Pre-validation for assessing the added predictive value of high-dimensional molecular data in binary classification

Eva-Marie Christina Endres

2014

Department of Statistics,
Ludwig Maximilian's University Munich

Pre-validation for assessing the added predictive value of high-dimensional molecular data in binary classification

Master Thesis

Eva-Marie Christina Endres

Supervision:

Prof. Dr. Anne-Laure Boulesteix,
Department of Medical Informatics, Biometry and Epidemiology

Dr. Riccardo De Bin,
Department of Medical Informatics, Biometry and Epidemiology

May 12th, 2014

Department of Statistics,
Ludwig Maximilian's University Munich

Abstract

Validation is a crucial step for the evaluation of new prediction rules. Within the framework of high-dimensional molecular data, the assessment of the added predictive value is of particular importance. That is, we want to verify whether the performance of a prediction rule can be improved if molecular data is included in addition to the standard clinical predictors. For this purpose we create combined prediction models which contain the clinical predictors and a molecular score, which aggregates the molecular information into a single predictor. A special challenge arises if there is no independent data set at hand, on which the added predictive value can be measured. When comparing the molecular score to the clinical predictors on the same data set that has been used to generate the score, overfitting mechanisms might make the score to seem more important than it actually is. To elude this problem, Tibshirani and Efron's (2002) pre-validation approach can be used. It embeds the score construction into a cross-validation loop which mimics the situation of training and test data to be at hand. Here we investigate and compare the added predictive value of prediction models including molecular scores that have been derived with and without pre-validation, within the scope of binary classification. In general, we use two approaches for score generation: the least absolute shrinkage and selection operator and a supervised principal component analysis. The investigation of the added predictive value in six different simulation studies and in a real breast cancer data set allows a comparison of molecular scores obtained with or without pre-validation.

Notations

General conventions:
Small or capital letters, such as n or G denote scalars, small bold letters, such as $\mathbf{y}$ denote vectors, and capital bold letters, such as $\mathbf{X}$ denote matrices. Estimates are marked by the circumflex accent $\hat{}$.

n	Number of observations
$\mathbf{Z} \in \mathbb{R}^{n \times q}$	Matrix of clinical predictors
$\mathbf{X} \in \mathbb{R}^{n \times p}$	Matrix of molecular predictors
$\mathbf{y} \in \mathbb{R}^{n \times 1}$	Vector of binary response
Σ	Covariance matrix
$\mathbf{R}$	Correlation matrix
ρ	Correlation coefficient
$L(\cdot)/\ell(\cdot)$	Likelihood function/log-likelihood function
$\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^\top$	Coefficients of the molecular predictors in a regression model
$\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_q)^\top$	Coefficients of the clinical predictors in a regression model
η_i	Linear predictor for the i-th observation
$\mathbf{x}_{score} = (x_{score_1}, \ldots, x_{score_n})^\top$	Non pre-validated molecular score
$\widetilde{\mathbf{x}}_{score} = (\widetilde{x}_{score_1}, \ldots, \widetilde{x}_{score_n})^\top$	Pre-validated molecular score
$o(g)$	Set of indices of all observations in group g
$f(\cdot)$	Rule for generating the omics score / Classification rule
λ/t	Penalty/tuning parameter for *Lasso* regression
$\Phi \in \mathbb{R}^{r \times n}$	Matrix of principal components, where $r = \min(n, p)$

Contents

1 Introduction

With the aid of microarray technology which was developed in the 1990s (see, for instance, Peterson, 2013, p. 1), it became possible to examine thousands of genes (more precisely gene expressions) simultaneously. These microarray gene expression data have been used for disease outcome prediction or diagnosis purposes (Boulesteix and Sauerbrei, 2011).
In the last years, an attempt has been made to improve diagnosis and prognosis of disease outcomes by the use of gene expressions. It means that gene expressions have been used to upgrade rather than to substitute standard clinical predictors for a disease outcome (De Bin, Herold and Boulesteix, 2014). While microarray technology has brought certain advantages, it has also brought new challenges.

Firstly, there is the so-called $n << p$ *problem*. Because gene expression data is difficult and expensive to collect, the number of patients that are examined is generally small (approximately $100 - 200$). In contrast, however, it is possible to measure thousands of gene expression values simultaneously with one microarray. This means that we have a lot of potential molecular predictors but only few observations. As a result, standard statistical methods are no longer applicable (cf. Boulesteix, Strobl, Augustin and Daumer, 2008). The second challenge is to determine if the omics data (high-throughput molecular data) has any additional predictive ability compared to standard clinical information.

Before the development of microarrays well-established, easy, and cheap to collect clinical information like patient age, tumor grade or hormone levels have, for example, been used for predicting the probability of cancer relapse. The question is now whether the performance of a prediction model is improved if gene expression values are included in addition to the standard clinical covariates. In other words, one needs to verify if the inclusion of gene expressions in a prediction model composed of clinical predictors, is able to improve its predictive ability (De Bin, Herold and Boulesteix, 2014).

To answer the question of the added predictive value of the omics data it is not enough to build either one classifier based on all predictors, without distinguishing between microarray and clinical predictors, or to build two classifiers: one based on clinical parameters, one based on microarray data (Boulesteix, Porzelius and

Daumer, 2008). In the former case, good clinical predictors may get lost in the huge amount of microarray predictors, while in the latter case one does not know whether microarray data do exactly the same as the clinical predictors if both classifier have similar predictive power (Boulesteix, Porzelius and Daumer, 2008). Thus, for assessing the added predictive value of molecular data, it is necessary to construct more complex classifiers that include both the clinical and the molecular predictors.

For the combination of low-dimensional clinical data and high-dimensional omics data, it is common practice to aggregate the molecular data into a single molecular score. For this purpose, we can apply suitable statistical or machine learning techniques that are able to handle the high-dimensionality of the omics data. The derived omics score is then used together with the clinical predictors, as independent covariate in a multivariate regression model. Subsequently, one can validate the added predictive value of the omics score by different strategies.
However, a problem arises if we do not have an independent validation data set at hand.

When comparing the omics score to the standard clinical predictors on the same data set that was used to derive the score, the results may strongly be biased in favor of the microarray predictor (Tibshirani and Efron, 2002). Although overfitting is a commonly recognized problem in microarray analysis, it continually happens that overoptimistic conclusions are drawn during the assessment of the added predictive value of high-dimensional molecular data (Boulesteix and Sauerbrei, 2011). It means that if this problem is ignored, the omics score will seem to be much more relevant than it actually is because it considerably overfits the data at hand (Boulesteix and Sauerbrei, 2011).
To elude this problem, Tibshirani and Efron (2002) suggest to use their pre-validation approach that mimics the situation of both training and test data to be at hand by embedding the construction of the molecular score into a kind of cross-validation loop. This will create a 'fairer' version of the omics score that in turn allows a fairer comparison to the standard clinical predictors (Tibshirani and Efron, 2002).

