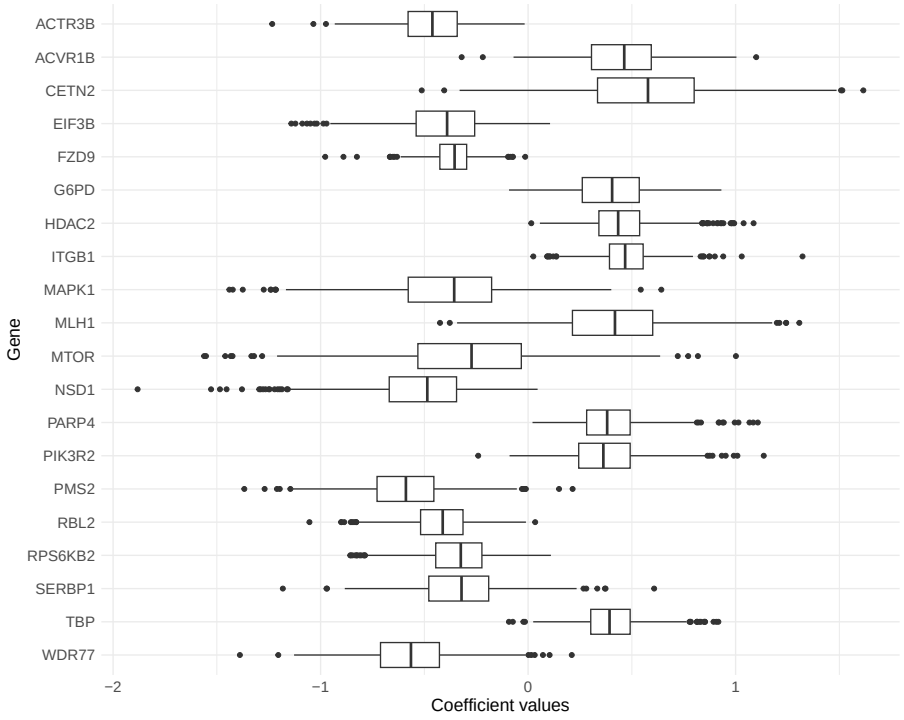


Figure 4.3: The distribution of the 20 largest mean absolute estimated coefficient sizes across the ridge models when using ROR score as response variable. The bootstrapping sampling approach was used.

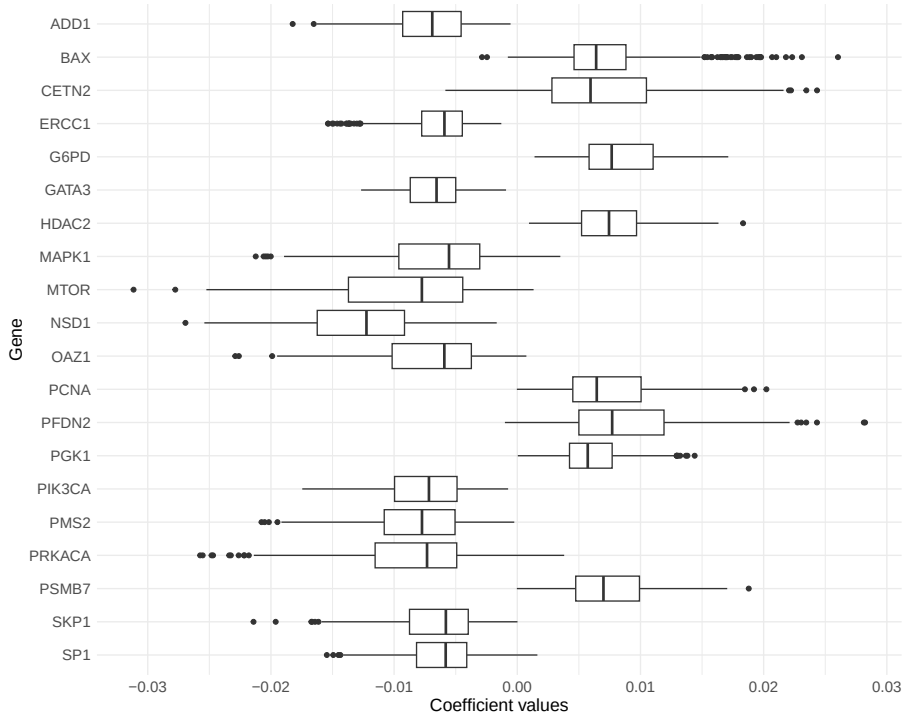


Figure 4.4: The distribution of the 20 largest mean absolute estimated coefficient sizes across the ridge models using the proliferation score as response variable. The repeated cross-validation was used to generate 1000 training and test datasets.

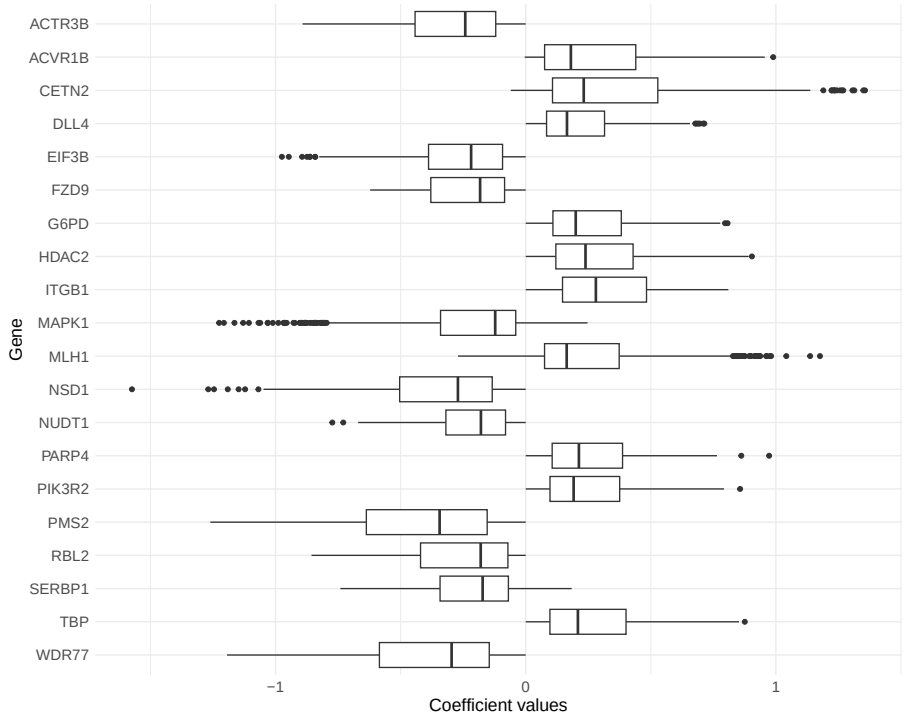


Figure 4.5: The distribution of the 20 largest mean absolute estimated coefficient sizes across the ridge models using the ROR score as response variable. The repeated cross-validation was used to generate 1000 training and test datasets.

In Table 4.1, the 20 largest mean absolute estimated coefficient sizes across 1000 ridge models for the four different combinations of sampling approaches and response variables are summarized. In total, 37 genes were selected, with seven genes (CETN2, G6PD, HDAC2, MAPK1, NSD1, PMS2, and PSMB7) appearing in all combinations. Importantly, when comparing the two different sampling approaches with the proliferation score as the response variable, there was an overlap of 18 genes. Similarly, for the ROR score, 14 genes overlapped. Thus, we observed more overlap when considering the response variable than sampling approach. This is of course expected, as different genes should have varying effects on different outputs. In conclusion, the relatively large overlap between the different sampling approaches for the same response lends confidence that these genes are of significance for the response.

Table 4.1: Coefficients of the ridge models

Gene	Bootstrap (mean (SD))		Repeated CV (mean (SD))	
	Proliferation score	ROR score	Proliferation score	ROR score
ACTR3B		-0.464 (0.173)		-0.297 (0.218)
ACVR1B		0.45 (0.2)		0.267 (0.233)
ADD1	-0.00775 (0.00319)		-0.00712 (0.00308)	
BAX	0.00786 (0.00547)		0.00705 (0.00363)	
CETN2	0.00711 (0.00738)	0.563 (0.342)	0.00654 (0.00565)	0.352 (0.315)
DLL4		0.332 (0.143)		0.208 (0.158)
EIF3B		-0.404 (0.206)		-0.26 (0.201)
ERCC1	-0.00676 (0.00311)		-0.00635 (0.00251)	
FZD9		-0.361 (0.101)		-0.222 (0.162)
G6PD	0.0092 (0.00357)	0.4 (0.186)	0.00843 (0.0033)	0.262 (0.195)
GATA3	-0.00757 (0.00181)		-0.0068 (0.00224)	
HDAC2	0.00878 (0.00311)	0.451 (0.159)	0.00756 (0.00288)	0.28 (0.193)
ITGB1		0.474 (0.131)		0.312 (0.196)
MAPK1	-0.00729 (0.00554)	-0.382 (0.298)	-0.00651 (0.00454)	-0.224 (0.253)
MLH1		0.414 (0.293)		0.253 (0.239)
MTOR	-0.0109 (0.00714)		-0.0091 (0.00563)	
NSD1	-0.0142 (0.00407)	-0.52 (0.257)	-0.0127 (0.00448)	-0.333 (0.25)
NUDT1				-0.207 (0.151)
OAZ1	-0.00828 (0.00514)		-0.00692 (0.00405)	
PARP4		0.395 (0.157)		0.252 (0.177)
PCNA	0.00836 (0.00384)		0.00733 (0.00356)	
PFDN2	0.00975 (0.00559)		0.00872 (0.00472)	
PGK1			0.0061 (0.00245)	
PIK3CA	-0.00864 (0.00338)		-0.00748 (0.00319)	
PIK3R2		0.37 (0.19)		0.245 (0.185)
PMS2	-0.00862 (0.00443)	-0.592 (0.208)	-0.00819 (0.00389)	-0.403 (0.289)
PPP2R1A	-0.00649 (0.0027)			
PRKACA	-0.00994 (0.00662)		-0.00837 (0.00494)	
PSMB7	0.00838 (0.00389)	0.33 (0.22)	0.00748 (0.00343)	
RBL2		-0.421 (0.158)		-0.253 (0.217)
RPS6KA5	0.00685 (0.00331)			
RPS6KB2		-0.34 (0.17)		-0.207 (0.173)
SERBP1				-0.219 (0.18)
SKP1	-0.00755 (0.00431)		-0.00657 (0.00328)	
SP1			-0.00623 (0.00285)	
TBP		0.4 (0.15)		0.253 (0.188)
WDR77		-0.566 (0.214)		-0.369 (0.273)

Performance measure

To evaluate the performance of the models we analysed the correlation between the predicted outcomes and the observed response values in the dataset, under both the bootstrap regime and the repeated cross validation regime (see table 4.2). For the bootstrap regime, the mean correlation was 0.82 (SD = 0.071) for the proliferation score and 0.78 (SD = 0.077) for the ROR score. The MSE between predicted and observed response was considerably higher for the ROR score than for the proliferation score, which is consistent with the different scales of these two scores.

When we applied the repeated cross-validation approach the performance of the models was lower. We obtained substantially lower correlation values for both the proliferation (0.53, SD=0.207) and ROR scores (0.33, SD=0.262). The MSE was also approximately three times higher.

The lower correlation and higher MSE for the repeated cross-validation approach compared to the bootstrapping approach is as expected since there is

an overlap in training and testing data for the bootstrap method while this in not the case for repeated cross-validation. Explaining the reduced performance of the models when using the ROR score as a response variable in comparison to the proliferation score is more difficult. However, the proliferation score is based solely on transcriptomic data while the ROR score also incorporates clinical findings. This difference in data types could potentially account for the observed difference in performance. Furthermore, from a biological perspective, cell proliferation is a less complex process compared to disease outcomes, were cancer cell proliferation is just one of many contributing factors. This increased complexity may make prediction more challenging.

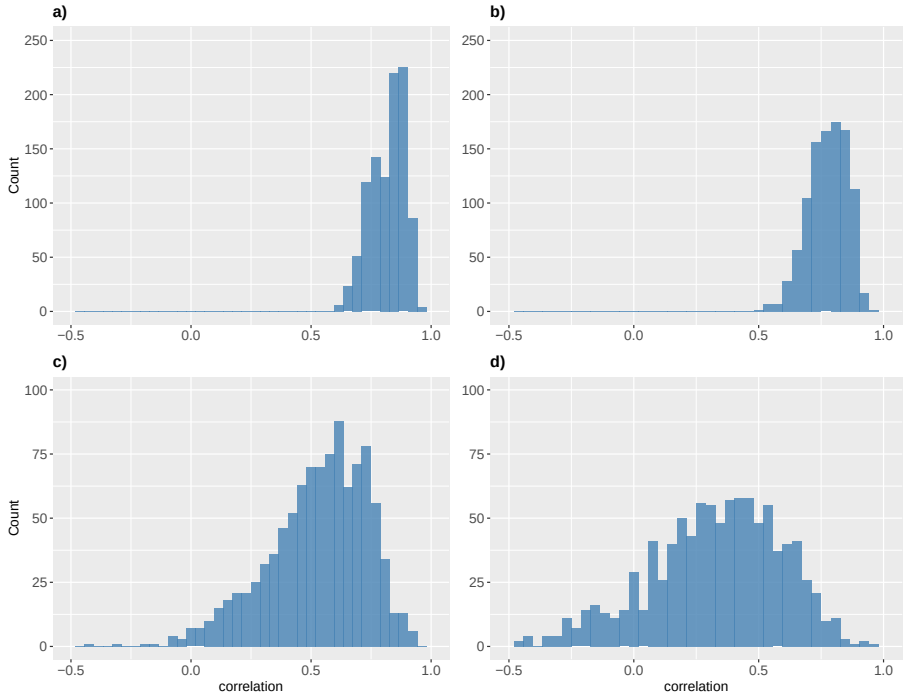


Figure 4.6: The ridge model using 771 genes as features. The histograms show the correlation between predicted response and the observed values in test sets. In a) and b) bootstrapping is used to generate training samples and the full dataset is used to test the models. In a) the response variable is the proliferation score while in b) the ROR is the response variable. In c) and d) repeated cross-validation is used to generate multiple training and testing datasets. In c) the response variable is the proliferation score while in d) the ROR is the response variable. When comparing the results of the two different regimes for model evaluation it is important to have in mind that the test set of the bootstrap method had 49 patients, while for the repeated cross-validation it was only around 10 patients, and 10 is a rather small number to use for correlations.

Table 4.2: Ridge performance summary

Model	Correlations (SD)	MSE (SD)
Proliferation (bootstrap)	0.82 (0.071)	0.057 (0.019)
ROR (bootstrap)	0.78 (0.078)	156 (44)
Proliferation (repeated cross-val.)	0.53 (0.207)	0.13 (0.070)
ROR (repeated cross-val.)	0.33 (0.262)	357 (150)

4.2 Lasso Regression

Lasso regression was applied using both the bootstrap and repeated cross-validation methods (section 3.9). We utilized the proliferation score and the ROR score as response variables in separate analyses. In this section, we first present the results related to the selection of the features (the genes) and then discuss the performance evaluation of the lasso model.

Feature Selection

For the proliferation score, 67 genes were selected at least 10% of the time in the 1000 bootstrapped fitted models (Figure 4.2). The genes CACNA1H, EFNA3, GATA3, and LEFTY2 were selected 50% or more of the times. The top 20 most selected genes were LEFTY2, GATA3, CACNA1H, EFNA3, HOXA9, CAMK2B, BMPR1B, NSD1, CA12, HOXA7, JAG1, APOE, PLA2G2A, TAPBP, S100A7, CALML5, HDAC2, CHIT1, CBLC, and FGF13.

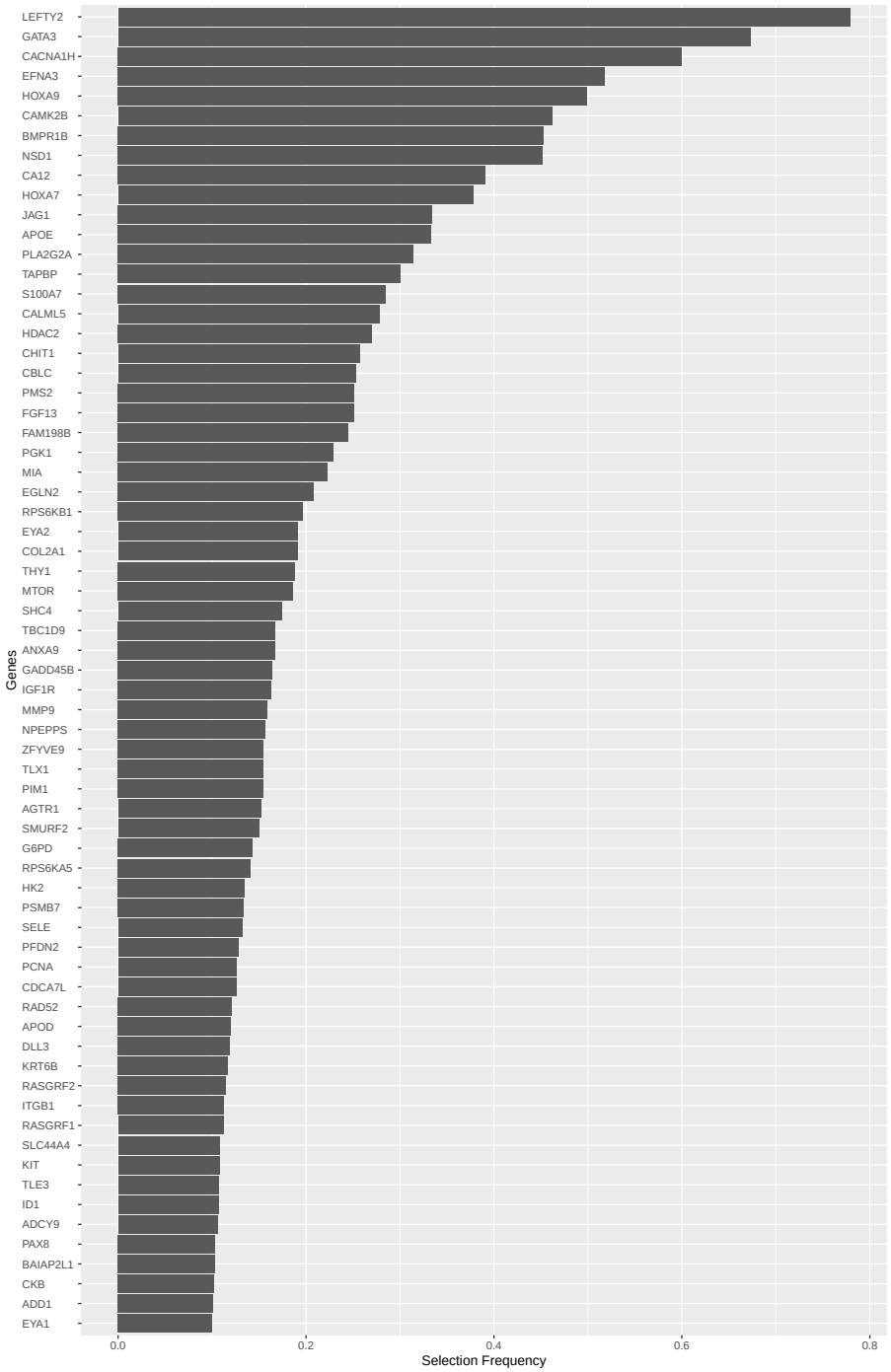


Figure 4.7: Genes selected in more then 10% of the lasso bootstrap models using proliferation score as response variable

The coefficient values for these 20 gene expressions are shown in figure 4.8.