In the present thesis Tibshirani and Efron's pre-validation approach is to be extended to the usage of the *least absolute shrinkage and selection operator* (*Lasso*) as well as for *supervised principal component analysis* (*superPC*) for generating the omics score.
Both approaches, the *Lasso* and the *superPC* analysis are introduced and utilized for building omics scores, whereby each a pre-validated and a non pre-validated

molecular score can be created. Afterwards the added predictive value of each molecular score is assessed on the same data set which has been used to generate the score. For the validation of the added predictive value, we will focus on two approaches: testing the molecular score in a multivariate regression model ans evaluating the predictive accuracy of the clinical and the combined model.
The main goal of this thesis is to determine whether the pre-validated omics score can overcome overfitting issues compared to its non pre-validated counterpart. This comparison is based on the values of the regression coefficients and their associated p-values.
If overfitting can be avoided by pre-validating the omics score, we would expect that the regression coefficient of a pre-validated score is smaller than its non pre-validated equivalent. Besides that, the regression coefficient of a pre-validated molecular score is expected to be less significant than the regression coefficient of a non pre-validated score. We would also expect that the predictive accuracy of the combined model including the pre-validated molecular score is smaller than the predictive accuracy of the model including the non pre-validated score.

In **Chapter 2** both the biological and the statistical background of this thesis is clarified. Logistic regression models are introduced for binary classification. Strategies for the combinations of low-dimensional clinical predictors and high-dimensional molecular predictors are represented and the problem of overfitting and its consequences is described.
Chapter 3 elucidates the fundamental idea of Tibshirani and Efron's pre-validation approach. Furthermore, the *least absolute shrinkage and selection operator* and *supervised principal component analysis* are characterized and applied for generating pre-validated and non pre-validated omics scores. The implementation ensues for the statistical software `R` (version 3.0.2).
A particular description of how the added predictive value of the molecular score can be assessed on the same data set that has been used to build the score, can be found in **Chapter 4**.
Chapter 5 contains specifications about the data simulation process and the real breast cancer data set that are used for practical applications. Also the results of the assessment of the added predictive value of pre-validated and non pre-validated omics scores on both data sets are described in this chapter.
A summary of the thesis follows in **Chapter 6**.

2 Background

2.1 Biological foundations and the microarray technology

Biological foundations

The human genome is estimated to consist of about 20,500 genes (National Human Genome Research Institute, 2012). All of these genes are located on the 23 chromosome pairs, and therefore, part of the deoxyribonucleic acid (DNA). Genes control the production of amino acids which in turn are combined to proteins. These proteins form the building blocks for structures within the cells and ultimately the whole body (Mandal, 2014). The activity of a gene i.e., how often it is transcribed and translated for the production of amino acids, is called gene expression. Generally all cells of an individual are genetically homogeneous but structurally and functionally heterogeneous owing to the differential expression of genes (Jaenisch and Bird, 2003). The expression of a gene is regulated in every cell by a wide range of mechanisms and determines the phenotype (e.g. a disease outcome) of an individual (Wikipedia, 2014).
The aim of gene expression analysis is to reveal differentially expressed genes due to the fact that the gene expression levels can give some indication about the presence or the future development of a disease. Hence, the analysis of gene expression may serve for diagnosis or prediction of a disease outcome (Lottaz et al., 2008). As a result, the analysis of gene expressions can, for example, be used to study regulatory gene defects in cancer and other devastating diseases (National Center for Biotechnology Information, 2014).

Microarray technology

The way from a gene to the phenotype leads through transcription and translation:

$$\text{DNA} \xrightarrow{\text{transcription}} \text{mRNA} \xrightarrow{\text{translation}} \text{amino acid.}$$

The messenger ribonucleic acid (mRNA) is a reverse copy of the DNA and contains the genetic information for amino acid production (Nguyen et al., 2002). This means that by quantifiying the relative amount of mRNA in a cell, one can draw

conclusions about the amino acids/proteins, and consequently about the phenotype of this cell (see Duggan et al., 1999).

Referring to Science Creative Quarterly (2014), the principle of microarray technology can be described as follows. So-called oligonucleotide arrays are based on small base pair gene fragments (probes) which are complementary to (segments of) specific genes. These probes are selected to have little cross-reactivity with other genes. To cope with the problem of non-specific hybridization, a second probe identical to the first except for a mismatched base, is placed next to the first.
The messenger RNA (mRNA) extracted from a cell is used to prepare cRNA by reverse transcription and further transcription. Fragments of the cRNA bind to their complementary probes on the microarray. By combining the perfect match and the mismatch probes, a single expression value can be derived for a specific gene. Via photolithography and chemical synthesis, the microarray can be manufactured which in turn gives indication about the involved genes (for detailed explanation, see, for instance, Dalma-Weiszhausz et al., 2006). Science Creative Quarterly (2014), for instance, shows a schematic illustration of the measurement of gene expressions by oligonucleotide microarrays.

2.2 Binary classification

One important objective of statistical analysis in the medical sector, is the prediction or diagnosis of a disease outcome. The dependent variable which displays a disease-related outcome, can be of different types. Typically it is categorical or a survival outcome. In the present thesis we will emphasize the situation of a binary outcome.
A very popular field of binary classification based on high-dimensional molecular data is cancer research. Common outcomes in cancer research might, for example, be the presence of a certain tumor type or the prediction of a cancer recidive (Boulesteix, Strobl, Augustin and Daumer, 2008). So as one can see, classification can be divided into two main challenges: diagnosis and prognosis. Both problems are treated identically from a statistical point of view (Boulesteix, Strobl, Augustin and Daumer, 2008).
Generally speaking, classification addresses the ability that a classifier can learn information from the features of object, and then make an accurate prediction to assign objects to their true class (Peterson, 2013, p. 4). In accord with Slawski et al. (2008) (p. 3) the binary classification problem can be framed as follows: Let us consider a predictor space $\mathcal{X} \subseteq \mathbb{R}^p$ and a set of class labels $\mathcal{Y}$, where $\mathcal{Y} = \{0, 1\}$ in the case of a binary outcome. A prediction rule f is then constructed on the basis of n realizations $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ of the vector $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$ of random

variables:

$$f: \mathcal{X} \to \mathcal{Y}$$
$$f(\mathbf{x}) \to y,$$

where $\mathbf{x_i} = (x_{i1}, \ldots, x_{ip})^\top$. In case of binary classification the prediction rule assigns the probability for class $y = 1$ to each new observation (cf. Boulesteix and Sauerbrei, 2011).

Logistic regression

In the special case of binary classification i.e., for a dichotomous response $y \in \{0, 1\}$, logistic regression models are commonly used along with linear discriminant analysis. In the present thesis, we will focus on logistic regression models.

The objective of logistic regression is the estimation of the influence of the covariates on the (conditional) probability $P(y_i = 1|\mathbf{x}_i)$. Since it has to be guaranteed that the estimated probabilities lie in the interval $[0, 1]$, we combine the conditional probabilities for $y_i = 1$ with the covariates via a logistic link function i.e.,

$$\log\left(\frac{P(y_i = 1|\mathbf{x}_i)}{1 - P(y_i = 1|\mathbf{x}_i)}\right) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip}.$$

With election of the logistic response function we obtain an equivalent model equation

$$P(y_i = 1|\mathbf{x}_i) = \frac{\exp(\eta_i)}{1 + \exp(\eta_i)},$$

where $\eta_i = \mathbf{x}_i^\top \boldsymbol{\beta} = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip}$ is the linear predictor, $\boldsymbol{\beta}$ the vector of regression coefficients and p the number of independent predictors.

The probability of response class $y = 1$ for a new observation $\mathbf{x}_{new} = (x_{new_1}, \ldots, x_{new_p})^\top$ can be predicted from

$$\hat{P}(y_{new} = 1|\mathbf{x}_{new}) = \frac{\exp(\hat{\beta}_0 + \hat{\beta}_1 x_{new_1} + \ldots + \hat{\beta}_p x_{new_p})}{1 + \exp(\hat{\beta}_0 + \hat{\beta}_1 x_{new_1} + \ldots + \hat{\beta}_p x_{new_p})},$$

where $\hat{\boldsymbol{\beta}}$ is usually estimated via maximizing the log-likelihood function

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left\{ y_i(\mathbf{x}_i^\top \boldsymbol{\beta}) - \log(1 + \exp(\mathbf{x}_i^\top \boldsymbol{\beta})) \right\}.^{1} \tag{2.1}$$

In the following, the linearity of the predictor η as well as the presence of only main effects in the regression model is assumed.
Since the emphasis is on classification based on both clinical and molecular predictors, we need to construct a regression model that handles both types of predictors.

Let us suppose that the conditional probability of $y_i = 1$ may be modeled via a linear combination of the available predictors. A logistic regression model might then have the form

$$P(y_i = 1|\mathbf{x}_i, \mathbf{z}_i) = \frac{\exp(\gamma_0 + \gamma_1 z_{i1} + \ldots + \gamma_q z_{iq} + \beta_1 x_{i1} + \ldots + \beta_p x_{ip})}{1 + \exp(\gamma_0 + \gamma_1 z_{i1} + \ldots + \gamma_q z_{iq} + \beta_1 x_{i1} + \ldots + \beta_p x_{ip})},$$

where from now on $\mathbf{z}_i = (z_{i1}, \ldots, z_{iq})^\top$ denote the clinical and $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^\top$ the molecular predictors for the i-th observation. The parameters $\gamma_1, \ldots, \gamma_q$ and $\beta_1, \ldots, \beta_p$ are the corresponding regression coefficients, while γ_0 characterizes the intercept term.
However, this regression model raises two particular problems.

Arising problems of regression with different types of predictors

Firstly, there is the high-dimensionality of the molecular data matrix $\mathbf{X}$, which is challenging even in the absence of clinical predictors (De Bin, Sauerbrei and Boulesteix, 2014). As mentioned above thousands of gene expression levels can be measured with one DNA-microarray. But since the manufacturing of microarrays is expensive, there are usually only less observations (Lai et al., 2006). This circumstance is often called the $n << p$ *problem* where n is the number of observations and p the number of predictors. Due to the $n << p$ problem standard statistical prediction methods are inapplicable (Boulesteix, Strobl, Augustin and Daumer, 2008). It means that the regression coefficients cannot be simply estimated as usual by maximization of the likelihood (De Bin, Sauerbrei and Boulesteix, 2014). The reason for this is that in the $n << p$ case $\mathbf{X}^\top\mathbf{X}$ has not full rank and is thus not invertible. This issue can be handled via either variable selection, dimension reduction or regularization techniques (De Bin, Sauerbrei and Boulesteix, 2014). Section 3 describes two appropriate strategies for handling high-dimensional data (the *least absolute shrinkage and selection operator* and *supervised principal component analysis*) in detail.
And secondly, we need to find an adequate strategy for the combination of predictors with different characteristics and dimensions, which is not straightforward (De Bin, Sauerbrei and Boulesteix, 2014).
Boulesteix and Sauerbrei (2011) (pp. 218) delineate five strategies that will be

[1] See Appendix A for the derivation.

briefly introduced below.

2.3 Combination strategies for predictors with different dimensionalities

Strategy 1: naive
In the first strategy the clinical and molecular predictors are treated in the exact same manner. That is the simplest way for combination. Variable selection, dimension reduction or regularization is applied to all of the available predictors. That is, the clinical predictors are considered as X variables (De Bin, Sauerbrei and Boulesteix, 2014). A benefit of this strategy is that one can use every prediction method which is convenient for high-dimensional data. However, the risk exists that the few clinical predictors (which are generally on average more predictive than omics predictors) may mostly be disregarded compared to the huge amount of molecular information (De Bin, Sauerbrei and Boulesteix, 2014).

Strategy 2: residuals
The basis of the second strategy is the derivation of a fix clinical prediction model (e.g. logistic regression in case of binary outcome). The resulting linear predictor is then used as an offset and updated by the molecular predictors. It should be noted that the clinical predictors may be subject to selection bias if variable selection has been executed. For further modifications of this strategy see Boulesteix and Sauerbrei (2011) (pp. 218).

Strategy 3: favoring
In this version clinical and molecular predictors are treated simultaneously in a prediction model. The distinct to Strategy 1 is that the clinical predictors are favored for the reason that they are approved predictors for the interesting outcome. Thus, the information content of the clinical predictors is more taken into account. Nevertheless, the influence of clinical predictors in the prediction model is affected by molecular predictors (Boulesteix and Sauerbrei, 2011).

Strategy 4: dimension reduction
This way of creating a combined prediction model is composed of two stages: The molecular predictors are first aggregated to one new component (hereinafter referred to as omics score) by the use of a dimension reduction technique. After this step the new molecular score and the clinical predictors are simultaneously used as independent covariates in a multivariate prediction model.

Strategy 5: replacement
Within the last strategy the clinical information is represented by a clinical index/a clinical score. If one of the components which build the clinical score has low relative importance it might be replaced by more objective molecular information.

In the present thesis we will focus on *Strategy 4* and aggregate the molecular data to a single omics score. Referring to Boulesteix and Sauerbrei (2011), it can be drawn as follows:

$$\mathbf{x}_{score} = \omega_1 \cdot \mathbf{x}_1 + \omega_2 \cdot \mathbf{x}_2 + \omega_3 \cdot \mathbf{x}_3 + \ldots \omega_p \cdot \mathbf{x}_p, \tag{2.2}$$

where the phrases $\mathbf{x}_1, \ldots, \mathbf{x}_p$ and $\omega_1, \ldots, \omega_p$ stand for the gene expression levels and their weights, respectively.
Since not all gene expressions are necessarily connected to the outcome of interest, it is definitely possible that some of the weights equal zero. If so, these gene expressions are not incorporated to the omics score.

As mentioned above, several possibilities are available for the construction of the omics score. In the following we will emphasize on two popular techniques: the regularization approach *least absolute shrinkage and selection operator* (*Lasso*) and a *supervised principal component analysis* (*SPC*) which is a combination of univariate variable selection and subsequent principal component analysis for dimension reduction. A detailed description of how these two approaches can be used for generating an omics score can be found in Sections 3.2.2 and 3.3.3.

After the omics score has been built, it is in a way considered as a 'new predictor' (Boulesteix and Sauerbrei, 2011). Subsequently, this molecular score as well as the clinical predictors will be used as independent covariates in a multivariate logistic regression model to appreciate their relation to the outcome of interest:

$$P(y_i = 1 | \mathbf{x}_i, \mathbf{z}_i) = \frac{\exp(\eta_i)}{1 + \exp(\eta_i)}, \tag{2.3}$$

where

$$\eta_i = \gamma_0 + \gamma_1 \cdot z_{i1} + \gamma_2 \cdot z_{i2} + \ldots + \gamma_q \cdot z_{iq} + \beta_{score} \cdot x_{score,i}$$

denotes the linear predictor. The application of this regression model enables us to compare the predictive power of the omics score to the predictive power of the standard clinical covariates $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_q)^\top$ in predicting the outcome $\mathbf{y}$.
Thus, we can verify if the inclusion of the molecular score in the prediction model yields to an improvement of its prediction ability (De Bin, Herold and Boulesteix,

2014). If so, the molecular score provides an added predictive value compared to the clinical predictors.
However, due to the problem of overfitting, the assessment of the added predictive value should be performed on independent validation data, anyway. This leads us to a substantial issue: How can we evaluate the added predictive value of the molecular score if there is no independent validation data set available?