We observe that GATA3, LEFTY2, HOXA9, and NSD1 have substantially higher absolute median values compared to the other genes, suggesting a significant effect of the genes on the proliferation score. However, NSD1 was selected less than 40% of the time, while the other two were selected more than 70% of the time. HOXA9 was selected almost 50% of the time.

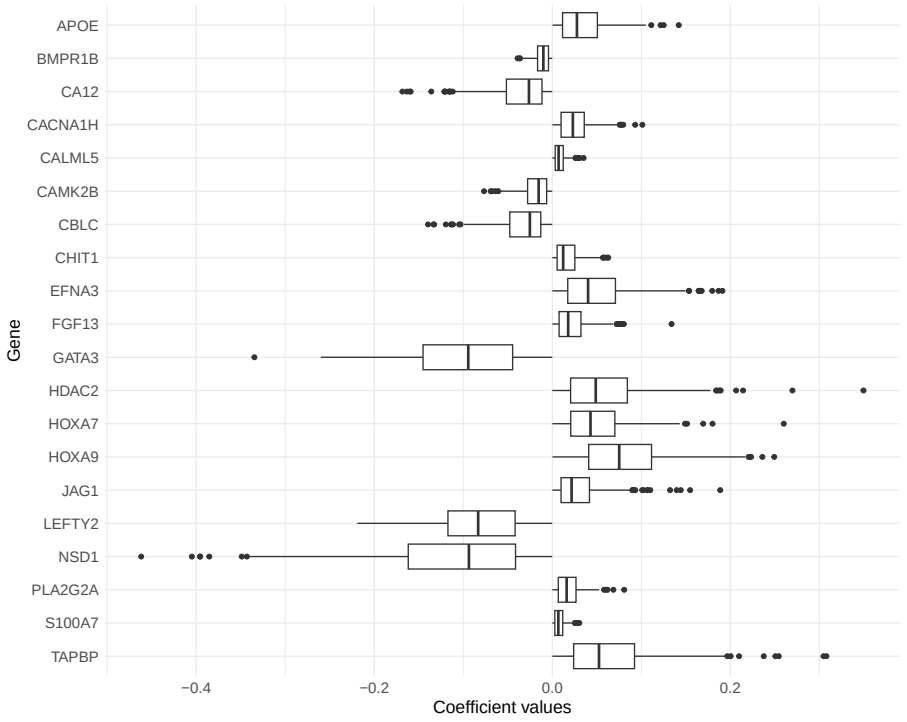


Figure 4.8: The coefficient values of the 20 most selected genes by the lasso models using the proliferation score as response variable (bootstrap sampling technique).

For the ROR score analysis, 62 genes were selected at least 10% of the time in the 1000 bootstrapped fitted models (Figure 4.9). Only one gene, CHIT1, was selected more than half the time. Among the 20 most selected genes, nine genes (CHIT1, LEFTY2, CA12, CACNA1H, HOXA7, EFNA3, APOE, CETN2, HDAC2) overlapped with the top 20 most selected genes when utilizing the proliferation score. Since the ROR score takes into account cancer cell proliferation, these genes potentially play an important role in regulating cell proliferation. In figure 4.10, a box plot of the coefficient values of the top 20 genes selected using ROR score are shown. The PMS2 gene had a substantially larger coefficient (in absolute value) than the other genes. PMS2 was selected in approximately one-third of the models (see Figure 4.9).

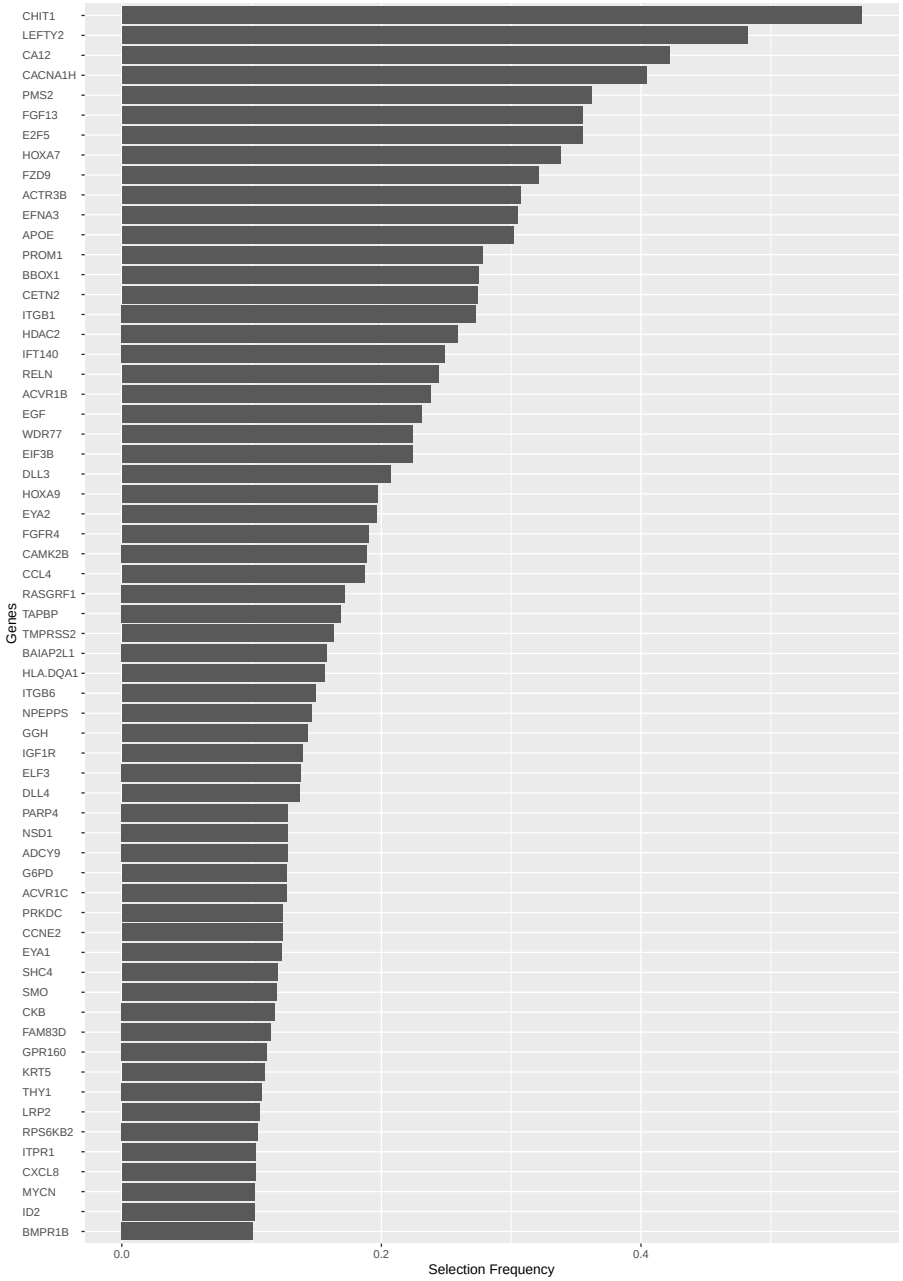


Figure 4.9: Genes selected in more then 10% of the lasso bootstrap models using ROR score as response variable

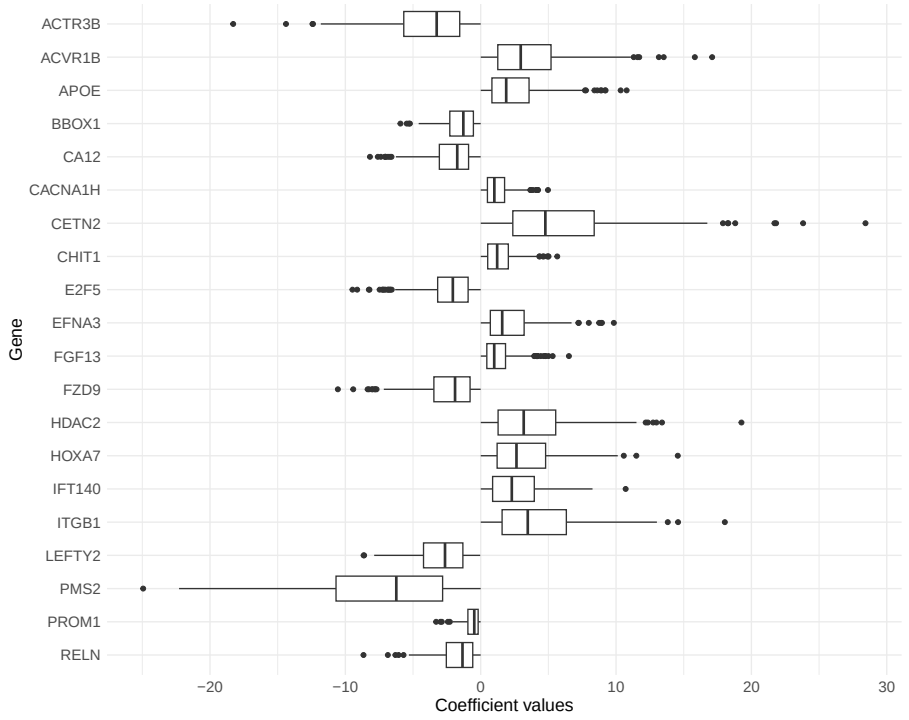


Figure 4.10: The coefficient values of the 20 most selected genes by the lasso models using the ROR score as response variable (bootstrap sampling technique).

Turning to the models fitted using the repeated bootstrap strategy and utilizing the proliferation score as a response variable, five genes (BMPR1B, CA12, CACNA1H, EFNA3, GATA3, HOXA9, LEFTY2) were selected more than 50% of the time, and 45 genes were selected at least 10% of the time. The top 20 selected genes included GATA3, LEFTY2, CACNA1H, HOXA9, CA12, BMPR1B, EFNA3, NSD1, HOXA7, CAMK2B, APOE, THY1, JAG1, CBLC, CHIT1, TAPBP, CALML5, RPS6KB1, PIM1, and EGLN2 (see Figure 4.11). The coefficient values of these genes reveal that GATA3 has the clearly largest absolute value (Figure 4.12).

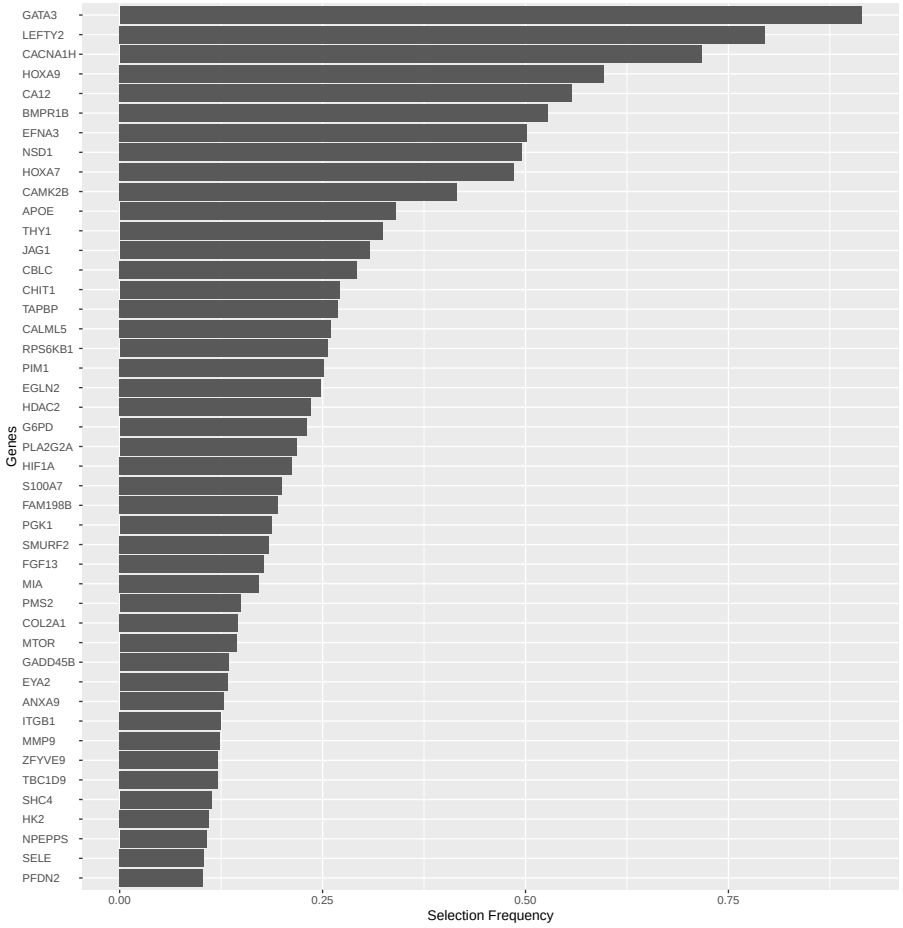


Figure 4.11: Genes selected in more then 10% of the lasso models achieved through repeated cross-validation using proliferation score as response variable

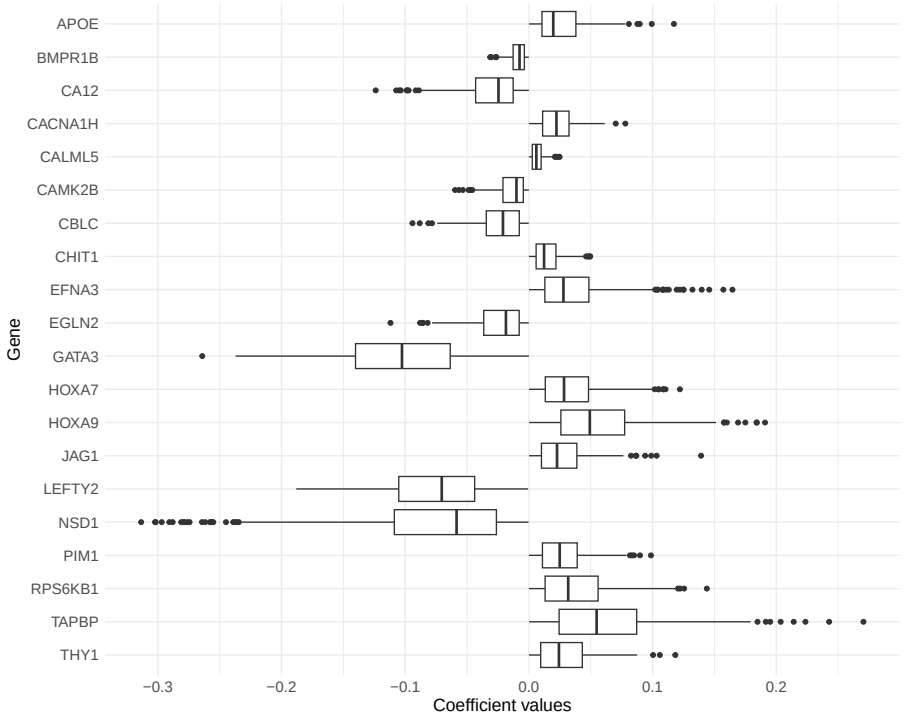


Figure 4.12: The coefficient values of the 20 most selected genes by the lasso models using the proliferation score as response variable (repeated cross-validation as sampling technique).

Finally, when using the ROR score as a response variable, no genes were selected in more than half of the models, and only 16 genes were selected more than 10% of the time (Figure 4.13). Upon examining the coefficient values, PMS2 demonstrated substantially larger absolute values compared to the other genes (Figure 4.14).

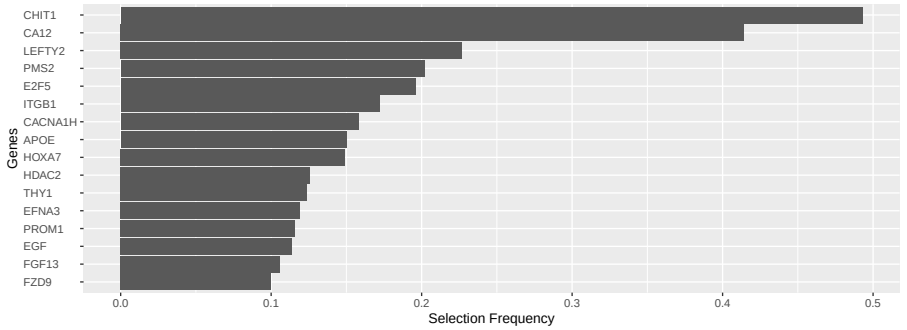


Figure 4.13: Genes selected in more then 10% of the lasso models achieved through repeated cross-validation using ROR score as response variable

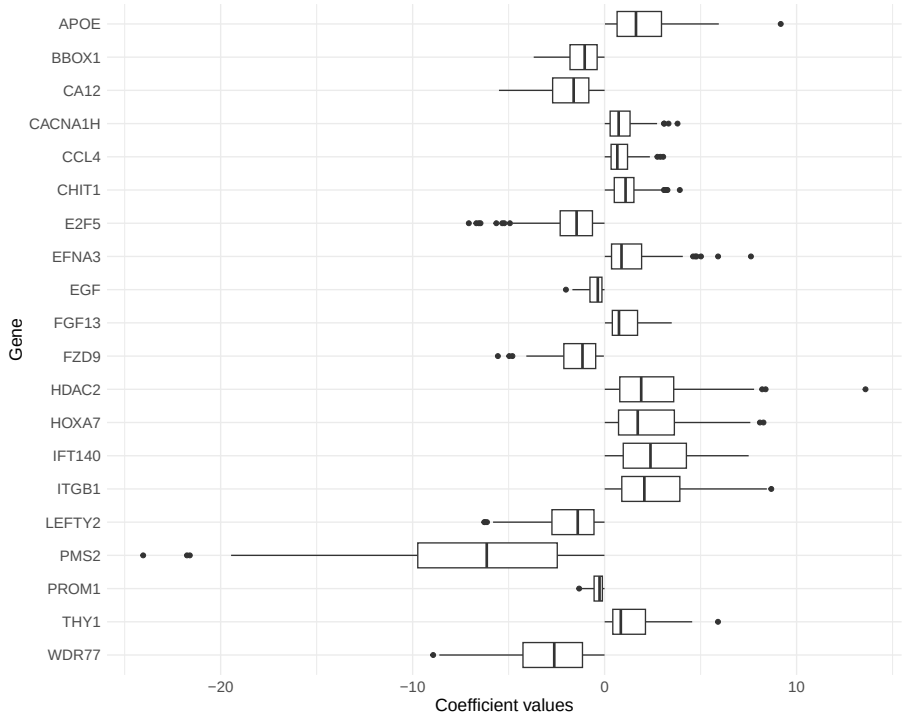


Figure 4.14: The coefficient values of the 20 most selected genes by the lasso models using the ROR score as response variable (repeated cross-validation as sampling technique).