2.4 Overfitting and its consequences

Especially in the $n << p$ setting, overfitting represents a general problem. As a result of high dimension, it is almost always possible to find a combination of molecular predictors that are associated with the outcome in the considered data set, independently of the predictive power (Boulesteix and Sauerbrei, 2011). Since the molecular score was derived by 'fishing' for relevant predictors, it is likely that the score is strongly correlated with the outcome even in the case of non-informative molecular predictors (Boulesteix and Sauerbrei, 2011).
Aside from that, overfitting arises since the outcome $\mathbf{y}$ has already been used in the construction of the molecular score (Tibshirani and Efron, 2002).
Consequently, the assessment of the added predictive value of the omics score on the same data which has been used to generate the score, is strongly biased in favor of the microarray predictor (Tibshirani and Efron, 2002). The molecular score will seem to be much more important than it realistically is the case.
To avoid the problem of overfitting and to create a 'fairer' version of the omics score, Tibshirani and Efron (2002) suggest to use their pre-validation approach. Pre-validation is supposed to ensure an unbiased comparison of the different predictors on the same data set on which the molecular score has been built. It should be applied to the dimension reduction step during the score generation (Boulesteix and Sauerbrei, 2011). In the following section the process of pre-validation will be described in detail.

3 Pre-validation

3.1 Fundamental idea

The primary reason for the usage of pre-validation is the creation of an omics score that acts if it hasn't seen the response $\mathbf{y}$ (Tibshirani and Efron, 2002). The practical realization is effected through a kind of cross-validation. The pre-validated omics score is then used as independent covariate in a multivariate regression model which is adjusted for the clinical predictors, to measure its influence on the response. This will allow us to verify if the molecular data i.e., the omics score adds any predictive power to the standard clinical predictors.
Referring to Tibshirani and Efron (2002) the process of pre-validation can be formulated as follows:

1. Divide the present observations into G (approximately) equal-sized groups.
2. Set aside one group g. Use the gene expression levels of the remaining observations for the derivation of the (linear) molecular score.
3. Apply the rule for generating the molecular score on the left-out observations of group g which yields the pre-validated molecular score.
4. Repeat steps 2-3 for each group $g = 1, \ldots, G$.
5. Fit a logistic regression model using both the pre-validated omics score and the q clinical predictors as independent covariates (cf. expression (2.3)).

Since steps 1-4 are used for the generation of the molecular predictor, the molecular data is exclusively required. Figure 3.1 gives a schematic illustration of this procedure.

As mentioned above, special nature of this pre-validation approach is that it creates a molecular score without the direct use of the response $\mathbf{y}$. This implies that the predictor for observation i has not seen the true class label for observation i and is thus, not biased in favor of the molecular data (Tibshirani and Efron, 2002). The procedure of pre-validation tries to mimic the situation of both a learning and a validation data set to be at hand. Usually, we would use a learning data set

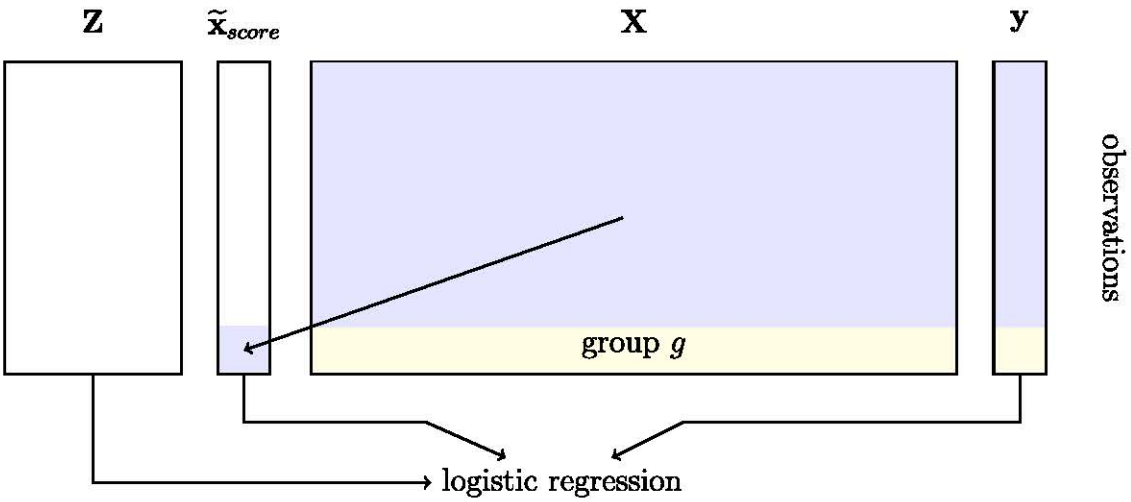

Figure 3.1: Schematic illustration of the pre-validation process referring to Tibshirani and Efron (2002).

to obtain the rules for generating the omics score. Afterwards, we would apply it to the validation data to asses its predictive ability while predicting $\mathbf{y}_{valid}$ from $\mathbf{X}_{valid}$ and $\mathbf{Z}_{valid}$. In this way, a fair comparison of the predictive power of both the molecular score and the clinical predictors is ensured.

In a formal way pre-validation can be expressed as follows:

$$\widetilde{\mathbf{x}}_{score}^{[o(g)]} = \hat{f}_{\mathbf{X}^{[-o(g)]},\mathbf{y}^{[-o(g)]}}(\mathbf{X}^{[o(g)]}),\ g = 1, \ldots, G. \tag{3.1}$$

The term $o(g)$ denotes the set of indices which represent the observations included in group g. Vice versa, $-o(g)$ stands for the indices of all observations not included in group g.
More precisely, the values of the omics score for observations in group g i.e., $\widetilde{\mathbf{x}}_{score}^{[o(g)]}$ are generated by applying the rule for score generation $\hat{f}_{\mathbf{X}^{[-o(g)]},\mathbf{y}^{[-o(g)]}}$ (which has been deduced from the observations $\mathbf{X}^{[-o(g)]}$ and $\mathbf{y}^{[-o(g)]}$, not included in group g) on the reduced data matrix $\mathbf{X}^{[o(g)]}$ (containing all molecular predictors but only from observations of group g). As mentioned above, we realize the generation of the omics score in two different ways in the present thesis.

On the one hand the *least absolute shrinkage and selection operator* (*Lasso*) and on the other hand a *supervised principal component analysis* (*superPC*) is used. The next sections will describe both approaches and their application within the scope of pre-validation.