Performance

When employing the lasso regression model class evaluated by the bootstrap strategy, the proliferation score analysis exhibited a slightly higher correlation between the predicted and observed proliferations than the corresponding measures for the ROR score analysis (Table 4.3). The predictions of the proliferation score had a mean correlation of 0.79 (SD = 0.090) with the observations, while the ROR score analysis obtained a mean correlation of 0.70 (SD = 0.099). The ROR score's MSE was notably larger, which is in agreement with their different scales.

Using the repeated cross-validation approach resulted in a decline in models performance, characterized by significantly lower correlation values and heightened MSE for both scores, with the ROR score being especially affected (Table 4.3). This pattern mirrors our observations in ridge regression and likely shares similar explanations.

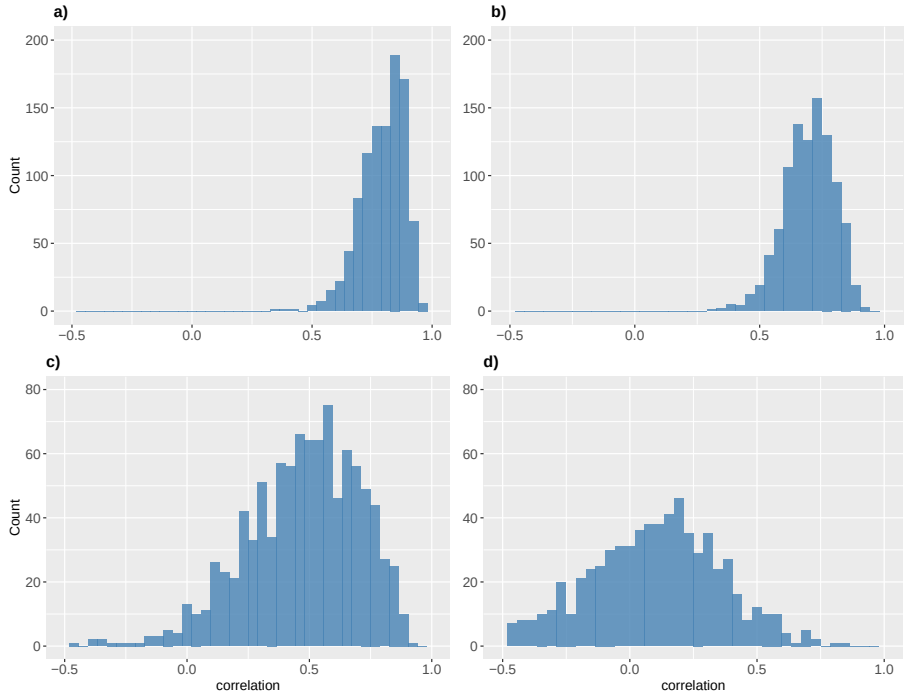


Figure 4.15: The lasso model using 771 genes as features. The histograms show the correlation between predicted response and the true values. In a) and b) bootstrapping is used to generate training samples. In a) the response variable is the proliferation score while in b) the ROR is the response variable. In c) and d) repeated cross-validation is used to generate multiple training and testing datasets. In c) the response variable is the proliferation score while in d) the ROR is the response variable.

Table 4.3: Lasso performance summary

Model	Correlations (SD)	MSE (SD)
Proliferation (bootstrap)	0.794 (0.090)	0.062 (0.024)
ROR (bootstrap)	0.697 (0.100)	203 (61)
Proliferation (repeated cross-val.)	0.474 (0.231)	0.138 (0.075)
ROR (repeated cross-val.)	0.081 (0.277)	393 (159)

4.3 Elastic net

Elastic net regression is a hybrid method combining ridge and lasso regression techniques. We have employed an α-value of 0.5 to assign equal weight to both ridge and lasso components (see 3.5 for a description of elastic net). We also here utilized the bootstrap and repeated cross-validation regimes. We analysed both the proliferation score and the ROR score. First, we present the results of feature selection and coefficient estimation for the predictors (the

gene expression values), followed by an evaluation of the elastic net model's performance on the clinical dataset.

Feature Selection

For the four different combinations of sampling approaches and response variables, with the exception of the repeated cross-validation with proliferation score as outcome, at least 10% of the samples selected all the genes. Applying a selection frequency cutoff of 30%, the models still identified a considerable number of genes (range 35 - 109, see table 4.4). Consequently, we focused on features selected more than 50% of the time. In the case of the bootstrap sampling with proliferation score as outcome, 16 genes were selected (Figure 4.16). For the bootstrap sampling and ROR score combination 26 genes were selected (Figure 4.17). We observed that the genes CA12, CACNA1H, CAMK2B, FGF13, HOXA7, and LEFTY2 were selected by both sampling methods.

In the case of repeated cross-validation, both the proliferation and ROR score analyses yielded eight genes (see histograms in Figure 4.18 and 4.19). Here, the common genes were CA12, CACNA1H, EFNA3, and LEFTY2.

When examining genes uniquely selected by the proliferation score across the sampling methods, the following genes were identified: BMPR1B, GATA3, and NSD1. Similarly, for the ROR score as response variable, CHIT1, E2F5, and FZD9 were selected in more then 50% of the models in both sampling methods.

Interestingly, the selection frequency appeared to be slightly higher for the ROR score, which is consistent with the fact that this score considers not only the proliferation of cancer cells but also other response variables (as described in chapter 2).

Table 4.4: Selection frequency for the elastic net model class using different cutoffs

Cutoff	Bootstrap		Repeated cross-validation	
	proliferation score	ROR score	proliferation score	ROR score
10%	771	771	634	771
30%	58	109	35	74
50%	16	26	8	8

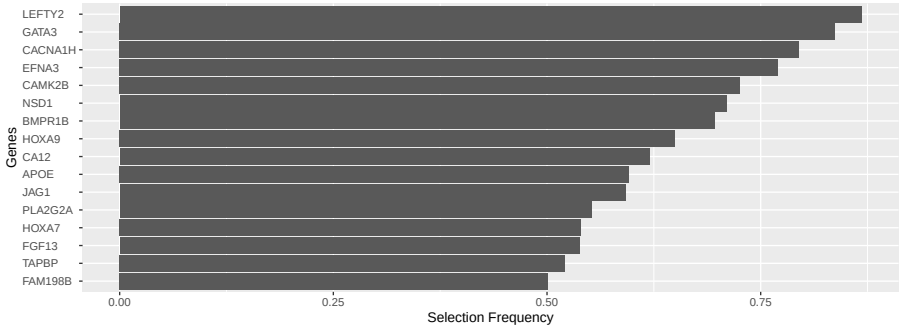


Figure 4.16: Genes selected in more than 50% of the elastic net models achieved through bootstrapping using proliferation score as response variable

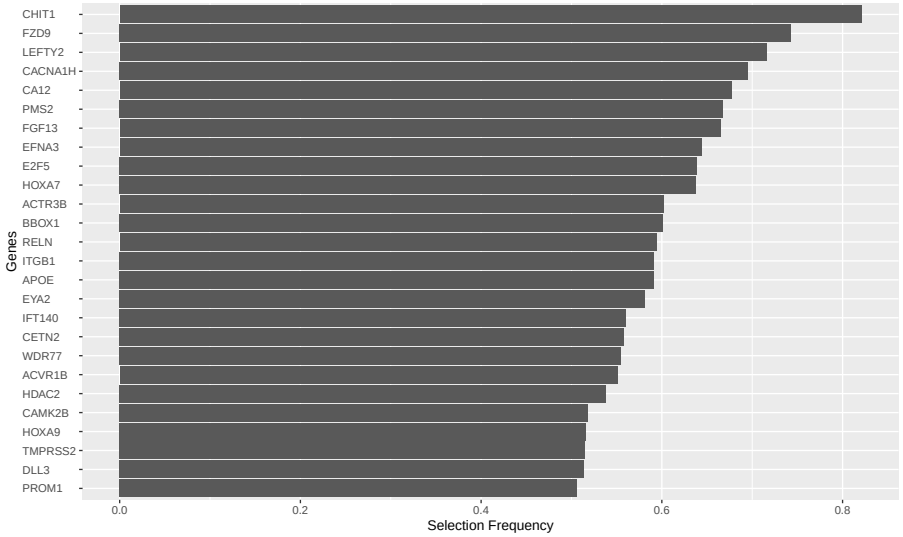


Figure 4.17: Genes selected in more than 50% of the elastic net models achieved through bootstrapping using ROR score as response variable

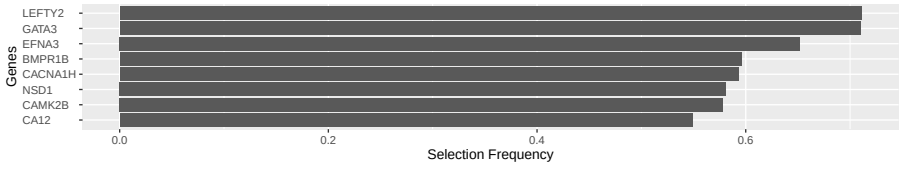


Figure 4.18: Genes selected in more than 50% of the elastic net models achieved through repeated cross-validation using proliferation score as response variable

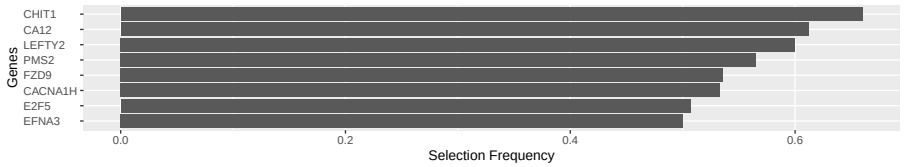


Figure 4.19: Genes selected in more than 50% of the elastic net models achieved through repeated cross-validation using ROR score as response variable

The coefficient values for the 20 gene expressions with highest selection frequency are shown in four successive boxplots (Figures 4.20, 4.21, 4.22 and 4.23)

We noted that GATA3, LEFTY2, and NSD1 exhibit markedly more negative median values for the models using proliferation score (in both sampling regimes) in comparison to other genes, implying a negative impact of these genes on cell proliferation. These genes all demonstrated a selection frequency above 50

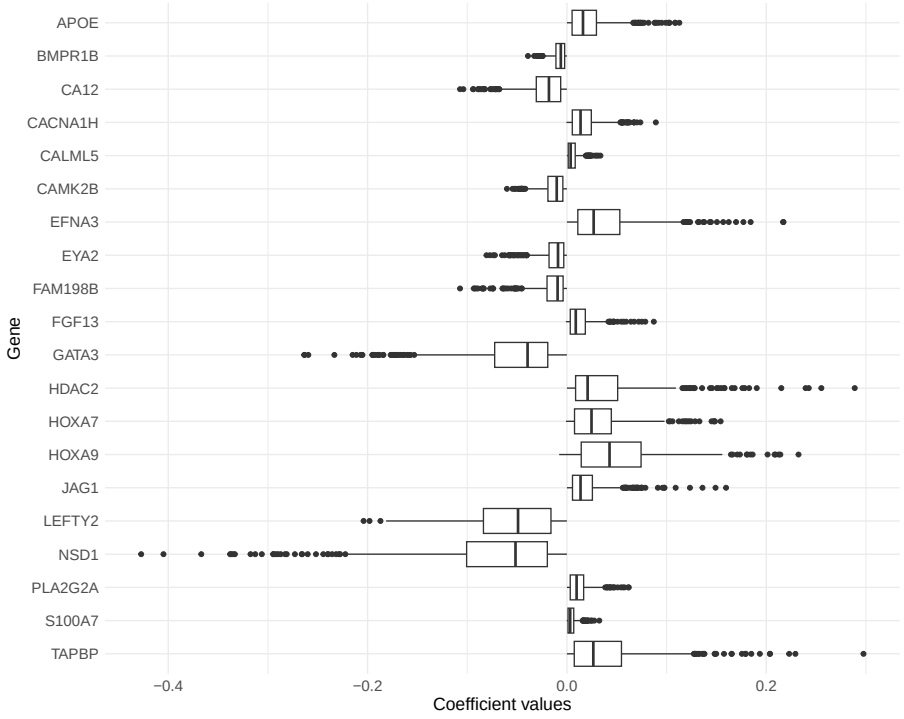


Figure 4.20: The coefficient values of the 20 most selected genes by the elastic net models using the proliferation score as response variable (bootstrap sampling technique).

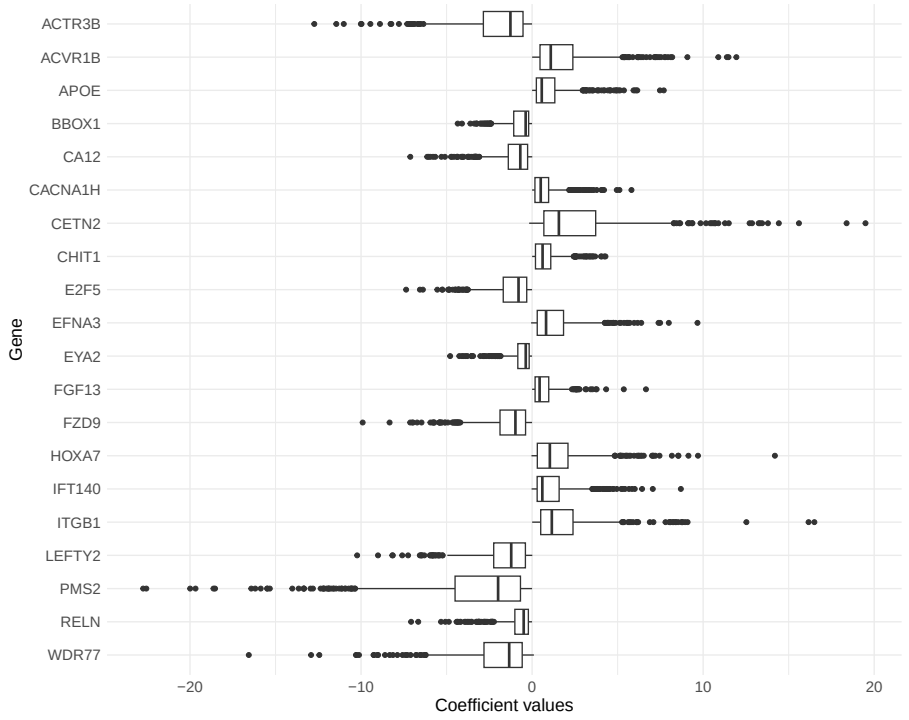


Figure 4.21: The coefficient values of the 20 most selected genes by the elastic net models using the ROR score as response variable (bootstrap).

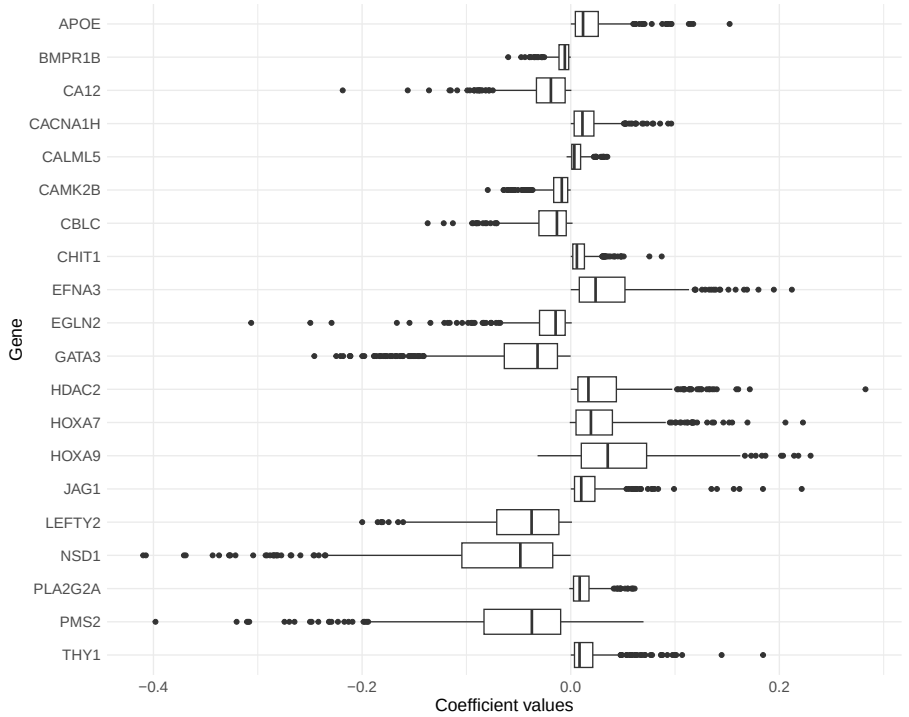


Figure 4.22: The coefficient values of the 20 most selected genes by the elastic net models using the proliferation score as response variable (repeated cross-validation).

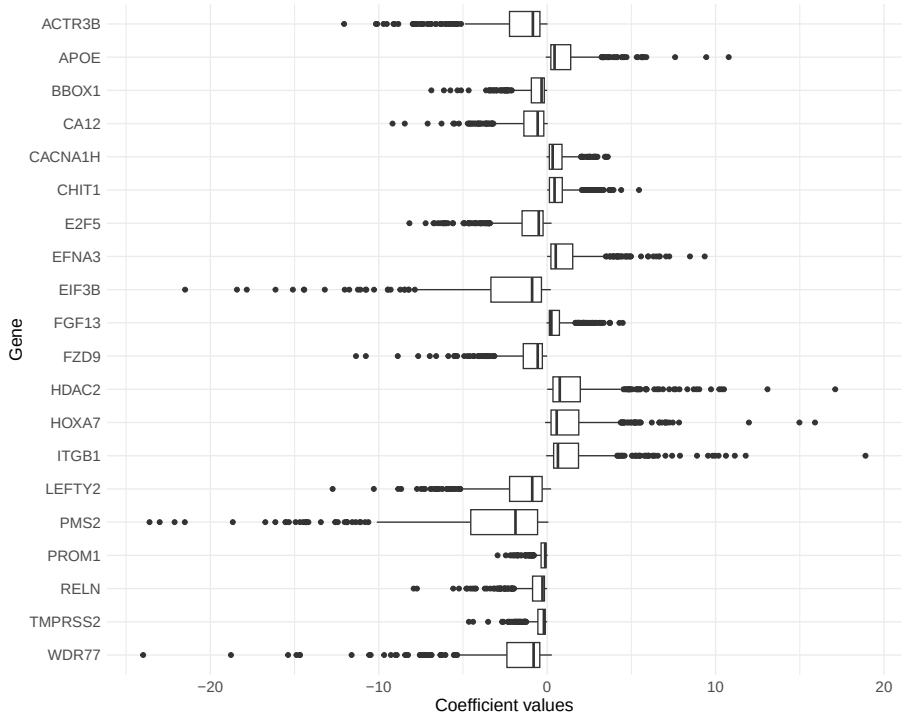


Figure 4.23: The coefficient values of the 20 most selected genes by the elastic net models using the ROR score as response variable (repeated cross-validation).

Performance

In our evaluation of the elastic net models employing the bootstrap strategy, both analyses with proliferation and ROR scores as responses exhibited comparable correlations between the predicted outcomes and observations in the test dataset (see Figure 4.24 and Table 4.5). The proliferation score had a mean correlation of 0.81 (SD = 0.083), while the ROR score was 0.76 (SD = 0.089). The ROR score's MSE was notably larger, aligning with their different scales (MSE: 164 for the ROR and 0.056 for proliferation).

Using the repeated cross-validation regimes, the models demonstrated reduced performance, with significantly lower correlation values for both scores and an MSE larger. This is similar to what we observed for ridge and lasso which is as expected.

Table 4.5: Elastic net performance summary

Model	Correlations (SD)	MSE (SD)
Proliferation (bootstrap)	0.81 (0.084)	0.06 (0.022)
ROR (bootstrap)	0.76 (0.089)	164 (52)
Proliferation (repeated cross-val.)	0.43 (0.264)	0.15 (0.082)
ROR (repeated cross-val.)	0.21 (0.311)	427 (186)

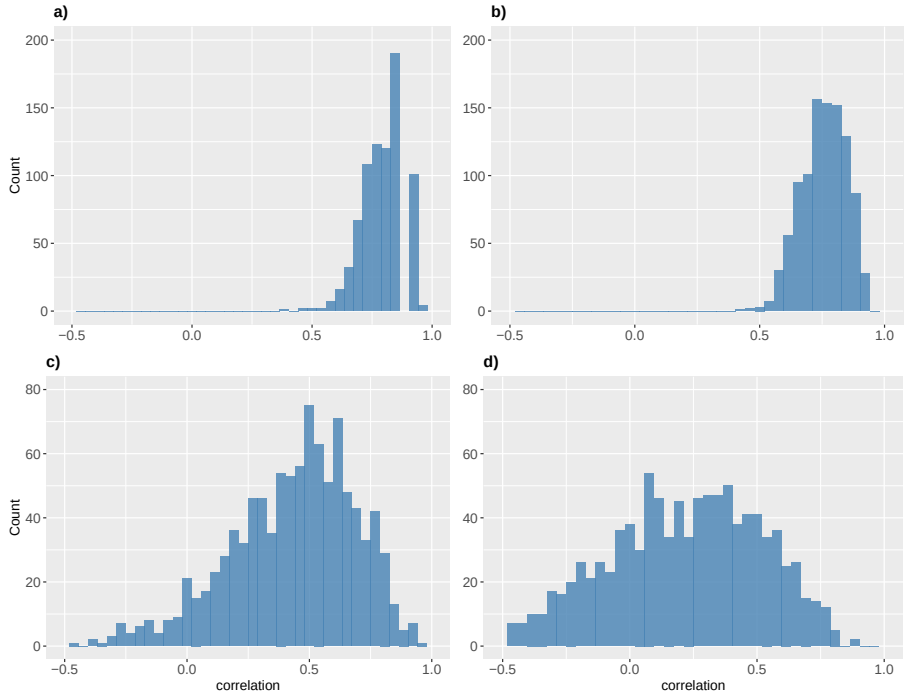


Figure 4.24: The elastic net model using 771 genes as features. The histograms show the correlation between predicted response and the true values. In a) and b) bootstrapping is used to generate training samples. In a) the response variable is the proliferation score while in b) the ROR is the response variable. In c) and d) repeated cross-validation is used to generate multiple training and testing datasets. In c) the response variable is the proliferation score while in d) the ROR is the response variable.

4.4 Non-linear model: boosting with stumps

In addition to the linear models examined above, we applied boosting using stumps as base learners in order to reveal potential non-linear relationships between features and responses in the dataset. We here also utilized the bootstrap and repeated cross-validation strategies (Section 3.9), along with the proliferation score and ROR score as two different response variables. First, we present the results of gene feature selections, followed by the performance evaluation of boosting on the dataset from the clinical trial.

Feature selection

The feature selection frequency of the four distinct combinations of sampling approaches and response variables are shown in Table 4.6. Interestingly, for the 50% and 30% cutoff, the proliferation score selected more genes than the ROR score. This is surprising as the ROR score is based on more outcome variables, and it was therefore expected that more genes would have a relevant

influence on the ROR score. We also observed a lower selection frequency for the repeated cross-validation, as was the case for elastic net.

Table 4.6: Selection frequency for the boosting with stumps as base learner

Cutoff	Bootstrap		Repeated cross-validation	
	proliferation score	ROR score	proliferation score	ROR score
10%	53	44	34	16
30%	10	6	15	6
50%	0	1	2	2

For the gene features selected 50% or more times, HDAC2 was selected for the ROR score in both sampling approaches, while the repeated cross-validation scheme additionally selected HOXA7. CALML5 and CHIT1 were selected in the repeated cross-validation strategy with the proliferation score as response variable (no genes were selected more than 50% of the times by bootstrapping for the proliferation score).

Turning to the genes selected at least 10% of the times, we observed the following. Examining the proliferation score in isolation, all genes selected by the repeated cross-validation approach were also selected by the bootstrap approach, besides three genes. The commonly seleted genes were: BMPR1B, BTG2, CA12, CACNA1H, CALML5, CAMK2B, CD84, CDCA7L, CHIT1, CKB, DKK1, EFNA3, EGLN2, EPAS1, ERCC1, G6PD, HDAC2, HIF1A, HK2, HOXA7, IDO1, KIT, LEFTY2, MAP2K4, NEIL1, NSD1, OLFML2B, PMS2, SFN, SFRP4, and ZFYVE9. For the ROR score, all the genes selected using repeated cross-validation were also selected when using bootstrapping. These genes were: ADCY9, APOD, APOE, CA12, CACNA1H, CHIT1, CYBB, E2F5, ELF3, HDAC2, HOXA7, LEFTY2, PROM1, RELN, RPS6KB2, and SFRP4.

The selection frequency for the genes that were selected at least 10% of the time is presented in the four histograms in the Figures 4.25, 4.26, 4.27 and 4.28.

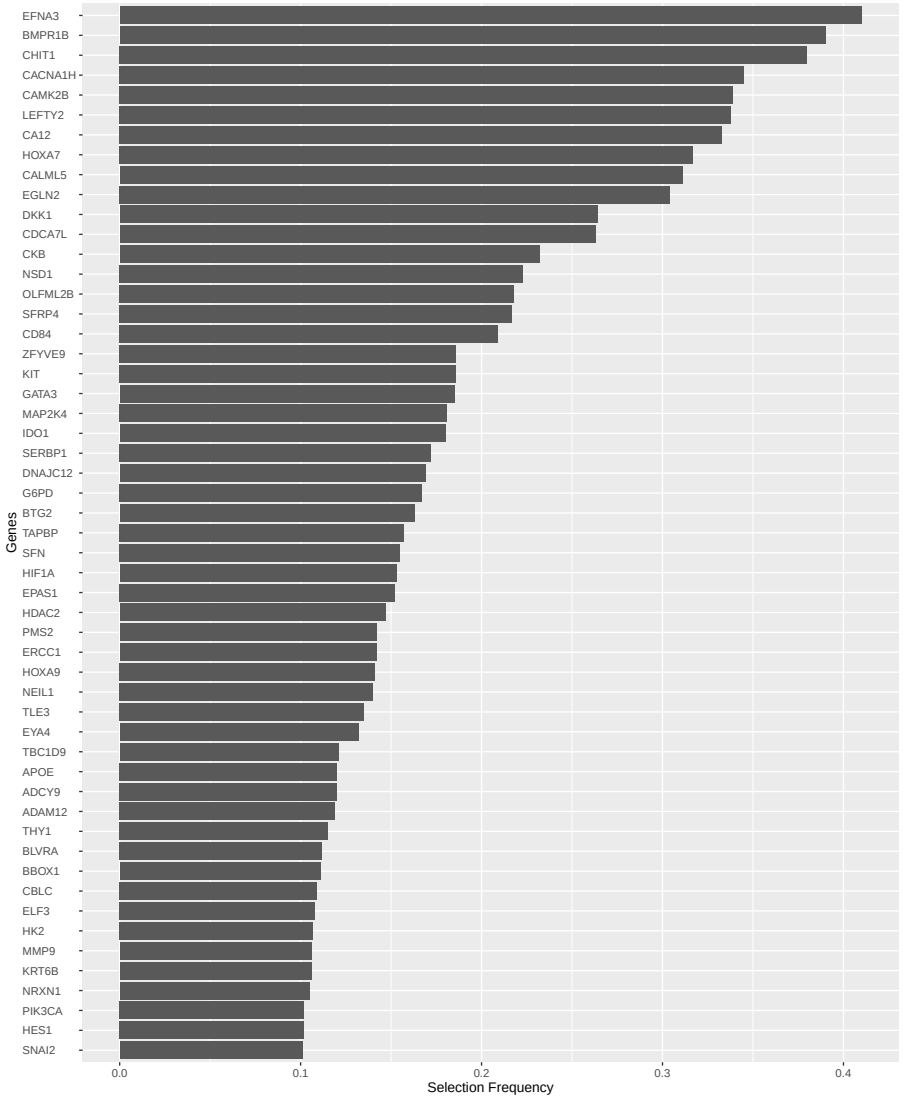

EFNA3
BMPR1B
CHIT1
CACNA1H
CAMK2B
LEFTY2
CA12
HOXA7
CALML5
EGLN2
DKK1
CDCA7L
CKB
NSD1
OLFML2B
SFRP4
CD84
ZFYVE9
KIT
GATA3
MAP2K4
IDO1
SERBP1
DNAJC12
G6PD
BTG2
TAPBP
SFN
HIF1A
EPAS1
HDAC2
PMS2
ERCC1
HOXA9
NEIL1
TLE3
EYA4
TBC1D9
APOE
ADCY9
ADAM12
THY1
BLVRA
BBOX1
CBLC
ELF3
HK2
MMP9
KRT6B
NRXN1
PIK3CA
HES1
SNAI2
Genes
0.0
0.1
0.2
0.3
0.4
Selection Frequency

Figure 4.25:

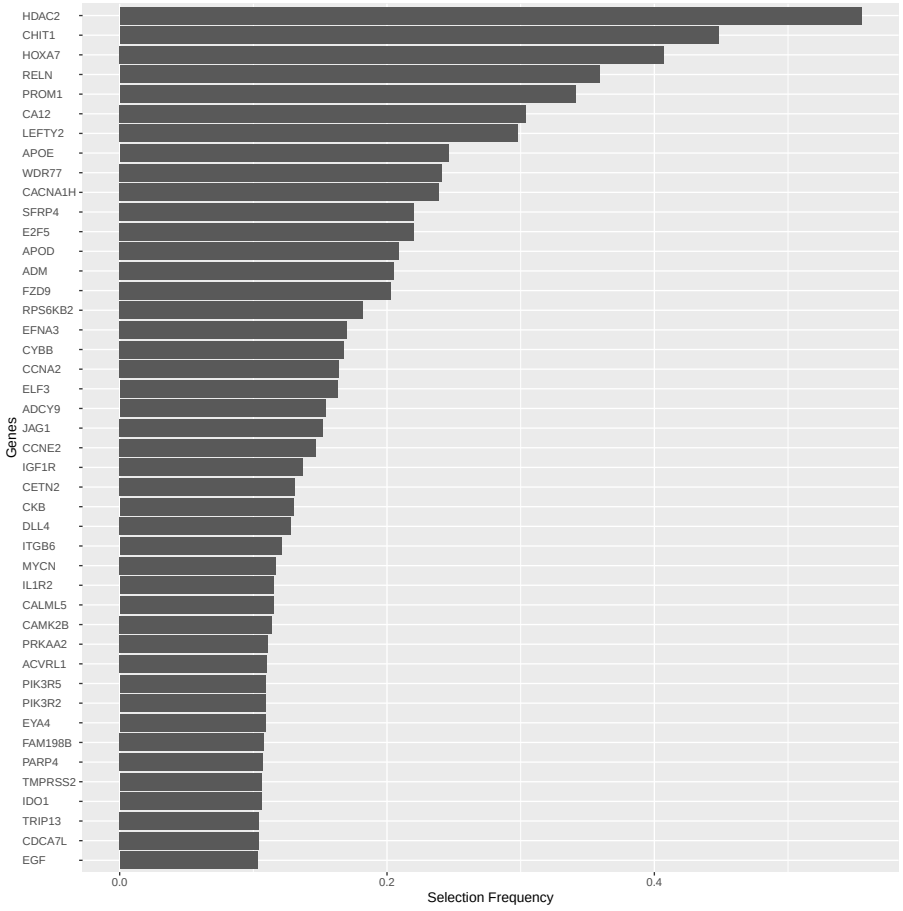

HDAC2
CHIT1
HOXA7
RELN
PROM1
CA12
LEFTY2
APOE
WDR77
CACNA1H
SFRP4
E2F5
APOD
ADM
FZD9
RPS6KB2
EFNA3
CYBB
CCNA2
ELF3
ADCY9
JAG1
CCNE2
IGF1R
CETN2
CKB
DLL4
ITGB6
MYCN
IL1R2
CALML5
CAMK2B
PRKAA2
ACVRL1
PIK3R5
PIK3R2
EYA4
FAM198B
PARP4
TMPRSS2
IDO1
TRIP13
CDCA7L
EGF
Genes
0.0
0.2
0.4
Selection Frequency

Figure 4.26:

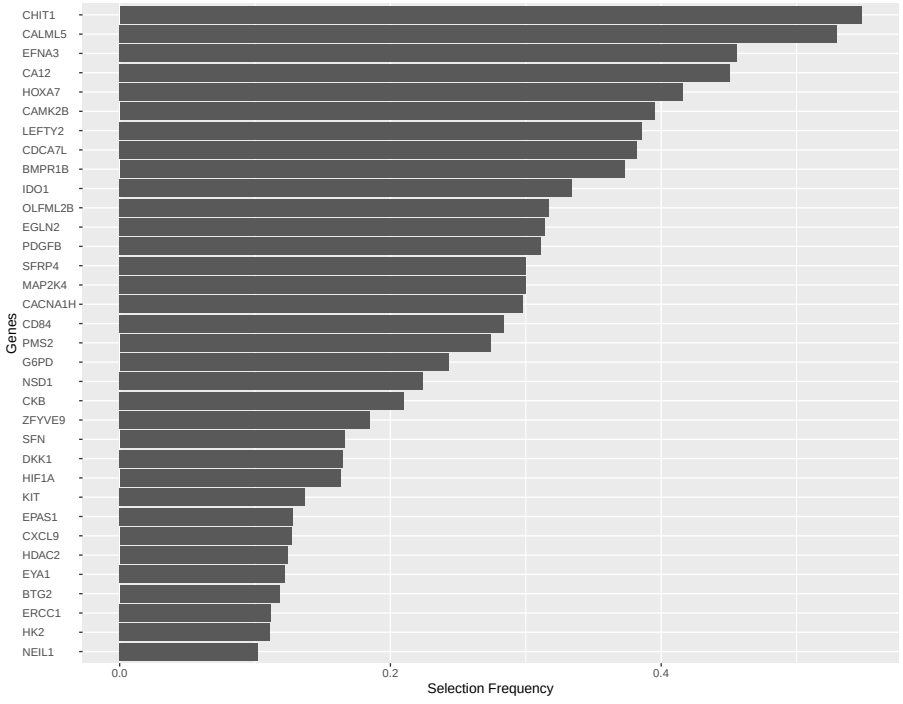


Figure 4.27:

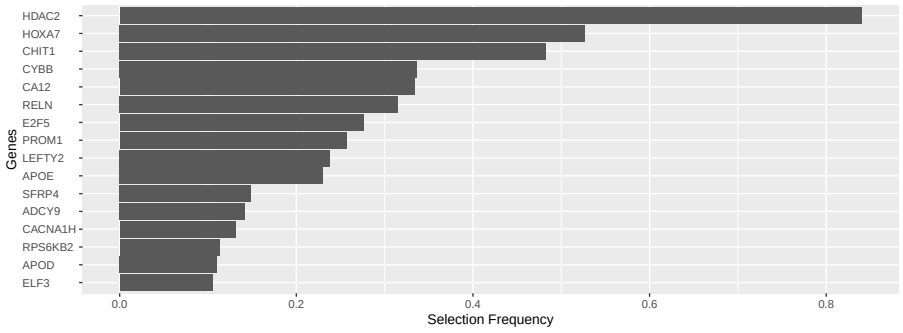


Figure 4.28:

Performance

Based on calculated correlations between predicted and observed responses,this method performed at the same level as the elastic net (see Figure 4.29 and Table 4.7). Consequently, we were unable to observe non-linear relationship of some genes with the responses in the data that could improve the the predictions.