3.2 Least absolute shrinkage and selection operator (Lasso)

3.2.1 Motivation and definition

In practice one can see two reasons why ordinary least squares regression yields no adequate models: prediction accuracy and interpretation (cf. Tibshirani, 1996). The former indicates that the ordinary least squares estimator often has a high variance which may affect the overall prediction accuracy, while the latter refers to the amount of potential predictors (cf. Tibshirani, 1996). The higher the number of predictors, the more difficult is the interpretation. To avoid these two problems one can use the *least absolute shrinkage and selection operator* (*Lasso*).

By the fact that the Lasso technique shrinks some coefficients and sets others to 0 a more precise prediction and a better interpretability of the resulting regression model can be ensured (Tibshirani, 1996). Furthermore, Lasso regression entails the great advantage that it can in addition handle the consequences of the $n << p$ *problem*.
Due to the situation of having more independent predictors than observations, the design matrix $\mathbf{X}$ has not full rank which leads to the issue that $\mathbf{X}^\top\mathbf{X}$ is not invertible. The Lasso provides a combination of good prediction accuracy and an intrinsic variable selection coupled with computational feasibility (Bühlmann and van de Geer, 2011, p. 7).
Since the results of Lasso regression are dependent on scaling, we hereinafter assume to have a standardized data matrix $\mathbf{X}$. The following elucidations are mainly based on Tibshirani (1996).

The Lasso

The Lasso is a so-called regularization approach which means that very small as well as very large regression coefficients are penalized. Within the scope of Lasso regression the ℓ_1-penalty term is used.
The regression parameters are estimated via

$$\hat{\boldsymbol{\beta}}_{Lasso} = \arg\min_{\boldsymbol{\beta}} \left\{ (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^\top (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + \lambda \cdot ||\boldsymbol{\beta}||_1 \right\},$$

where $(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^\top (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i^\top \boldsymbol{\beta}\right)^2$ denotes the residual square sum, $\lambda \geq 0$ the penalization parameter, and $||\boldsymbol{\beta}||_1 = \sum_{j=1}^{p} |\beta_j|$ the ℓ_1-penalty for the

restriction of the absolute regression coefficients (see, for instance, Fahrmeir et al., 2013, p. 208).
As well as the least squares estimator, the *Lasso* estimator minimizes the residual square sum, but under the restriction $\sum_{j=1}^{p} |\beta_j| \leq t$. The tuning parameter $t \geq 0$ controls the amount of shrinkage in this expression.

The Lasso for binary response

A great advantage of the ℓ_1-penalization is that it can be used with any linear regression model, which means that it is also suitable for a logistic regression model with binary response (Hastie et al., 2009, p. 125). For Lasso regression (ℓ_1-penalty), a penalized version of the log-likelihood function (cf. expression (2.1)) is to be maximized (Hastie et al., 2009, p. 125):

$$\ell_{penalized}(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left[y_i(\mathbf{x}_i^\top \boldsymbol{\beta}) - \log(1 + \exp(\mathbf{x}_i^\top \boldsymbol{\beta})) \right] - \lambda ||\boldsymbol{\beta}||_1. \tag{3.2}$$

Geometric properties of the Lasso

The specific about the Lasso is that some of the estimated regression parameters may exactly be zero, which implies a variable selection. The reason for this is the ℓ_1-geometry which is based on the ℓ_1-norm (Bühlmann and van de Geer, 2011, p. 9). For a graphical description of the ℓ_1-geometry let us assume $p = 2$. Because of the quadratic form of the parameters which results from solving the least squares criterion, the contour lines of these parameter values are ellipses with the specific shape determined by $\mathbf{X}^\top\mathbf{X}$, and center at the least squares estimator (Fahrmeir et al., 2013, p. 211). For $p = 2$ the ℓ_1-geometry defines diamond-shaped contour lines (Fahrmeir et al., 2013, p. 213). See figure 3.2 for a graphical illustration.
The Lasso regression estimators arise as the points of intersections between the ℓ_1-penalty and the ellipses, based on the least squares criterion. If the contact point is located in one of the corners of the diamond, some of the coefficients will be estimated to be zero (Fahrmeir et al., 2013, p. 213).

The tuning parameter t

The estimations of the Lasso coefficients and especially the number of selected predictors are dependent on the hyperparameter t or λ (see, for instance, Tibshirani, 1996). The hyperparameter controls the strength of penalization.
$\lambda = 0$ leads to the non-penalized least squares estimator (in the case of its existence), whereas the regression parameters shrink with increasing λ. Vice versa, for

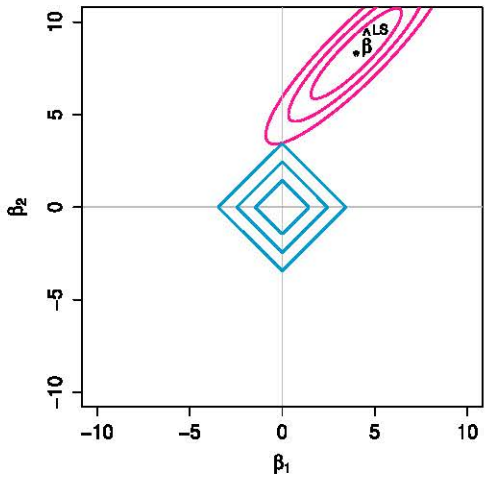

Figure 3.2: Graphical illustration of the least shrinkage and selection operator (cf., for instance, Tibshirani, 1996).

$t = 0$ all estimators equal zero and the penalization decreases with increasing t. The choice of an appropriate penalty parameter is crucial, since it influences the whole analysis. Tibshirani (1996) describe three methods for the estimation of an optimal tuning parameter t: cross-validation, generalized cross-validation and an analytical unbiased estimate of risk. The first two methods are appropriate in the case where the observations $(\mathbf{X}, \mathbf{y})$ are drawn from some unknown distribution (Tibshirani, 1996). The third analytical estimate applies to the $\mathbf{X}$-fixed case (Tibshirani, 1996).
It is prohibited to perform the parameter tuning a posteriori and to just report the best results (see Slawski et al., 2008). In the present thesis the tuning parameter t is chosen via a 5-fold cross-validation. The corresponding criterion is the error rate which has to be minimized. See Appendix B for a detailed description of the parameter tuning process.

3.2.2 Derivation of the omics score using the Lasso

The regression coefficients $\hat{\beta}_{Lasso,1}, \ldots, \hat{\beta}_{Lasso,p}$ (hereinafter, for generating the omics score, no intercept term $\hat{\beta}_{Lasso,0}$ is used) of the p available gene expressions, which has been estimated via the *Lasso*, can be used as weights $\omega_1, \ldots, \omega_p$ to compute the linear molecular score (cf. expression 2.2).
Let again $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)^\top$ denote the gene expression levels of n observations,

and $\hat{\boldsymbol{\beta}}_{Lasso} = \left(\hat{\beta}_{Lasso,1}, \ldots, \hat{\beta}_{Lasso,p}\right)^\top$ their corresponding estimated *Lasso* coefficients, derived by the regression model

$$P(y_i = 1|\mathbf{x}_i) = \frac{\exp(\hat{\beta}_{Lasso,1} \cdot x_{i1} + \ldots + \hat{\beta}_{Lasso,p} \cdot x_{ip})}{1 + \exp(\hat{\beta}_{Lasso,1} \cdot x_{i1} + \ldots + \hat{\beta}_{Lasso,p} \cdot x_{ip})}.$$

For an observation i the omics score can be generated as

$$x_{score,i} = \hat{\beta}_{Lasso,1} \cdot x_{i1} + \hat{\beta}_{Lasso,2} \cdot x_{i2} + \ldots + \hat{\beta}_{Lasso,p} \cdot x_{ip}. \tag{3.3}$$

Since our primarily objective is the comparison of non pre-validated and pre-validated molecular scores, we will also adapt the pre-validation approach for the usage of the *Lasso*.