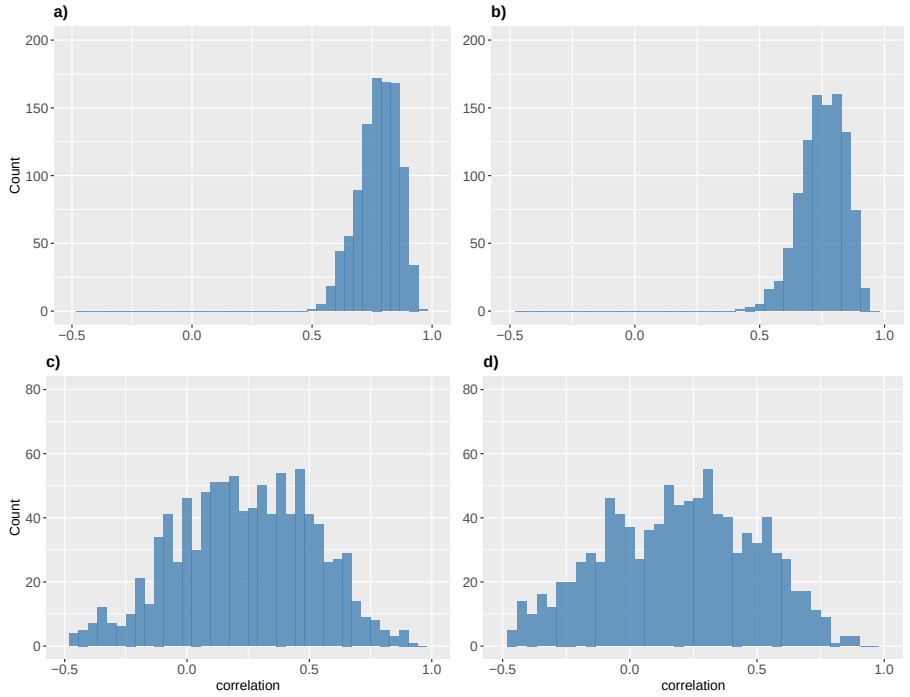


Figure 4.29: The boosting model using 771 genes as features. The histograms show the correlation between predicted response and the true values. In a) and b) bootstrapping is used to generate training samples. In a) the response variable is the proliferation score while in b) the ROR is the response variable. In c) and d) repeated cross-validation is used to generate multiple training and testing datasets. In c) the response variable is the proliferation score while in d) the ROR is the response variable.

Table 4.7: Summary results of the performance of boosting with stumps

Model	Correlations (SD)	MSE (SD)
Proliferation (bootstrap)	0.78 (0.083)	0.07 (0.021)
ROR (bootstrap)	0.75 (0.088)	165 (51)
Proliferation (repeated cross-val.)	0.24 (0.280)	0.17 (0.076)
ROR (repeated cross-val.)	0.17 (0.315)	394 (171)

sd_

4.5 Comparison of the standard machine learning models ridge regression, elastic net, lasso and boosting

To assess the differences between the machine learning models ridge regression, elastic net, lasso, and boosting, we compared their selected features and performance on the dataset derived from the clinical trial.

Feature selection

We compared the feature selection of lasso, elastic net and boosting, as they are specifically designed for feature selection. Our analysis concentrated on the 20 most frequently selected genes in each model. In a set of Venn diagrams of results from the various combination of sampling approach and the proliferation- or ROR score as outcome, we observed that lasso and elastic net had considerably more overlap compared to boosting (Figure 4.30). These two models had 18 genes in common, while boosting had 9 exclusively selected genes. Out of the total 31 genes, 10 genes were selected by all models. See table 4.8 for overview of all selected genes for the combination of the ROR score and repeated cross-validation.

It is not surprising that lasso and elastic net behave so similarly, since they are of the same family of models. Boosting with stumps is more adapted to capture nonlinear relationships. Consequently, the non-overlapping genes likely convey non-linearity with respect to the response variables.

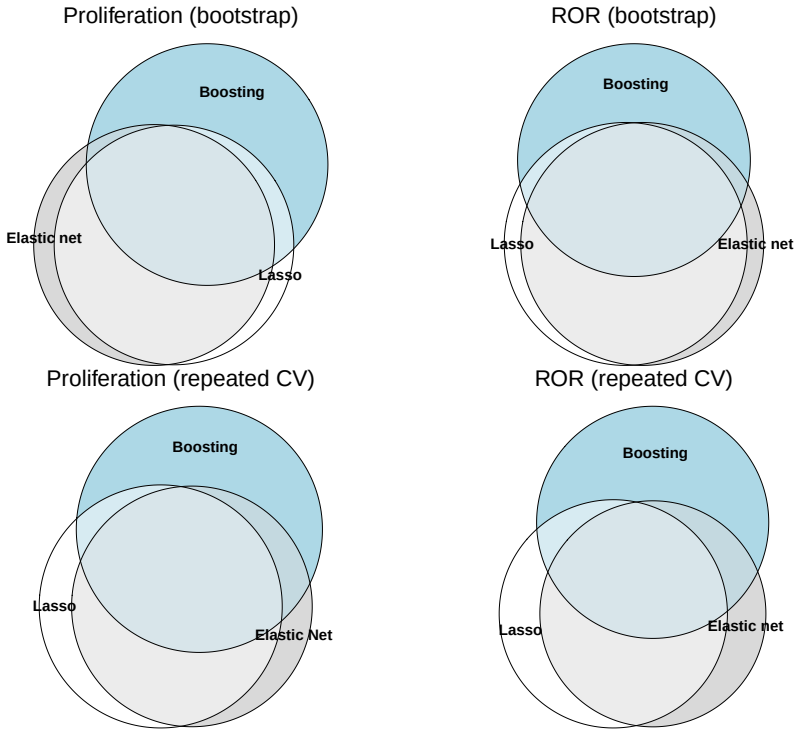


Figure 4.30:

Table 4.8: Genes selected in lasso, elastic net and boosting

Gene	Lasso	Elastic net	Boosting
BMPR1B	X	X	X
CA12	X	X	X
CACNA1H	X	X	X
CALML5	X	X	X
CAMK2B	X	X	X
EFNA3	X	X	X
GATA3	X	X	X
HOXA7	X	X	X
LEFTY2	X	X	X
NSD1	X	X	X
APOE	X	X	
CHIT1	X		X
FGF13	X	X	
HDAC2	X	X	
HOXA9	X	X	
JAG1	X	X	
PLA2G2A	X	X	
S100A7	X	X	
TAPBP	X	X	
CBLC	X		
CD84			X
CDCA7L			X
CKB			X
DKK1			X
EGLN2			X
EYA2		X	
FAM198B		X	
KIT			X
OLFML2B			X
SFRP4			X
ZFYVE9			X

Performance

The models exhibited differences in predictive power with respect to the correlation between predicted and observed outcomes. In Table 4.9 the performance values for all the models using the ROR score in combination with the repeated cross-validation approach is shown for comparison purposes. This combination is also used throughout the rest of the thesis. Ridge outperformed lasso, while elastic net results fell between the two. Boosting, showed similar correlation as elastic net. However, substantial variance was observed. In fact, when comparing the MSE, we noticed minimal differences between the three models. It is not feasible to conclude that any of the models is significantly better than any of the others, however, ridge have the highest correlation and the lowest MSE. This is in accordance with ridge been the preferred model if prediction is the only output of interests.

Table 4.9: Predictive performance of standard models using the ROR score and the repeated cross validation approach.

Model	Correlations (SD)	MSE (SD)
Ridge	0.33 (0.262)	357 (150)
Lasso	0.08 (0.277)	393 (159)
Elastic net	0.21 (0.311)	427 (186)
Boosting	0.17 (0.315)	394 (171)

CHAPTER 5

Mechanistic model combined with machine learning models

In the research group where this thesis is conducted, a mechanistic model of cancer cell proliferation is being developed. The model encompasses the intracellular signaling pathway between the estrogen receptor and a transcription factor, RB1, which is involved in regulating the transition from the growth phase 1 to the synthesis phase of the cell cycle (see Figure 2.2 in Chapter 2). This pathway is where CDK4/6 potentially have its target and the model includes the response to estrogen based hormone therapy and the CDK4/6 inhibitor. We have compared the performance of this model to machine learning models and have also attempted to integrate the mechanistic model with machine learning models for improved results.

5.1 Mechanistic pathway model of response to hormone therapy and CDK4/6 inhibitors

We here give a brief description of the mechanistic model, the parameter estimation and how it is used.

Model description

We employed a mechanistic mathematical model, based on the model developed in He et al. 2020. This model describes protein-protein and drug-protein interactions of the key protein signaling pathways affected by the treatment with hormone therapy and CDK4/6 inhibitors (Figure 5.1A) and is based on prior knowledge. We adapted the model from He et al. 2020 to also consider mRNAs, which can be used to individualize the model based on gene expression data. The model consists of a set of coupled ordinary differential equations (ODE), which describe how the concentrations of the proteins and mRNAs change over time after drug treatment. The ODEs can generally be denoted as

$$\dot{\boldsymbol{x}}(t, \theta, u) = f(\boldsymbol{x}(t, \boldsymbol{\theta}, \boldsymbol{u}), \boldsymbol{\theta}, \boldsymbol{u}), \quad \boldsymbol{x}(t_0, \boldsymbol{\theta}, \boldsymbol{u}) = \boldsymbol{x}_0(\boldsymbol{\theta}, \boldsymbol{u}).$$

$\boldsymbol{x} \in \mathbb{R}^{n_x}$ denotes the state vector, that describes the concentration of the modeled species. In this case, these are the mRNAs, proteins and protein complexes. $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$ are parameters of the model, such as binding and degradation rates. In total, the model consists of $n_x = 23$ states and $n_\theta = 64$

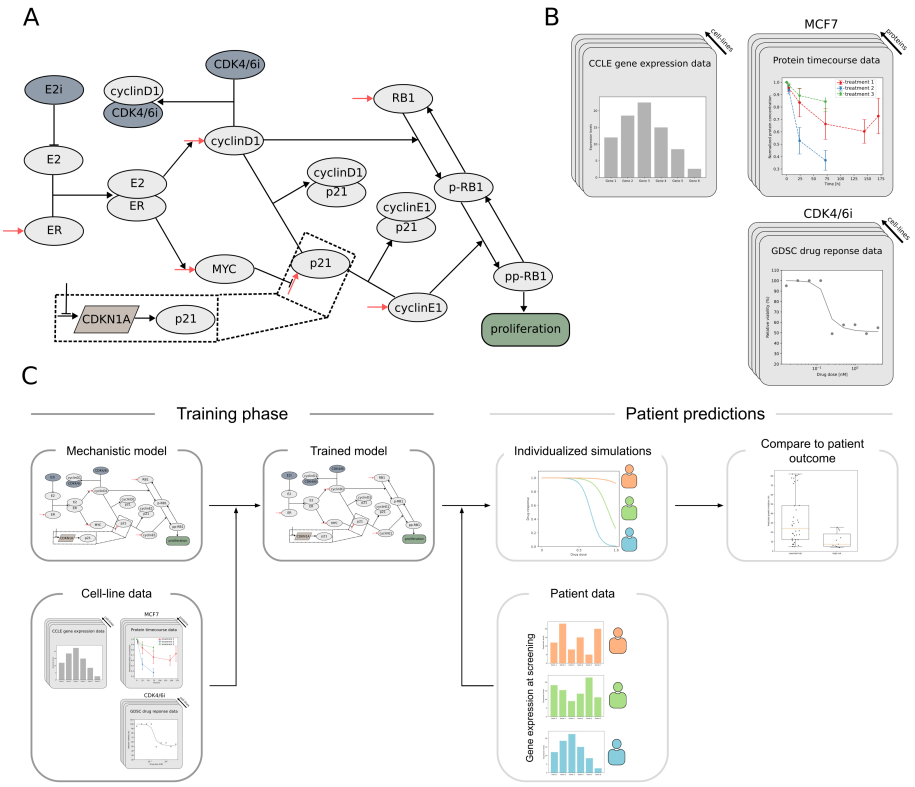


Figure 5.1: (A) Schematic of the mechanistic model. E2i and CDK4/6i denote the hormone therapy and CDK4/6 inhibitor, that bind and block the activity of E2 and cyclinD1 respectively. The activated estrogen receptor (ER bound to E2) increases the production of cyclinD1 and MYC. CyclinD1 and cyclinE1 phosphorylate RB1, resulting in hyperphosphorylated RB1 (pp-RB1), which drives cell-cycle progression. pp-Rb1 is linked to cell proliferation, which is used as readout for predictions. p21 acts as a natural inhibitor of cyclinD1 and cyclinE1 The dashed box in the lower part of the figure on p21 shows that protein production is modeled in two steps by mRNA transcription and protein translation. CDKN1A is the gene encoding the protein p21. (B) Data used for parameter estimation. CCLE gene expression data is used to individualize the model to different cell-lines. Protein timecourses for different drug treatments measured in MCF7 cells and CDK4/6i drug response data from GDSC measured in multiple cell-lines is used for estimating the unknown parameters of the mechanistic model. (C) Overview of the general estimation and prediction workflow. Parameters are estimated first in the training phase using the previously described cell-line data. The finalized model is subsequently individualized to patients using gene expression data from the six genes in the model. Simulated response are then compared to the actual outcome after treatment.

parameters. $\boldsymbol{u} \in \mathbb{R}^{n_u}$ is an input vector, that denotes different experimental conditions, which describe here the drug treatment. $\boldsymbol{x}_0 \in \mathbb{R}^{n_x}$ are the parameter- and condition-dependent states at initial time t_0. In particular, gene expression data is used here as initial conditions to individualize the model. $f \in \mathbb{R}^{n_x}$ determines the interactions of the states and their temporal changes. Complex formations and drug inhibitions are modeled using mass action kinetics (Ingalls 2013) E.g. the differential equation describing the concentration of the E2:ER complex is given by

$$\frac{d[\text{E2:ER}]}{dt} = k_{\text{binding}}[\text{E2}][\text{ER}] - k_{\text{unbinding}}[\text{E2:ER}] - k_{\text{degradation}}[\text{E2:ER}], \quad (5.1)$$

with (un-)binding rates k_{binding} and $k_{\text{unbinding}}$ and degradation rate $k_{\text{degradation}}$. Here, $[x]$ denotes the concentration of the molecular species x. Phosphorylations and transcriptional regulation are modeled using Hill functions (Ingalls 2013) E.g. the phosphorylation of RB1 by cyclinD1 is described by

$$k_{\text{RB1cyclinD1}}[\text{cyclinD1}]\frac{[\text{RB1}]^{p_2}}{p_1^{p_2} + [\text{RB1}]^{p_2}} \quad (5.2)$$

with the phosphorylation rate $k_{\text{RB1cyclinD1}}$ and parameters p_1, p_2 defining the shape of the Hill function. Cancer cell proliferation is modeled assuming logistic growth and is dependent on the levels of pp-RB1 via a Hill function. The ODE is given by

$$\begin{aligned}\frac{d[\text{proliferation}]}{dt} = k_1 &\left(1 + k_{\text{proliferationppRB1}}\frac{[\text{pp-RB1}]^{p_2}}{p_1^{p_2} + [\text{pp-RB1}]^{p_2}}\right) \\ &[\text{proliferation}]\left(1 - \frac{[\text{proliferation}]}{k_{\text{carrying}}}\right),\end{aligned} \quad (5.3)$$

with carrying capacity k_{carrying}, Hill parameters p_1, p_2 and basal and pp-RB1 dependent proliferation rates k_1, $k_{\text{proliferationppRB1}}$.

Parameter estimation

The model consists of unknown parameters $\boldsymbol{\theta}$, which have to be estimated from data. For this, publicly available data from breast cancer cell-lines were used. To individualize the model to cell-lines, gene expression data from the Cancer Cell Line Encyclopedia (CCLE) (Barretina et al. 2012) was used. These are measurements in untreated cell-lines. Parameteres were then estimated based on two datasets, resulting in 213 datapoints used for parameter estimation (Figure 5.1B):

- Protein timecourse measurements taken from He et al. 2020, were different proteins in the model were measured add multiple timepoints after drug treatment in the MCF7 breast cancer cell-line.
- Drug response viability measurements from the Genomics of Drug Sensitivity in Cancer (GDSC) (Yang et al. 2012), were cell viability was measured after treatment, relative to the untreated control in 12 different breast cancer cell-lines

Parameters were estimated by minimizing the negative log-likelihood function, assuming additive, normally distributed measurement noise, i.e. the optimal parameters $\hat{\boldsymbol{\theta}}$ are given by

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \frac{1}{2} \sum_{i=1}^{n} \log(2\pi\sigma^2) + \frac{(\bar{y_i} - y_i(\boldsymbol{\theta}))^2}{\sigma^2} \tag{5.4}$$

with number of measurements n, standard deviation of the measurement noise σ, measurements $\bar{y}$ and model output $y(\boldsymbol{\theta})$. This was done using multi-start gradient-based local optimization, were several local optimizations were initiated from different randomly sampled starting points to find a global optimum (Raue et al. 2013).

Model readout for patient predictions

The parameters of the model are estimated using the cell-line data described above. Subsequently, the model is used to make predictions in patients by using gene expression data as initial states (Figure 5.1C). As a readout of the model, that is used to predict patient response to the here considered treatments, we used the inhibition of proliferation induced by the drugs. To this end, the model was simulated without and with treatment until $T = 72$h, i.e. we calculated $\boldsymbol{x}(T, \boldsymbol{\theta}, 0)$ and $x(T, \boldsymbol{\theta}, \boldsymbol{u})$ for each patient by using the gene expression data as $\boldsymbol{x}_0$ and then considered the ratio of the proliferation state

$$y^{mm} = \frac{[\text{proliferation}](T, \hat{\boldsymbol{\theta}}, \boldsymbol{u})}{[\text{proliferation}](T, \hat{\boldsymbol{\theta}}, 0)},$$

y^{mm} becomes the response variable of the mechanistic model. This gives a score between $0 - 1$, where 1 indicates no response and 0 is the best possible response to the treatment. The final timepoint 72h and the drug doses $\boldsymbol{u}$ were chosen to mimic the GDSC drug response experiments.

5.2 Comparison of the mechanistic model with machine learning models

The mechanistic model utilizes mRNA expression data from six genes within the dataset and the prediction of the proliferation score exhibits a correlation of 0.38 with the score in the dataset. We first employed a linear regression model on the same six genes to see how a statistical model using the same amount of feature information compared to the mechanistic model. The bootstrap approach gave a mean correlation of 0.35 (SD=0.063), while the repeated cross-validation approach produces a mean correlation of 0.09 (SD=0.31). As presented in Chapter 4, using the full gene set results in correlations of approximately 0.8 and 0.5 for the different models using the bootstrap and repeated cross-validation approaches, respectively.

However, these correlation values are not directly comparable. Since the mechanistic model is not trained on the dataset, it can be considered as an external dataset for this model. Therefore, it might be most appropriate to compare the correlation values with those obtained through repeated cross-validation. Nonetheless, for the statistical models, these values are based on only

10 patients, whereas the mechanistic model is based on 49 patients. Therefore, to provide a more fair comparison between the models, we randomly sampled 10 patients 1000 times from the predictions of the mechanistic model and the corresponding values in the dataset. We then calculated the mean correlation of these samples, which yielded a value of 0.36 (SD=0.23). To sum up, it appears the mechanistic model performs substantially better than a statistical model when considering the amount of information it uses, but it is possible for a machine learning model to do equally good or better if enough feature data is provided.

5.3 Integrating the mechanistic model with a machine learning model

Next, we wanted to see if it was possible to a make a combined mechanistic and machine learning model. We explored two main approaches to integrate machine learning and mechanistic models. In the first approach, as the mechanistic model in addition to predicting outcome also estimates the concentrations of protein complexes in the signaling pathway regulating proliferation, we utilized these concentrations as features in a regression model. This resulted in a correlation of the proliferation score between predictions and observed responses in the dataset of 0.41 (SD=0.064) in the bootstrap approach and 0.28 (SD=0.277) for the repeated cross-validation approach. Although these results do not outperform the mechanistic model alone, they are noteworthy as the prediction of protein concentrations is an intermediate step in the mechanistic model. This indicates that these concentrations possess predictive power and that the mechanistic model employs them in a reasonable manner.

In the second approach, our goal was to utilize a machine learning model to capture information from the data that the mechanistic model did not use, and then combine the predictions from both models. To achieve this, we used the difference between the outcomes of the mechanistic model and the observed values in the dataset as the response variable for training the machine learning model. We used all the genes as covariates trying to explain the part of the outcome that the mechanistic model did not take care of. We evaluated two strategies based on different ways of computing the difference, denoted as d_i.

First, we considered the residuals, by simply subtracting the predicted values from the mechanistic model $\hat{y}_i^{mm}$ from the observed response variables y_i in the dataset. Second, we considered the fraction between the two, by dividing y_i by $\hat{y}_i^{mm}$, resulting in the two alternatives

$$d_i = y_i - \hat{y}_i^{mm}$$
$$d_i = \frac{y_i}{\hat{y}_i^{mm}}$$

Before computing d_i, we re-scaled both $\hat{y}_i^{mm}$ and y_i individually to have a min value of 0 and a max value of 1, by subtracting the lowest value and subsequently dividing by the max value. As we are interested in both feature selection and prediction we applied the elastic net regression where the parameters are found by optimisation of the loss function

$$\hat{\boldsymbol{\beta}} = \arg\min_{\beta} \left\{ \|\boldsymbol{d} - X\boldsymbol{\beta}\|_2^2 + \lambda(\alpha\|\boldsymbol{\beta}\|_2^2 + (1-\alpha)\|\boldsymbol{\beta}\|_1) \right\}, \tag{5.5}$$

where $\boldsymbol{d}$ is the vector of d_i's. For the final results we either added the predictions of the mechanistic model and the machine learning model $\hat{d}_i$ or multiplied them depending of whether the d_i was the residual or the fraction,

$$\begin{aligned} \hat{y}_i &= \hat{d}_i + \hat{y}i^{mm} \\ \hat{y}_i &= \hat{d}_i \cdot \hat{y}_i^{mm}. \end{aligned} \tag{5.6}$$

We found that this approach improved the predictive power of the mechanistic model to a level comparable to using the machine learning models alone. For the bootstrap approach, the additive method gave a correlation of 0.78 (SD=0.085), while the multiplicative method gave 0.73 (SD=0.089). For the repeated cross-validation approach, the additive method resulted in a correlation of 0.49 (SD=0.220), while the multiplicative method produced 0.40 (SD=0.230). Interestingly, upon examining the selection frequency of the genes used to initiate the mechanistic model, we observed that with the standard elastic net, they exhibited a selection frequency ranging from 10% to 20%. However, in the combined mechanistic and machine learning scenario, their selection frequencies dropped to below 0.3%. We notice that genes that were highly selected for the machine learning model alone still have high selection frequency in the combined model, suggesting no changes in the general selection frequency (Figure 5.2).

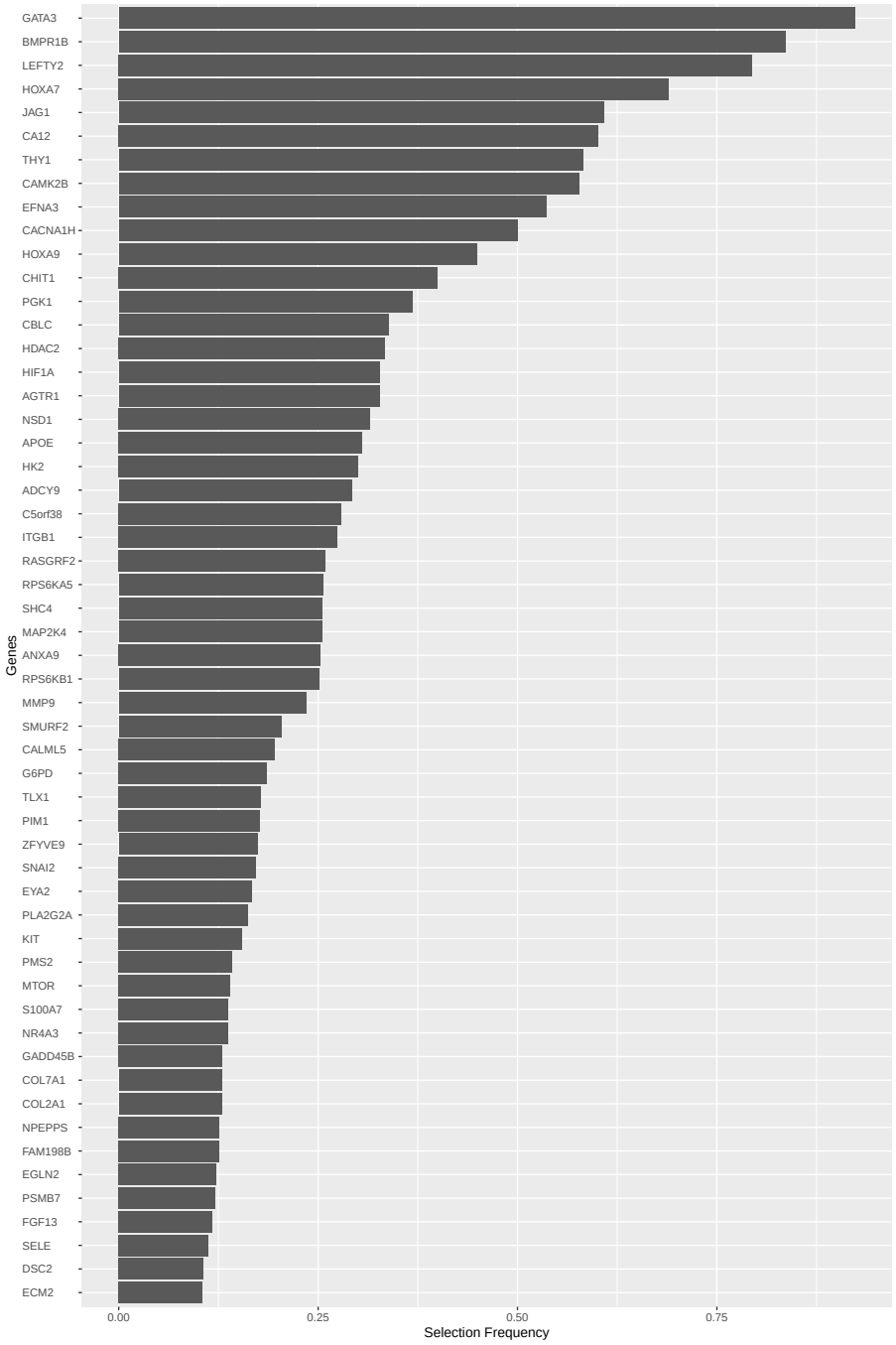


Figure 5.2: Genes selected in more than 10% of the models trained on the additive residual between the mechanistic model prediction and response in the training data

CHAPTER 6

Integrating cancer biological domain knowledge in machine learning models

As detailed in the Chapter describing the medical data, the feature set can be subdivided into subsets of genes based on cancer biological knowledge (Chapter 2). These subsets are referred to as signature gene sets, and each set is named according to the specific part of cancer biology it represents. In this thesis, various methods were investigated to leverage this domain knowledge. First, we investigated feature engineering by principal component analysis (PCA), followed by the ensemble model stacking. Finally, we assessed the effect of including interaction terms between groups of features composed of the signature gene sets. This was done as prolongation of the PCA. In contrast to what we did for the naive machine learning models, we focused on the ROR score here and consequently used a repeated cross-validation approach to fit 1000 models.

6.1 Group and sparse group lasso

Since the grouping of the features (the signature gene sets) is to be exploited it would be beneficial to apply sparsity at group level. To achieve this, we need models that provide regularization and selection at group level. In this specific section, we discuss two penalized linear regression models that apply regularization and selection at group level. These models could be natural choices to address our problem. The group lasso accomplish this by penalizing the features on a group-wise basis, i.e. the features within one group are penalized collectively (Yuan and Lin 2006). This approach has the following minimization problem

$$\hat{\boldsymbol{\beta}}_{„} = \arg\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \left\{ \|y - X\boldsymbol{\beta}\|_2^2 + \lambda \sum_{k=1}^{K} \sqrt{p_k} \|\boldsymbol{\beta}^{(k)}\|_2 \right\}, \tag{6.1}$$

where y is the response variable, X is the design matrix, $\boldsymbol{\beta}$ is the coefficient vector, p_k is the number of predictors in each group k, $\boldsymbol{\beta}^{(k)}$ represents the coefficient vector for group k (i.e. a subvector of $\boldsymbol{\beta}$) and K is the total number of groups. The first term represents the standard RSS, while the second term

is a penalty that encourages sparsity at the group level, where λ controls the degree of regularization.

Sparse group lasso, is a refinement of group lasso, which also takes into account sparsity within the groups (Simon et al. 2013). The model is defined by

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \left\{ \|y - X\boldsymbol{\beta}\|_2^2 + \lambda \sum_{k=1}^{K} \sqrt{p_k} \|\boldsymbol{\beta}^{(k)}\|_2 + \gamma \sum_{k=1}^{K} \sqrt{p_k} \|\boldsymbol{\beta}^{(k)}\|_1 \right\}. \tag{6.2}$$

Thus, sparse group lasso is similar to the group lasso, but with an additional penalty term that promotes sparsity within each group. The amount of regularization is controlled by the parameters λ and γ. Typically λ and γ sum to one. This approach addresses both group-wise sparsity and within-group sparsity.

The signature gene sets contain overlapping genes, and both of these models are designed for grouping by partitioning the feature set. Therefore, they are not appropriate for addressing our problem (Park et al. 2015).

6.2 Dimensionality reduction by domain knowledge guided PCA

In order to represent the data set in a lower dimensional space while retaining as much of the predictive information as possible we applied PCA. However, we did not do this on the full data set, but instead we applied the PCA separately on the sub-feature spaces as defined by the signature gene sets. In this section we first give a short review of the PCA method. Afterwards, we present the results with respect to feature selection and in the end the performance measure of using this approach.

The PCA

In brief, PCA involves projecting the original data points onto a set of orthogonal vectors, known as principal components (PCs). By data points, we here refer to the feature space. The PCs are oriented to capture the maximum variance in the data. Mathematically, PCA is performed by computing the empirical covariance matrix of the centered data with respect to the features. This matrix becomes symmetric, which means its eigenvectors are orthogonal and can thus be utilized as PCs. The corresponding eigenvalues indicate the amount of variance captured by each vector and can thus be used to select a subset of the PCs that explains most of the variability in the feature space. The PCs create a new set of features which are linearly uncorrelated, and a selection of these, typically much fewer than in the original data set, will capture most of the variance in the data (Jolliffe 2002). From a practical point of view PCA can filter out irrelevant features and aggregate correlated features. By selecting only the PCs that explain the majority of the variance in the data, PCA effectively filters out the feature directions with lower variance. Consequently, only the information from the selected PCs, and thus, the information from the

original features contributing to these PCs, is retained. The importance of less informative or irrelevant features, which contribute less to the overall variance will be reduced. Since the PCs are linear combinations of the original features, they aggregate correlated features. Correlated features exhibit a high amount of shared variance, which is identified by the PCs. Thus, highly correlated features are combined into single components. Then by projecting the original data onto the PCs, PCA represents the correlated features in a lower-dimensional space. This can lead to less noise and reveal underlying patterns in the data that are of relevance for predictive power. The PCA algorithm is outlined in Algorithm 2.

Results of the PCA based approach for utilizing domain knowledge

We applied PCA to six signature gene sets, each representing different potentially important functional units in breast cancer biology. These included regulation of cell proliferation (2), cell migration (2), immune cell infiltration (2), DNA repair (1), estrogen receptor signaling (1) and angiogenesis (2). The numbers in parentheses denote the number of selected principal components (PCs) that explained at least 90% of the variance in the data of each signature gene set. Subsequently, we used the predictor values corresponding to each PC as features in machine learning models. For both lasso and elastic net, all PCs were selected in at least one model. The selection frequency is displayed in Table 6.1.

In both models, the two PCs from cell migration showed relatively low selection frequencies. This suggests that cell migration may not have a significant predictive effect on the targeted treatment. We observed that the estrogen receptor signaling, the target of the drug, was selected approximately half of the time. We, thus, could have anticipated a somewhat higher frequency for this. Immune cell infiltration and angiogenesis (development of blood vessels) are not specific targets of the drugs, but they exert substantial influence on tumor growth and are therefore important predictors of cancer outcome in general.

Table 6.1: Selection frequency (in %) of PC's for the various signature gene sets in 1000 models fitted using lasso or elastic net

	Lasso	Elastic net
Angiogenesis 1	20.6	24.6
Angiogenesis 2	71.1	73.8
Cell migration 1	5.1	6.8
Cell migration 2	5.9	6.6
Cell proliferation 1	35.1	45.5
Cell proliferation 2	3.0	3.5
DNA repair 1	47.5	51.3
Estrogen receptor signaling	46.4	55.0
Immune infiltration 1	63.5	66.8
Immune infiltration 2	51.4	55.5

Algorithm 2 Principal Component Analysis (PCA)

Given a data matrix $X \in \mathbb{R}^{n \times p}$ and the desired number of principal components k.

1. Standardize the data within each feature so that the data have zero mean and unit variance. This ensures that all features are on a comparable scale preventing a potential influence from differences in scales of the features in the original data.

$$\bar{x}_{ij} = \frac{x_{ij} - \mu_j}{SD_j}.$$

2. Compute the empirical covariance matrix $C \in \mathbb{R}^{p \times p}$ of the standardized data $\bar{x}_{ij}$. The covariance matrix is a $p \times p$ symmetric matrix, where the entry c_{gj} represents the covariance between the g^{th} and j^{th} features. Since the data $\bar{X}$, the matrix of the standardized data $\bar{x}_{ij}$, now is centred C can be calculate by

$$C = \frac{1}{n-1} \bar{X}^T \bar{X}.$$

3. Calculate the eigenvalues λ_i and eigenvectors $\boldsymbol{v}_i$ of the covariance matrix C using singular value decomposition,

$$C\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i.$$

4. Sort the eigenvalues in descending order, and select the top k eigenvectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ corresponding to the top k eigenvalues $\lambda_1, \ldots, \lambda_k$.

5. Create a projection matrix $V \in \mathbb{R}^{p \times k}$ with the selected eigenvectors as columns

$$V = \begin{bmatrix} | & | & & | \\ \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_k \\ | & | & & | \end{bmatrix}.$$

   The eigenvectors $\boldsymbol{v}$ are the PCs.

6. Compute the reduced data matrix $\tilde{X} \in \mathbb{R}^{n \times k}$ by projecting the standardized data onto the k selected PCs in V

$$\tilde{X} = \bar{X} \cdot V$$

   $\tilde{X}$ are now the PC based transformed feature data. $\tilde{X}$ has k linearly uncorrelated features that capture most of the variance in the original data.

Performance of PCA-based models with signature gene sets

We examined the correlation between predicted and true values in the test sets when using the 6 signature gene sets to generate PC's. Ridge regression demonstrated better performance than both lasso and elastic net, which exhibited comparable results to each other (Table 6.2). However, all models performed at a similar level as their corresponding naive models analyzed in Chapter 4. Notably, the PCA-based models used a total of 332 genes, which is fewer than the 771 genes used for the naive models.

To determine whether incorporating more information could enhance performance, we added four additional groups of genes to the analysis: antigen presentation, apoptosis, cytokines and chemokine, and tumor metabolism. This addition led to a larger mean correlation for all models, however, the variance is substantial, making hard to draw conclusions (Table 6.2). Furthermore, when comparing the MSE we did not observe any significant reduction (Table 6.3).

In conclusion, we observed a tendency for improved performance by using PCA guided by biological domain knowledge when using correlation as a performance measure. Additionally, we gained insights into the predictive power of genes related to different functional aspects of cancer biology, which may be utilized in the development of more refined models.

Table 6.2: Predictive performance of PCA based on domain knowledge. Mean (SD) correlation between prediction and observed values in test sets.

Model	Naive analysis	6 groups	10 groups
Ridge	0.30 (0.278)	0.32 (0.302)	0.41 (0.240)
Lasso	0.08 (0.277)	0.14 (0.296)	0.32 (0.271)
Elastic net	0.20 (0.311)	0.15 (0.295)	0.43 (0.261)

Table 6.3: Predictive performance of PCA based on domain knowledge. Mean (SD) MSE between prediction and observed values in test sets.

Model	Naive analysis	6 groups	10 groups
Ridge	358 (150)	360 (148)	344 (154)
Lasso	394 (159)	388 (154)	373 (159)
Elastic net	427 (186)	385 (152)	366 (168)

6.3 Two-stage model based on domain knowledge

In another effort to include the domain knowledge of the signature gene sets in a machine learning model, we explored a two-stage structure. This method is based on the ensemble model often referred to as generalized stacked models or simply stacking (Wolpert 1992, Breiman 1996). Most often in stacking, multiple models are first trained independently on the same dataset. The predictive outputs of these models then serve as input features for a second-stage model. The function of the second stage model is to merge the predictions from the first-stage models, creating a single comprehensive aggregated prediction.

The idea behind stacking is that the different models will capture distinct complexity in the data and that errors will be averaged out. It combines

the strengths of different models by leveraging their individual predictions to enhance overall performance. In the second-stage individual weighting of the prediction from the first-stage models is included, which potentially can reduce overfitting. Therefore, diversity in the first models is important. This is typically done either through utilizing various model types or varying hyperparameters.