3.2.3 Pre-validation adapted for the Lasso

For this purpose, the estimation of the *Lasso* coefficients is included into the pre-validation loop:

1. Divide the available observations into G approximately equal-sized groups.
2. Leave group g out and perform a *Lasso* regression on the remaining observations to derive the vector $\hat{\boldsymbol{\beta}}_{Lasso}^{[-o(g)]}$ including regression coefficients for every molecular predictor.
3. Compute the linear molecular score for person $i \in o(g)$ as weighted sum over all molecular predictors with $\hat{\boldsymbol{\beta}}_{Lasso}^{[-o(g)]}$ used as weights.

$$\widetilde{x}_{score,i} = \hat{\beta}_{Lasso,1}^{[-o(g)]} \cdot x_{i1}^{[o(g)]} + \ldots + \hat{\beta}_{Lasso,p}^{[-o(g)]} \cdot x_{ip}^{[o(g)]}.$$

4. Repeat steps 2-3 for every group $g = 1, \ldots, G$.

In comparison with Equation (3.1), in step 3 of this routine the omics scores for observations of group g are computed by applying the *Lasso* regression coefficients $\hat{\boldsymbol{\beta}}_{Lasso}^{[-o(g)]}$ (derived on observations not in group g) on the reduced data $\mathbf{X}^{[o(g)]}$.

3.2.4 Implementation in R

The generation of the omics score using the *Lasso* has been implemented in R through two functions: `lasso.with.prevalidation(·)` and `lasso.without.prevalidation(·)`. The former computes the pre-validated omics score, while the

latter computes the non pre-validated version of the molecular score.
The input consists of the binary response vector $\mathbf{y}$, the matrix of gene expression values $\mathbf{X}$, and the term `norm.fraction` which corresponds to the tuning parameter t (smaller values yield higher penalization). As default, t is chosen via a 5-fold cross-validation. If the pre-validated molecular score should be computed the number of pre-validation folds has to be additionally inputted.
The Bioconductor package `CMA` developed by Slawski et al. (2008) forms the basis of both functions. The package allocates the function `LassoCMA(·)` for estimating ℓ_1-penalized regression models for binary outcomes. Furthermore, the function `tune(·)` is used to chose the optimal value for the tuning parameter t via a cross-validation.
Both functions, `lasso.with.prevalidation(·)` and `lasso.without.prevalidation(·)`, output the estimated *Lasso* coefficients for every molecular predictor. In case of pre-validation, a list containing the vectors of the Lasso estimators for every pre-validation fold is outputted. If the omics score has been obtained without pre-validation, the output is the single vector of the Lasso regression coefficients. Subsequently, the function `score(·)` has to be invoked. After the deliveration of an outcome object obtained from on of the two functions `lasso.with.prevalidation(·)` or `lasso.without.prevalidation(·)`, it computes the score values for every observation.

Among regularization approaches, dimension reduction techniques are frequently used for classification in high-dimensional settings. In the present thesis also a supervised principal component analysis is applied to derive the molecular score. Details of this procedure are given in the next section.

3.3 Supervised principal component analysis

3.3.1 Principal component analysis

Generally speaking, principal components are a sequence of the data, mutually uncorrelated and ordered in variance (Hastie et al., 2009, p. 534). With the aid of principal component analysis, the latent structure of a data set can be revealed i.e., genes with similar component loadings can be identified to construct groups of genes with similar expression profiles (Peterson, 2013, p. 161).
The main goal of principal component analysis is to represent the data in terms of a smaller number of variables which already comprise a large amount of the whole variability (Nikulin and McLachlan, 2010, p. 82). These new variables i.e., the principal components can then be used as covariates in a regression model, instead of the original variables. This special form of regression is called *principal*

component regression (see, for instance, Fahrmeir et al., 2013, p. 159).
The principal components, which are uncorrelated linear combinations of the original variables, capture the largest proportion of the variance in the original data in a minimal number of dimensions (Nikulin and McLachlan, 2010, p. 82). It means that although a dimension reduction is performed, the loss of information is minimal.
Following Tutz (2013), the statistical background of principal component analysis based on observations will be described below.

Let us assume that $\mathbf{S} = \frac{1}{n-1}\mathbf{X}^\top\mathbf{X} \in \mathbb{R}^{p\times p}$ is the empirical covariance matrix of the (column-) centered data matrix $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)^\top$ with $\mathbf{x}_j^\top = (x_{1j}, \ldots, x_{nj})$. For the derivation of the first principal component $\boldsymbol{\phi_1} \in \mathbb{R}^{1\times n}$, we need to find a vector $\boldsymbol{\alpha}_1 = (\alpha_{11}, \ldots, \alpha_{1p})^\top \in \mathbb{R}^{p\times 1}$ which maximizes the variance of the linear combination

$$\boldsymbol{\phi_1} = \boldsymbol{\alpha}_1^\top\mathbf{X} = \alpha_{11}\mathbf{x}_1 + \ldots + \alpha_{1p}\mathbf{x}_p.$$

It means that

$$\begin{aligned} Var(\boldsymbol{\phi}_1) &= Var(\boldsymbol{\alpha}_1^\top\mathbf{X}) \\ &= \boldsymbol{\alpha}_1^\top\mathbf{S}\boldsymbol{\alpha}_1 \\ &= \frac{1}{n-1}\left[\boldsymbol{\alpha}_1^\top\mathbf{X}^\top\mathbf{X}\boldsymbol{\alpha}_1\right] \\ &= \frac{1}{n-1}\left[\boldsymbol{\phi}_1^\top\boldsymbol{\phi}_1\right] \\ &= \frac{1}{n-1}\sum_{i=1}^{n}\phi_{i1}^2 \rightarrow \max_{\boldsymbol{\alpha}_1} \end{aligned}$$

under the constraint $||\boldsymbol{\alpha}_1||^2 = \boldsymbol{\alpha}_1^\top\boldsymbol{\alpha}_1 = 1$. The constraint is necessary since the variance could be increased without limit by increasing the components of $\boldsymbol{\alpha}_1$ (Tutz, 2013). $\boldsymbol{\phi}_1$ then denotes the first principal component which contains the largest variability.
Further (maximal $r = \min(n, p)$) principal components are obtained by looking for weights which maximize the variance under the additional restriction that the weight is orthogonal to the weights of the previous principal components (Tutz, 2013). This restriction implicates that the principal components are independent from each other.
So the challenge is to find vectors $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_r$ such that

$$Var(\boldsymbol{\phi}_j) = Var\left(\boldsymbol{\alpha}_j^\top\mathbf{X}\right) = \boldsymbol{\alpha}_j^\top\mathbf{S}\boldsymbol{\alpha}_j \rightarrow \max_{\boldsymbol{\alpha}_j}$$