However, rather than training various models on the complete data set at the first stage, we trained models on different subsets of the dataset, specifically the signature gene sets. In this way, each signature gene set provides a prediction of the response and these are combined in stage-two to possibly enhance overall performance. As diversity at the first stage is achieved by the varying data, we could use the same model type for all the subsets at the first stage. We utilized ridge regression at the first-stage, as predictive power is more critical at this stage than feature selection. For the second-stage model either ridge regression, lasso or elastic net was employed to combine the predictions from the first-stage models (see Algorithm 3)

Algorithm 3 Two-stage approach for integrating feature groups based on domain knowledge

1. **First stage**
   a) Construct K groups of features (e.g. genes) based on domain knowledge.
   b) Regress outcome variable $\boldsymbol{y}$ on each group of features via a ridge model separately.
   c) Predict outcomes $\hat{\boldsymbol{y}}^{(k)}$ $(k = 1, \cdots, K)$ based on each fitted ridge model.
2. **Second stage**
   Apply a machine learning model to regress the outcome variable $\boldsymbol{y}$ on $\hat{\boldsymbol{y}}^{(1)}, \cdots, \hat{\boldsymbol{y}}^{(K)}$.

In the stacking approach, we did not observe any significant differences in the performance of the three model classes when evaluating the correlation and MSE between predictions and responses in the test sets (Table 6.4). Moreover, the performance was on par with that of the standard ridge models. Nevertheless, stacking can potentially offer insights into the importance of various groups. To explore this further, we assessed the selection frequency and estimated the coefficient size of when using lasso in the stage-two model using the same six gene sets as in the PCA above ((Table 6.4)). Our observations revealed that gene sets representing proliferation, estrogen receptor signaling and DNA repair mechanisms exhibited the highest selection frequency. This finding is intriguing, as the target drug in the clinical trial aims to modulate the signaling pathway between estrogen stimulation of the cell and the regulation of cell replication. In terms of coefficient size, the SD is relatively high when compared to the estimated values so it is difficult to interpreted the results.

Table 6.4: Predictive performance of models using domain knowledge

Model	Correlation mean (SD)	MSE mean (SD)
Ridge	0.363 (0.254)	367 (171)
Lasso	0.337 (0.261)	386 (179)
Elastic	0.342 (0.258)	383 (177)

Table 6.5: Selection frequency of the different groups of genes and their estimated coefficient size of the stage-two model using lasso

Signature gene set	Selection (%)	Mean coefficient size (SD)
DNA repair	78.2	0.61 (3.02)
Immune infiltration	66.4	0.57 (1.02)
Proliferation	78.2	1.45 (2.99)
Estrogen receptor signaling	75.1	0.93 (3.38)
Antagonises	53.8	-2.97 (8.82)
Cell migration	50.2	-2.14 (5.28)

6.4 Interactions based on domain knowledge

In cancer biology different parts of the system interact in ways that cause more complex effects than just additive effects. These effects are called synergistic effects and can be implemented in regression models by introducing interaction terms.

From a statistical point of view, this is challenging in high dimensional data because the number of parameters to fit increases substantially. The number of interaction terms of order k is calculated as $\binom{p}{k}$, where p is the number of measured variables. For our dataset of 771 genes even if we focused only on pairwise interactions ($k = 2$), the number of interaction terms amounts to almost three hundred thousand. As a substantial amount of these interactions is not relevant it would be natural to assume sparsity and employ the lasso regression. However, the ordinary lasso treats main variables and interaction variables equally and could potentially select an interaction term while ignoring its corresponding main effects. E.g., if two variables are correlated, which can happen between main terms and interaction terms when they effect the response variable similarly, lasso tends to select one of them as they contain similar information about the response variable. It is a well-established practice among statisticians when fitting models to include an interaction term only if the corresponding main effects are also present in the model. A solution to this problem is outlined in the model called hierarchical lasso (Bien, Taylor and Tibshirani 2013). This method is computationally super intensive and it doesn't handle high dimensional data well as our dataset is.

However, in the field of cancer biology assessing interactions between every gene is not necessary constructive. As outlined in the introduction, cancer biology can be partitioned into distinct functional units (see Chapter 1). Therefore, it is more natural to investigate the interactions between genes within these functional units or examining the interactions between the units

themselves. As our objective is to explore prediction based on a dataset that covers many such units we have focus on the latter. The signature gene sets are designed to represent potential relevant functional units in breast cancer biology and we have aimed to consider interactions between them using the condensed feature produced by the PCA. Thus, the regression model based on dimension reduction on the signature gene sets and including pair-wise interactions, can be formulated

$$y_i = \beta_0 + \sum_{k=1}^{K}\sum_{j=1}^{p_k} \beta_j^k \tilde{x}_{ji}^k + \sum_{k=1}^{K}\sum_{m>k}^{K}\sum_{j=1}^{p_k}\sum_{h=1}^{p_m} \gamma_{jh}^{km} \tilde{x}_{ji}^k \tilde{x}_{hi}^m + \varepsilon_i, \tag{6.3}$$

where $\tilde{x}_{ji}^k$ ($\tilde{x}_{hi}^m$) is the value of the jth (hth) PC in the kth (mth) signature gene set for observation i, p_k (p_m) is the number of PC chosen for the kth (mth) group. The β_j^k are the coefficients of the main effects and the γ_{jh}^{km} are the coefficients of the interaction effects.

We have also introduced penalization in the form of ridge, lasso, and elastic net as was done when not including interaction terms.

With this approach, the predictions and response values of the test set showed higher correlation and the MSE decreased for all models (Table 6.6). Although the variance is large, this indicates that incorporating interactions between cancer biological functional units can enhance the predictive performance of a machine learning model.

Table 6.6: Predictive performance when using interactions between groups defined by the signature gene sets based PC transformed features. The mean (SD) of both correlation and MSE computed for the prediction and true values over the test sets.

	Correlation		MSE	
Model	no interaction	interactions	no interaction	interactions
Ridge	0.32 (0.302)	0.53 (0.246)	360 (148)	294 (158)
Lasso	0.14 (0.295)	0.43 (0.267)	388 (154)	339 (167)
Elastic	0.15 (0.295)	0.45 (0.265)	385 (152)	331 (166)

To interpret potentially interesting interactions, we performed the selection models using only the first PC of each signature gene set. This revealed a selection frequency of almost 100% of the interaction between cell proliferation and estrogen receptor signaling (see Table 6.7). The coefficient size was also substantially higher than the rest. Since the cancer treatment investigated targets the intracellular pathway linking estrogen stimulation of the cells and cell proliferation, this was an interesting observation.

Table 6.7: Feature selection and estimated coefficient values of interaction terms in the domain knowledge guided PCA approach.

Interaction	Selection frequency		Mean coefficient values	
	Lasso	Elastic net	Lasso	Elastic net
DNA repair * immune infiltration	59.6	69.6	0.0631	0.0715
DNA repair * cell proliferation	30.1	37.5	-0.0294	-0.0307
DNA repair * estrogen receptor signaling	73.8	90.8	0.1301	0.1549
DNA repair * angiogenesis	2.7	3.8	-0.0004	0.0000
DNA repair * cell migration	73.9	83.8	0.0796	0.0845
immune infiltration * cell proliferation	1.8	2.9	-0.0029	-0.00398
immune infiltration * estrogen receptor signaling	11.8	13.2	0.0107	0.0099
immune infiltration * angiogenesis	17.5	22.6	-0.0299	-0.0321
immune infiltration * cell migration	7.4	9.3	0.0078	0.0089
cell proliferation * estrogen receptor signaling	96.1	97.6	0.3532	0.3167
cell proliferation * angiogenesis	10.0	14.6	-0.0078	-0.0099
cell proliferation * cell migration	21.7	32.4	0.0092	0.0125
estrogen receptor signaling * angiogenesis	2.9	3.1	-0.0147	-0.0148
estrogen receptor signaling * cell migration	54.3	74.3	0.1182	0.1427
angiogenesis * cell migration	3.0	3.7	-0.0007	-0.0004

CHAPTER 7

Exploring a new approach for group interactions

As elaborated in Chapter 6, in the classical linear regression framework a model that includes interactions between groups can be relevant in cancer biology. We utilized this in the PCA regression model, however, in that model learning main effect of the single genes was not possible. In this chapter, we explore the possibility of using a model approach which both consider interactions between groups of features and also estimate coefficient for single features.

This work does not encompass a comprehensive model evaluation, but it constituted a significant portion of the thesis work and brought some interesting results.

7.1 The model

To formulate the model, we will use the following notation for the usual covariate matrix X of a linear model

$$X = (\mathbf{1}, \mathbf{x}_1, ...\mathbf{x}_p) \in \mathbb{R}^{n\times(p+1)} \tag{7.1}$$

and for the coefficient vector $\boldsymbol{\beta}$

$$\boldsymbol{\beta}^\top = (\beta_0, \beta_1, ..., \beta_p) \in \mathbb{R}^{1\times(p+1)} \tag{7.2}$$

Let us call $\mathcal{G}$ the set of all gene sets

$$\mathcal{G} = \{G^k : k = 1, ...K\}, \tag{7.3}$$

where we define an index set G^k containing the q_k indexes (i.e. $g_1^k, \cdots, g_{q_k}^k$) of the genes in the kth gene set:

$$G^k = \{g_1^k, ..., g_{q_k}^k\} \tag{7.4}$$

Note that the gene sets are not disjoint in general, so there may exist k and l for which $G^k \cap G^l \neq \emptyset$. We further define a submatrix of X associated to the index set G^k

$$X_{G^k} = (\mathbf{x}_{g_1^k}, ..., \mathbf{x}_{g_{q_k}^k}) \in \mathbb{R}^{n\times q_k} \tag{7.5}$$

and a subvector of the coefficient vector $\boldsymbol{\beta}$ associated to the index set G^k

$$\boldsymbol{\beta}_{G^k}^{\top} = (\beta_{g_1^k}, ..., \beta_{g_{q_k}^k}) \in \mathbb{R}^{1 \times q_k} \tag{7.6}$$

We propose the following model

$$Y = X\boldsymbol{\beta} + \sum_{\{k,l:G^k,G^l \in \mathcal{G}, k<l\}} (X_{G^k}\boldsymbol{\beta}_{G^k}) \odot (X_{G^l}\boldsymbol{\beta}_{G^l})\gamma_{kl} + \boldsymbol{\varepsilon}, \tag{7.7}$$

where $\odot$ denotes the Hadamard product, i.e., element-wise multiplication of the two column vectors and γ_{kl} are the interaction terms between the groups. Note that the model can include variables in the main effect term $X\boldsymbol{\beta}$ not assigned to a specific group of features.

7.2 Characteristics of the model

To gain an understanding of the model, we examined it under the simplest data configuration. Consider a scenario with features (e.g., gene expression levels) from only two groups (e.g., representing two functional units in cancer biology) and just two features within each group. Specifically, let X_Z and X_W be the $n \times 2$ matrices representing the features of the two groups. In this case, the model is given in the following form

$$Y = X_{ZW}\beta_{ZW} + (X_Z\beta_Z \odot X_W\beta_W)\gamma + \varepsilon, \tag{7.8}$$

where X_Z and X_W are merged into a single term by defining $X_{ZW} = (X_Z, X_W)$ and β_{ZW} is a concatenated vector consisting of both β_Z and β_W. γ is the interaction coefficient between group Z and W (indexes is drooped since there is only one interaction term in this model).

To uncover properties of the model, we computed the interaction term of equation 7.8 for a single observation $(x_{z1}, x_{z2}, x_{w1}, x_{w2}, y)$ where each element represents either a feature or the response. The interaction term of equation 7.8 can then be expressed as

$$(x_{z1}\beta_{z1}x_{w1}\beta_{w1} + x_{z1}\beta_{z1}x_{w2}\beta_{w2} + x_{z2}\beta_{z2}x_{w1}\beta_{w1} + x_{z2}\beta_{z2}x_{w2}\beta_{w2})\gamma. \tag{7.9}$$

This elucidate two properties of the model. First, we observe that the model represents pairwise interactions between elements of the two groups, with a shared interaction coefficient in γ. These interactions satisfy an hierarchical structure. The hierarchical structure in statistical models refers to a model-building approach where lower-order terms (e.g., main effects) are included in the model before considering higher-order terms (e.g., interactions). Adherence to the hierarchical structure is regarded as a best practice in the development of models. Specifically, interaction terms should only be incorporated if the main effects of the involved features are already present in the model. This practise often ensures a more meaningful interpretation of the interaction effects and helps in reducing potential biases in their coefficient estimates.

When utilizing a model class with feature selection in a model containing interaction terms, it is possible for the model to select interaction terms without selecting the main effect term with the corresponding features. This situation can arise when interaction variables are correlated with the original variables

from which they are derived. However, our model approach prevents this from happening, as features that are not selected as main effects will not appear in the interaction term.

The second notable property of the model involves the weighting of various interaction terms by the main effects of the corresponding features. As a result, the features with the largest main effects contribute the most to the interaction effects. The relevance of such property has been emphasized by Cox (1984), who asserted that "Large component main effects are more likely to lead to appreciable interactions than small components." In simpler terms, Cox states that if the main effects of two features are large, there is a higher chance that their interaction is of significant importance. Taking this into account, the weighting by the main effects incorporated in the interaction terms in our model may enhance the statistical power in an analysis, making it easier to detect significant interactions between features when they are present.

One limitation of our model approach is that the common interaction term for all pairwise interactions is a rather strong constraint on the interaction parameter. On the other hand, this also contribute with structural sparsity, a desirable property. In scenarios where two features exhibit large main effects but no interaction effects between them, the model may mask interaction effects of features with smaller main effects. Nevertheless, this emphasize that the model does not focus on single interaction between two features but rather on group-wise interactions. In this model, the interaction effect is derived from the sum of all quadratic main estimates multiplied by their corresponding quadratic feature values.

7.3 The algorithm

The challenge with this model lies in its non-linearity with respect to the parameters, as the interaction term contains the main effect and the interaction effect parameters, causing identifiability issues. This makes it impossible to uniquely estimate the values of the parameters in the interaction terms simultaneously. To address this, we propose an iterative approach in which the content inside the parentheses of equation 7.7 is treated as constant during the actual fitting process. Consequently, only the β parameters inside the main effect term and the γ parameters of the interaction term are estimated in each iteration. The β values inside the interaction term are subsequently updated iteratively after the fitting process, becoming the values of the main effect βs from the previous iteration. This algorithmic approach is outlined in algorithm 4.

Algorithm 4 Synergistic learning model algorithm

1. **Input:** $X, Y, \mathcal{G} = \{G^k\}$
2. **Initialize:**
   - Initialize the main parameters $\boldsymbol{\beta}^{(0)}$ using ridge regression on the model without the interaction term:
   $$\boldsymbol{\beta}^{(0)} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \|Y - X\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_2^2 \right\}$$
   - Initialize any interaction term $f_{G^k} \odot f_{G^j}$ $(k < j)$, where
   $$f_{G^k} = X_{G^k} \boldsymbol{\beta}_{G^k}^{(0)},$$
   $$f_{G^j} = X_{G^j} \boldsymbol{\beta}_{G^j}^{(0)}$$
3. **For L-th iteration** :
   - Estimate main effects $\boldsymbol{\beta}^{(L)}$ and interaction effects $\boldsymbol{\gamma}^{(L)}$ using elastic net
   $$(\boldsymbol{\beta}^{(L)}, \boldsymbol{\gamma}^{(L)}) = \underset{\boldsymbol{\beta}, \boldsymbol{\gamma}}{\operatorname{argmin}} \Bigg\{ \|Y - X\boldsymbol{\beta} - \sum_{\{k,j: G^k, G^j \in \mathcal{G}, k<j\}} (f_{G^k} \odot f_{G^j}) \boldsymbol{\gamma}_{ij} \|_2^2 + \boldsymbol{\lambda} \left(\alpha \left\| \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{\gamma} \end{bmatrix} \right\|_1 + \frac{1-\alpha}{2} \left\| \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{\gamma} \end{bmatrix} \right\|_2^2 \right) \Bigg\}$$
   - Update the interaction terms
   $$f_{G^k} = X_{G^k} \boldsymbol{\beta}_{G^k}^{(L)},$$
   $$f_{G^j} = X_{G^j} \boldsymbol{\beta}_{G^j}^{(L)}$$
   - Check for convergence criteria. If it does not converge, $L \leftarrow L + 1$
4. **Output:** main effects $\boldsymbol{\beta}$ and interaction effects $\boldsymbol{\gamma}$

We used the elastic net model with an alpha value of 0.5 to solve the estimation problem in step 3 of the algorithm.

As stooping criteria we have used a combination of a maximum number of iterations (100) and the stationarity of the model deviance.

7.4 Model testing on simulated data

To evaluate the model, we simulated datasets with three different sample sizes ($n = 50, 100, 500$). The datasets each contained 120 features, where 20 had effect on the response variable and 100 parameters did not. The 20 responsive

parameters were divided into four groups of 5. X and $\boldsymbol{\beta}$ were generated from a normal distribution and a uniform distribution, respectively

$$X \sim N(0,1)$$
$$\boldsymbol{\beta} \sim U(-1,1)$$

The interaction effect γ between groups were set to 1. The simulated response was generated by using an ordinary linear model. We considered two scenarios as follows:

- **Scenario I:** Set one interaction $f_{G^1} \odot f_{G^2}$, where
$$f_{G^1} = X_{G^1}\beta_{G^1},$$
$$f_{G^2} = X_{G^2}\beta_{G^2}.$$
Simulate responses
$$Y = X\beta + (f_{G^1} \odot f_{G^2})\gamma_{12} + \varepsilon,$$
where the noise term is simulated from the standard normal distribution.
- **Scenario II:** Set three interactions $f_{G^1} \odot f_{G^2}$, $f_{G^1} \odot f_{G^4}$ and $f_{G^2} \odot f_{G^4}$, where
$$f_{G^1} = X_{G^1}\beta_{G^1},$$
$$f_{G^2} = X_{G^2}\beta_{G^2},$$
$$f_{G^4} = X_{G^4}\beta_{G^4},$$
Simulate responses
$$y = X\beta + (f_{G^1} \odot f_{G^2})\gamma_{12} + (f_{G^2} \odot f_{G^4})\gamma_{24} + (f_{G^1} \odot f_{G^4})\gamma_{14} + \epsilon.$$

Table 7.1 summarizes the selection frequency of the interaction terms in both scenarios across 1000 simulations. At a low sample size, the model fails to select interaction terms for both scenarios, suggesting that it may not perform well in high-dimensional data settings. However, promising results were observed for a sample size of 500. At an intermediate sample size (n=100), the model effectively identified one interaction term, but the results for three interaction terms were unsatisfactory.

In general, the model rarely selected incorrect interactions, resulting in high specificity (Table 7.2). However, sensitivity was low for smaller sample sizes. There may be an option to fine-tune the model for higher sensitivity, but this would likely come at the expense of the specificity.

Table 7.1: Selection performance on the interaction terms in percentage using simulated data

Scenario	n	$f_{G^1} \odot f_{G^2}$	$f_{G^1} \odot f_{G^3}$	$f_{G^1} \odot f_{G^4}$	$f_{G^2} \odot f_{G^3}$	$f_{G^2} \odot f_{G^4}$	$f_{G^3} \odot f_{G^4}$
I	50	4	2	0	2	1	1
	100	78	11	12	14	14	8
	500	100	0	0	2	1	1
II	50	1	1	0	0	0	0
	100	18	10	17	2	16	3
	500	100	15	100	5	100	12

Table 7.2: Sensitivity and specificity of the selection of the interaction terms with the simulated data.

Scenario	n	Sensitivity	Specificity
I	50	0.04	0.98
	100	0.76	0.88
	500	1	0.99
II	50	0.003	0.003
	100	0.17	0.95
	500	1	0.89

Next, we evaluated the model using the repeated cross-validation method to assess its performance in terms of prediction accuracy (Table 7.3). We calculated the correlation of the test data, but only when the model selected the interaction terms (as shown in the table 7.1). In this analysis, we employed 100 simulations. For the lowest samples size for scenario I few models were fitted with the interaction term, but they performed rather good. The best perdition performance was achieved with the highest sample size.

Table 7.3: Correlation of the models using simulated data

	n	Correlation	SD	MSE	SD
I	50	0.59	0.163	0.96	0.076
	100	0.64	0.105	0.76	0.157
	500	0.93	0.019	0.16	0.038
II	50	-	-	-	-
	100	0.64	0.181	0.94	0.111
	500	0.93	0.020	0.16	0.65

7.5 Testing the model on the dataset of the clinical trial

We attempted to apply the model to the dataset derived from the clinical trial, including only the interaction term between the signature group sets of proliferation and estrogen receptor signaling. However, the model did not select the interaction term in any of the 1,000 models generated during the repeated cross-validation process. This outcome is in accordance with the findings from the simulated data, suggesting that the model may not be suitable for high-dimensional datasets.

CHAPTER 8

Discussion and further perspective

In this thesis, we analyzed a dataset composed of 771 features, representing gene expression levels, and two response variables reflecting cancer development and prognosis. The primary objective was to evaluate the performance of machine learning models in such a setting, with particular emphasis on leveraging cancer biology domain knowledge. Success in such endeavor would be of importance for selecting individual patients for the most promising treatment regimens.

Initially, we evaluated standard model classes used for high-dimensional data in machine learning, including ridge regression, elastic net and boosting. In terms of predictive performance, ridge regression emerged as the most successful. Next, we explored incorporation of domain knowledge in machine learning models. The idea involved utilizing the genes belonging to functional units within the cancer biological ecosystem of the tumor. This was achieved by modeling gene expression in machine learning models that allowed grouping structures. Performance-wise, we found the most success with PCA regression, which outperformed standard ridge regression. However, we must be careful with drawing conclusions, as the variability in our analysis is relatively high. Nevertheless, the high-dimensional setting is generally challenging, and our results suggest that the incorporation of domain knowledge is a promising direction, although further studies is clearly necessary.

The PCA approach enabled us to introduce interactions between groups. An interesting result was the strong effect between genes belonging to the proliferation signature gene set and the genes of the estrogen signaling, as the drug investigated in the clinical trial targets the signaling pathway between the estrogen receptor and regulation of cell proliferation. It is not surprising that genes within this pathway are important for the treatment effect of a drug targeting this pathway, however, it is a confirmation of the model's usefulness. More importantly, this finding also suggests that it may be possible to identify genes that can serve as markers in treatment selections.

When selecting PCs that explained at least 90% of the variance in the predictors, typically between one and two components were chosen. This suggests that the gene expression of many genes covaries. This finding implies that selecting potential genetic markers may not necessitate involving too many genes. However, a challenge with mRNA expression is the short and dynamic levels within the cell, which further has an indirect contribution to the cell's functionality through the proteins. This means that at two different time points of the same patient the mRNA level of a particular gene might be very

different while the protein level is the same and it is the latter that directly determines the cells response to a treatment. Therefor, it is difficult to use single mRNA as markers, as the amount in a sample is dependent on the time point of sampling. However, if multiple genes point in the same direction, their aggregated measurement can provide a more reliable marker.

We also utilized a stacking approach, which demonstrated comparable prediction performance to ridge regression. We assessed its potential for unraveling the underlying cancer biology, as this method provides the opportunity to examine both the main effects of gene expressions and interaction effects between groups. Our analysis highlighted the significance of genes involved in proliferation and estrogen signaling mechanisms.

In an attempt to integrate both individual feature main effects and interaction effects between the groups of features in a linear model, we investigated the application of an iterative approach as described in Chapter 7. The evaluation using simulated data demonstrated satisfactory results when working with large amounts of data. However, challenges emerged when the number of features surpassed the number of observations. Notably, in comparison to the other models tested, this approach is unique in that it incorporates both group interactions and main effects. Consequently, it offers potential advantages beyond the capabilities of the other models examined, and there is potential for further development of the model by extending the algorithm, e.g., by integrating the adaptive lasso. This method can be used to selectively penalize only the main effects, applying a reduced penalty on the group interaction terms (Zou 2006). This technique can potentially accelerate model convergence with respect to the coefficients, as smaller coefficients are more heavily penalized and are more likely to shrink towards zero. Another potential improvement is using alternative convergence criteria, such as directly focusing on changes in coefficient size instead of model deviance as we did.

A limitation of our study is the lack of testing against an external dataset. Additionally, our analysis with repeated cross-validation gave rather high variability. This might be caused by the small test set of around 10 observations. Consequently, we are far from being able to conclude that any of these model approaches would be useful in clinical settings. However, as a comparison of the model approaches, it suggests that incorporating domain knowledge could benefit a prognostically useful model. Moreover, for research purposes, this strategy gives promising opportunities for understanding cancer biology.

Finally, we would like to mention some of the genes whose expression was consistently selected in many of the model analysis scenarios and also showed values significantly distinct from zero in ridge regression. HDAC2 and LEFT2 were persistently selected in all model scenarios, while GATA3 was almost exclusively selected in the analyses using the proliferation score as a response variable. GATA3 (GATA Binding Protein 3) is a transcription factor that plays a crucial role in cell lineage determination and differentiation. It belongs to the proliferation signature gene set. In breast cancer, GATA3 is recognized as a key regulator of luminal epithelial cell differentiation and is often used as a marker for luminal breast cancer subtypes (Mehra et al. 2005). Abnormal expression of GATA3 has been associated with tumor progression and poor prognosis in breast cancer patients. LEFTY2 (Left-Right Determination Factor 2) is a member of the signature gene set representing the transforming growth factor-beta (TGF-beta) superfamily, which plays a role in many developmental processes

in the body. In cancer, LEFTY2 has been implicated in the regulation of cell proliferation and differentiation. Dysregulation of LEFTY2 expression has been associated with various cancer types (Yue and Mulder 2001). HDAC2 (Histone Deacetylase 2) is an enzyme involved in the regulation of gene expression through regulating the chromatin structure and thereby repression of gene transcription. Abnormal expression and activity of HDAC2 have been observed in various cancer types, including breast cancer. Altered HDAC2 expression has been linked to tumor progression and HDAC inhibitors have emerged as a potential therapeutic strategy for cancer treatment (Huang et al. 2015).

The potential of these genes, along with other selected genes, could be further evaluated by conducting similar analyses within the chemotherapy arm of the clinical trial. This would enable the determination of whether some of these markers are specifically important for the targeted drug treatment, rather than merely exhibiting a general effect on cancer treatments that inhibit cancer cell proliferation. Furthermore, since the mechanistic model using only six genes has proven to be quite successful, and it is known that the targeted drug influences more than just CDK4/6, there is potential for further development of this model (Fassl, Geng and Sicinski 2022). Genes identified in our study could serve as potential candidates for such advancements.

In brief, this thesis focused on the evaluation of machine learning models for cancer precision medicine, with an emphasis on leveraging cancer biology domain knowledge. In conclusion, our findings suggest that incorporating domain knowledge related to the tumor ecosystem into the modeling process has the potential to improve their prognostic utility and thereby assist in the selection of individualized treatment regimens and contribute to a deeper understanding of cancer biology.

Appendices

Bibliography

Aiken, L. S., West, S. G. and Reno, R. R. (1991). *Multiple regression: testing and interpreting interactions*. Newbury Park, Calif: Sage Publications. 212 pp.

Altman, D. G. and Royston, P. (6th May 2006). 'The cost of dichotomising continuous variables'. In: *BMJ : British Medical Journal* vol. 332, no. 7549, p. 1080.

Azuaje, F. (25th Feb. 2019). 'Artificial intelligence for precision oncology: beyond patient stratification'. In: *npj Precision Oncology* vol. 3, no. 1. Number: 1 Publisher: Nature Publishing Group, pp. 1–5.

Barretina, J. et al. (2012). 'The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity'. In: *Nature* vol. 483, no. 7391. Publisher: Nature Publishing Group, pp. 603–607.

Bien, J., Taylor, J. and Tibshirani, R. (June 2013). 'A LASSO FOR HIERARCHICAL INTERACTIONS'. In: *Annals of statistics* vol. 41, no. 3, pp. 1111–1141.

Bray, F. et al. (Nov. 2018). 'Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries'. In: *CA: a cancer journal for clinicians* vol. 68, no. 6, pp. 394–424.

Breiman, L. (1st July 1996). 'Stacked regressions'. In: *Machine Learning* vol. 24, no. 1, pp. 49–64.

Burstein, H. J. et al. (10th Dec. 2021). 'Endocrine Treatment and Targeted Therapy for Hormone Receptor-Positive, Human Epidermal Growth Factor Receptor 2-Negative Metastatic Breast Cancer: ASCO Guideline Update'. In: *Journal of Clinical Oncology: Official Journal of the American Society of Clinical Oncology* vol. 39, no. 35, pp. 3959–3977.

Chen, T. and Guestrin, C. (13th Aug. 2016). 'XGBoost: A Scalable Tree Boosting System'. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794. arXiv: 1603.02754 [cs].

Cox, D. R. (1984). 'Interaction'. In: *International Statistical Review / Revue Internationale de Statistique* vol. 52, no. 1. Publisher: [Wiley, International Statistical Institute (ISI)], pp. 1–24.

Fassl, A., Geng, Y. and Sicinski, P. (1st Jan. 2022). 'CDK4 and CDK6 kinases: From basic science to cancer therapy'. In: *Science (New York, N.Y.)* vol. 375, no. 6577, eabc1495.

Friedman, J. H., Hastie, T. and Tibshirani, R. (2nd Feb. 2010). 'Regularization Paths for Generalized Linear Models via Coordinate Descent'. In: *Journal of Statistical Software* vol. 33, pp. 1–22.

Goel, S., Bergholz, J. S. and Zhao, J. J. (2022). 'Targeting CDK4 and CDK6 in cancer'. In: *Nature Reviews Cancer* vol. 22, no. 6. Publisher: Nature Publishing Group, pp. 356–372.

Hanahan, D. (12th Jan. 2022). 'Hallmarks of Cancer: New Dimensions'. In: *Cancer Discovery* vol. 12, no. 1, pp. 31–46.

Hastie, T., Tibshirani, R. and Friedman, J. (1st Feb. 2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics)*.

He, W. et al. (26th Aug. 2020). 'Mathematical modelling of breast cancer cells in response to endocrine therapy and Cdk4/6 inhibition'. In: *Journal of The Royal Society Interface* vol. 17, no. 169. Publisher: Royal Society, p. 20200339.

Hoerl, A. E. and Kennard, R. W. (1st Feb. 1970). 'Ridge Regression: Biased Estimation for Nonorthogonal Problems'. In: *Technometrics* vol. 12, no. 1. Publisher: Taylor & Francis _eprint: https://www.tandfonline.com/doi/pdf/10.1080/00401706.1970.10488634, pp. 55–67.

Hofner, B. et al. (14th Feb. 2012). *Model-based Boosting in R: A Hands-on Tutorial Using the R Package mboost*. Volume: 120. URL: https://epub.ub.uni-muenchen.de/12754/ (visited on 12/05/2023).

Huang, H. et al. (25th Mar. 2015). 'Quantitative proteomic analysis of histone modifications'. In: *Chemical Reviews* vol. 115, no. 6, pp. 2376–2418.

Ingalls, B. P. (2013). *Mathematical modeling in systems biology: an introduction*.

Jolliffe, I. T. (2002). *Principal Component Analysis*. Springer Series in Statistics. New York: Springer-Verlag.

Li, G., Tian, Y. and Zhu, W.-G. (2020). 'The Roles of Histone Deacetylases and Their Inhibitors in Cancer Therapy'. In: *Frontiers in Cell and Developmental Biology* vol. 8.

Marusyk, A., Janiszewska, M. and Polyak, K. (13th Apr. 2020). 'Intratumor Heterogeneity: The Rosetta Stone of Therapy Resistance'. In: *Cancer Cell* vol. 37, no. 4, pp. 471–484.

Mehra, R. et al. (15th Dec. 2005). 'Identification of GATA3 as a breast cancer prognostic marker by global gene expression meta-analysis'. In: *Cancer Research* vol. 65, no. 24, pp. 11259–11264.

Park, H. et al. (Feb. 2015). 'Sparse Overlapping Group Lasso for Integrative Multi-Omics Analysis'. In: *Journal of Computational Biology* vol. 22, no. 2. Publisher: Mary Ann Liebert, Inc., publishers, pp. 73–84.

Perou, C. M. et al. (17th Aug. 2000). 'Molecular portraits of human breast tumours'. In: *Nature* vol. 406, no. 6797, pp. 747–752.

Prat, A. et al. (1st Jan. 2020). 'Ribociclib plus letrozole versus chemotherapy for postmenopausal women with hormone receptor-positive, HER2-negative, luminal B breast cancer (CORALLEEN): an open-label, multicentre, randomised, phase 2 trial'. In: *The Lancet Oncology* vol. 21, no. 1. Publisher: Elsevier, pp. 33–43.

Raue, A. et al. (2013). 'Lessons learned from quantitative dynamical modeling in systems biology'. In: *PloS one* vol. 8, no. 9. Publisher: Public Library of Science San Francisco, USA, e74335.

Saito, A. et al. (June 2013). 'Suppression of Lefty expression in induced pluripotent cancer cells'. In: *FASEB journal: official publication of the Federation of American Societies for Experimental Biology* vol. 27, no. 6, pp. 2165–2174.

Sammut, S.-J. et al. (Jan. 2022). 'Multi-omic machine learning predictor of breast cancer therapy response'. In: *Nature* vol. 601, no. 7894. Number: 7894 Publisher: Nature Publishing Group, pp. 623–629.

Simon, N. et al. (1st Apr. 2013). 'A Sparse-Group Lasso'. In: *Journal of Computational and Graphical Statistics* vol. 22, no. 2. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/10618600.2012.681250, pp. 231–245.

Sung, H. et al. (2021). 'Global Cancer Statistics 2020: GLOBOCAN Estimates of Incidence and Mortality Worldwide for 36 Cancers in 185 Countries'. In: *CA: A Cancer Journal for Clinicians* vol. 71, no. 3. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.3322/caac.21660, pp. 209–249.

Swanson, K. et al. (13th Apr. 2023). 'From patterns to patients: Advances in clinical machine learning for cancer diagnosis, prognosis, and treatment'. In: *Cell* vol. 186, no. 8. Publisher: Elsevier, pp. 1772–1791.

Sørlie, T. et al. (11th Sept. 2001). 'Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications'. In: *Proceedings of the National Academy of Sciences of the United States of America* vol. 98, no. 19, pp. 10869–10874.

Tay, J. K., Narasimhan, B. and Hastie, T. (23rd Mar. 2023). 'Elastic Net Regularization Paths for All Generalized Linear Models'. In: *Journal of Statistical Software* vol. 106, pp. 1–31.

Tibshirani, R. (1996). 'Regression Shrinkage and Selection via the Lasso'. In: *Journal of the Royal Statistical Society. Series B (Methodological)* vol. 58, no. 1. Publisher: [Royal Statistical Society, Wiley], pp. 267–288.

Tibshirani, R. et al. (24th Nov. 2010). *Strong rules for discarding predictors in lasso-type problems*. arXiv: 1011.2234 [math, stat].

Waks, A. G. and Winer, E. P. (22nd Jan. 2019). 'Breast Cancer Treatment: A Review'. In: *JAMA* vol. 321, no. 3, pp. 288–300.

Wallden, B. et al. (22nd Aug. 2015). 'Development and verification of the PAM50-based Prosigna breast cancer gene signature assay'. In: *BMC medical genomics* vol. 8, p. 54.

Wolpert, D. H. (1st Jan. 1992). 'Stacked generalization'. In: *Neural Networks* vol. 5, no. 2, pp. 241–259.

Wu, E. et al. (Apr. 2021). 'How medical AI devices are evaluated: limitations and recommendations from an analysis of FDA approvals'. In: *Nature Medicine* vol. 27, no. 4. Number: 4 Publisher: Nature Publishing Group, pp. 582–584.

Yang, W. et al. (2012). 'Genomics of Drug Sensitivity in Cancer (GDSC): a resource for therapeutic biomarker discovery in cancer cells'. In: *Nucleic acids research* vol. 41 (D1). Publisher: Oxford University Press, pp. D955–D961.

Yoon, N. K. et al. (1st Dec. 2010). 'Higher Levels of GATA3 Predict Better Survival in Women with Breast Cancer'. In: *Human pathology* vol. 41, no. 12, pp. 1794–1801.

Yuan, M. and Lin, Y. (2006). 'Model selection and estimation in regression with grouped variables'. In: *Journal of the Royal Statistical Society: Series*

B (Statistical Methodology) vol. 68, no. 1. Publisher: Wiley Online Library, pp. 49–67.

Yue, J. and Mulder, K. M. (1st July 2001). 'Transforming growth factor- signal transduction in epithelial cells'. In: *Pharmacology & Therapeutics* vol. 91, no. 1, pp. 1–34.

Zou, H. (1st Dec. 2006). 'The Adaptive Lasso and Its Oracle Properties'. In: *Journal of the American Statistical Association* vol. 101, no. 476, pp. 1418–1429.

Zou, H. and Hastie, T. (2005). 'Regularization and Variable Selection via the Elastic Net'. In: *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* vol. 67, no. 2. Publisher: [Royal Statistical Society, Wiley], pp. 301–320.