subject to $||\boldsymbol{\alpha}_j||^2 = 1$ and $\boldsymbol{\alpha}_j^\top \boldsymbol{\alpha}_s = 0$, $s = 1, \ldots, j-1$. With the use of the Lagrange multiplier λ, this maximization problem can be restated to an eigenvalue problem (see, for instance, Tutz, 2013). So the maximization term is solved by the eigenvectors $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_r$ of $\mathbf{S}$, that correspond to the largest eigenvalues $\lambda_1 \geq \ldots \geq \lambda_r$ (Tutz, 2013). For the derivation of the eigenvectors the spectral decomposition of the covariance matrix is recommended:

$$\mathbf{S} = \mathbf{P}\boldsymbol{\Lambda}\mathbf{P}^\top.$$

The columns of $\mathbf{P} = (\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_r) \in \mathbb{R}^{r \times r}$ characterize the eigenvectors of $\mathbf{S}$, and $\boldsymbol{\Lambda} \in \mathbb{R}^{r \times r}$ is a diagonal matrix containing the corresponding eigenvalues $(\lambda_1, \ldots, \lambda_r)$. So the principal components are represented by $\boldsymbol{\Phi} = \mathbf{P}^\top \mathbf{X} \in \mathbb{R}^{r \times n}$ and thus, uncorrelated linear combinations of the original predictors (Tutz, 2013). The covariance of the principal components is the defined by

$$Cov(\boldsymbol{\Phi}) = Cov\left(\mathbf{P}^\top \mathbf{X}\right) = \mathbf{P}^\top \mathbf{S}\mathbf{P} = \boldsymbol{\Lambda}$$

with $Var(\boldsymbol{\phi}_i) = \lambda_i$ and $Cov(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j) = 0$, $i \neq j$.

3.3.2 Supervised principal component analysis

As one can see from the definitions above, only the data matrix $\mathbf{X}$ is used for the derivation of the principal components. Such procedures are called "unsupervised".
This means that the response $\mathbf{y}$ is not used to build the principal components which leads to the problem that there is no guarantee that the principal components are correlated to the clinical outcome (Bair and Tibshirani, 2004). Another disadvantage of unsupervised principal component analysis is the fact that a combination of all available molecular predictors is used to predict the outcome (Bair and Tibshirani, 2004). Based on the assumption that most of the gene expression values in the available data set are unrelated to the binary outcome, the predictive ability of the deduced classifier is lessened. Thus, methods, which use only a subset of genes, generally perform better (Bair and Tibshirani, 2004).
For that reason Bair and Tibshirani (2004) developed a supervised principal component analysis for survival prediction. After accomplishing some modifications, this approach can also be used for binary classification. It will be described below.

The basic idea of supervised principal component analysis is to use only molecular predictors which are related to the outcome for the generation of the principal components, instead of using all of them (Bair and Tibshirani, 2004). For the identification of subsets of gene expressions which are correlated to the binary

outcome, several methods are available.
In contrast to Bair and Tibshirani (2004) who analyze data with survival outcome, we will not rank the gene expression values on basis of a so-called Cox score but regarding to their p-values from the Wald test.
The hypotheses $H_0 : \beta_j = 0$ versus $H_1 : \beta_j \neq 0$ are tested for every available molecular predictor $\mathbf{x}_j, j = 1, \ldots, p$, in an univariate logistic regression model.
During the univariate variable selection it is possible to adjust for the clinical predictors. Bøvelstad et al. (2009) and Ntzani and Ioannidis (2003) recommend to use the adjusted version of variable selection to ensure that the principal components are associated to the outcome in the multivariate model.
From the top-list of gene expressions (sorted in descending order by their p-values), the first k predictors are used to generate the principal components. For the determination of k, we can, for example, define a threshold for the p-value, or perform a cross-validation to tune this parameter. Following van Wieringen et al. (2009), in this thesis k is not estimated but set fix to $k = 25$. This will guarantee sets of molecular predictors of equal size for every pre-validation fold.
After variable selection was performed, $\mathbf{X}$ is only composed of the k top molecular predictors, not of all available gene expressions. Apart from that, the supervised principal component analysis follows the same principal component analysis scheme as depicted above.

Because the intended aim is the generation of a molecular score, the next section will describe how supervised principal component analysis can be used in this scope.

3.3.3 Derivation of the omics score using supervised principal components

For this purpose an (univariate) variable selection must be performed on the omics data to obtain a top-list of relevant molecular predictors. The first k predictors of this list (i.e. the k predictors with the smallest p-values) create the data matrix $\mathbf{X}$, whose principal components shall be determined. The other omics predictors are not further considered for the principal component analysis.
The obtained principal components can then be used as independent covariates in a principal component regression model. Bair and Tibshirani (2004), for example, only use the estimated first or second principal components for predicting the survival outcome. But they recommend to take a linear combination of several principal components rather than simply taking the first two principal components to improve the predictive power of the model (Bair and Tibshirani, 2004). In the present thesis, the number m of principal components that should be used in the

prediction model, is determined via a 5-fold cross-validation, where the maximal number of principal components is chosen to be 10.
Currently, we have obtained m supervised principal components that are linear combinations of the original molecular predictors and which are chosen for outcome prediction in the regression model

$$P(y_i = 1|\boldsymbol{\phi}_i) = \frac{\exp(\beta_{superPC,1} \cdot \phi_{i1} + \ldots \beta_{superPC,m} \cdot \phi_{im})}{1 + \exp(\beta_{superPC,1} \cdot \phi_{i1} + \ldots \beta_{superPC,m} \cdot \phi_{im})}.$$

This yields a vector of regression coefficients $\hat{\boldsymbol{\beta}}_{superPC} = \left(\hat{\beta}_{superPC,1}, \ldots, \hat{\beta}_{superPC,m}\right)^\top$ for every principal component which are then used as weights for obtaining the molecular score $\tilde{\mathbf{x}}_{score}$. In other words, the omics score for an observation $i = 1, \ldots, n$ is computed as

$$x_{score_i} = \hat{\beta}_{superPC,1} \cdot \phi_{i1} + \ldots + \hat{\beta}_{superPC,m} \cdot \phi_{im}.$$

3.3.4 Pre-validation adapted for supervised principal component analysis

To use supervised principal component analysis in the scope of pre-validation, one can use the following routine:

1. Divide the available observations into G approximately equal-sized groups.
2. Leave group g out and
   a) perform (univariate) variable selection on the remaining observations to obtain a top-list of the molecular predictors.
   b) Perform a principal component analysis on the basis of the first $k = 25$ predictors in the top-list.
   c) Determine the number of principal components m that should be used as predictors via a 5-fold cross-validation.
   d) Use the first m of the derived principal components as independent covariates in a multivariate (logistic) regression model to estimate the vector $\hat{\boldsymbol{\beta}}_{superPC}^{[-o(g)]}$, including the regression coefficients for every principal component.
3. Compute the linear molecular score for person $i \in o(g)$ as weighted sum over the m principal components with $\hat{\boldsymbol{\beta}}_{superPC}^{-o(g)}$ used as weights

$$\widetilde{x}_{score,i} = \hat{\beta}_{superPC,1}^{[-o(g)]} \cdot \phi_{i1}^{[o(g)]} + \ldots + \hat{\beta}_{superPC,m}^{[-o(g)]} \cdot \phi_{im}^{[o(g)]}$$

4. Repeat steps 2-3 for every group $g = 1, \ldots, G$.

3.3.5 Implementation in R

For the computation of the supervised principal component score on basis of the molecular data, two functions have been implemented in R: `superpc.with.prevalidation(·)` and `superpc.without.prevalidation(·)`, whereby the former generated the omics score with pre-validation, and the latter without pre-validation. The number of genes in the top-list (default $k = 25$), the maximum number of principal components to use as predictors (default $\max(m) = 10$) as well as the number of cross-validation folds for the determination of m can be given as arguments.
The variable selection can either be performed while adjusting for the clinical predictors, or not. The first $k = 25$ predictors of the derived top-list are passed to the R-function `prcomp(·)` from the package `stats` which performs an *R-mode* principal component analysis via the singular value decomposition of the correlation matrix of $\mathbf{X}$, since the results of the principal component analysis are scale-dependent.
The m chosen principal components which are determined via a cross-validation within the function `number.of.pcs.cv(·)`, are then used as independent covariates in a multivariate logistic regression model. To deal with the problem of separation which leads to infinite estimates and standard errors, we use the function `brglm(·)` from the R-package of the same name. It fits a generalized linear model using Firth's (1993) modified score procedure. That is, the maximum likelihood estimate $\hat{\beta}_j$ is not solution to the score function

$$\frac{\partial \ell(\boldsymbol{\beta})}{\partial \beta_j} = \sum_{i=1}^{n} (y_i - \pi_i) x_{ij} \stackrel{!}{=} 0$$

but to the modified score function

$$\sum_{i=1}^{n} \left(y_i - \pi_i + h_i (\frac{1}{2} - \pi_i) \right) x_{ij} \quad (j = 1, \ldots, p),$$

where h_i denotes the i-th diagonal element of the *hat matrix*

$$\mathbf{H} = \mathbf{W}^{1/2} \mathbf{X} \left(\mathbf{X}^\top \mathbf{W} \mathbf{X} \right)^{-1} \mathbf{X}^\top \mathbf{W}^{1/2},$$

with $\mathbf{W} = diag\{\pi_i(1 - \pi_i)\}$. For further details see, for instance, Heinze (1999).
Following this, the supervised principal components and their related coefficients outputted from the functions `superpc.with.prevalidation(·)` or `superpc.without.prevalidation(·)`, are passed to the function `score(·)` which in turn com-

putes the score values of the pre-validated or non pre-validated omics score for every observation in the data.

After the omics score has been generated and the prediction rule has been built, we can focus on the actual problem of how the added predictive value can be measured.

4 Assessment of the added predictive value

Boulesteix and Sauerbrei (2011) and De Bin et al. (2014) provide different techniques to assess the added predictive value. In the present thesis we will focus on two of them:

- testing the molecular score in a multivariate regression model adjusting for clinical predictors, and
- evaluating the predictive accuracy of the models with (combined model) and without (clinical model) the molecular score.

Since the main objective of this thesis is to determine whether pre-validation fulfills its tasks during the assessment of the added predictive value, we will describe the following approaches under the assumption that the molecular score is tested on the same data set on which it has been generated. Besides that, we follow the manner of De Bin et al. (2014) in the description below.

4.1 Testing the molecular score in a multivariate regression model

With respect to the multivariate regression model in expression (2.3) and its corresponding linear predictor

$$\eta_i = \gamma_0 + \gamma_1 \cdot z_{i1} + \ldots + \gamma_q \cdot z_{iq} + \beta_{score} x_{score,i},$$

we assess the added predictive value of the molecular predictor by testing the hypotheses

$$H_0 : \beta_{score} = 0 \quad \text{versus} \quad H_1 : \beta_{score} \neq 0.$$

This allows to draw conclusions about the connection between the molecular score and the response $\mathbf{y}$. For this purpose we use the Wald test in the present thesis. However, the likelihood ratio test or the score test would also be possible.
If the resulting p-value is smaller than a pre-defined significance level, the regression

coefficient β_{score} differs significantly from zero.
As mentioned above, the omics score usually tends to overfit the data at hand, whereby the regression coefficient β_{score} as well as the corresponding p-value is biased. For that reason we will check these results against the results brought from the pre-validated omics score. It means that we will also test the significance of the regression coefficient $\widetilde{\beta}_{score}$ of the pre-validated molecular score from the following model:

$$P(y_i = 1|\mathbf{x}_i, \mathbf{z}_i) = \frac{\exp(\gamma_0 + \gamma_1 \cdot z_{i1} + \ldots + \gamma_q \cdot z_{iq} + \widetilde{\beta}_{score} \cdot \widetilde{x}_{score,i})}{1 + \exp(\gamma_0 + \gamma_1 \cdot z_{i1} + \ldots + \gamma_q \cdot z_{iq} + \widetilde{\beta}_{score} \cdot \widetilde{x}_{score,i})}.$$

In the case that we can overcome the problem of overfitting by the usage of pre-validation, we expect on the one hand that the regression coefficient of the pre-validated omics score is smaller than the coefficient of the non pre-validated omics score i.e., $\widetilde{\beta}_{score} < \beta_{score}$. And on the other hand, we expect the p-value of the pre-validated omics score to be larger than the p-value of the non pre-validated omics score $\widetilde{p} > p$.

However, with reference to Altman and Royston (2000) usefulness is determined by how well a model works in practice, not by how many zeros there are in the associated p-values. Furthermore, it should be emphasized that the p-value decreases with increasing sample size and we will not get any information about the predictive ability of the prediction model while using this approach.
Thus, besides assessing the added predictive value by testing the significance in a multivariate regression model, it is also common practice to evaluate the prediction accuracy of the obtained model via investigating the discrimination ability within the scope of model validation. But also for this validation strategy, overfitting displays a serious issue.

4.2 Evaluating the predictive accuracy of the clinical and the combined model

For this approach, we usually need to fit two prediction models. Both of them are multivariate logistic regression models

$$P(y_i = 1|\mathbf{x}_i, \mathbf{z}_i) = \frac{\exp(\eta_i)}{1 + \exp(\eta_i)},$$

but they differ in their linear predictor.
Model 1, the clinical model, consists of the clinical predictors and has the linear

predictor

$$\eta_i^{clin} = \gamma_0 + \gamma_1 \cdot z_{i1} + \ldots + \gamma_q \cdot z_{iq}.$$

The second model considers the clinical predictors and the molecular score. The corresponding linear predictor is represented by

$$\eta_i^{comb} = \gamma_0 + \gamma_1 \cdot z_{i1} + \ldots + \gamma_q \cdot z_{iq} + \beta_{score} \cdot x_{score_i}.$$

Afterwards, the prediction accuracies of both models are compared.
The omics score provides additional predictive power if the prediction accuracy of the combined model is superior to the prediction accuracy of the clinical model (De Bin, Herold and Boulesteix, 2014).

Since the combined model is again fitted on the same data set which has been used for score generation, we expect this model to overfit the data at hand. If so, the combined model would seem to have better predictive power than it actually has. To avoid overfitting, we will also fit a third prediction model including the clinical predictors and the pre-validated molecular score, with the linear predictor

$$\eta_i^{prev} = \gamma_0 + \gamma_1 \cdot z_{i1} + \ldots + \gamma_q \cdot z_{iq} + \widetilde{\beta}_{score} \cdot \widetilde{x}_{score_i},$$

and compare it to the clinical and combined model from above.
In case of pre-validation fulfills its tasks, we expect the third model to perform worse than the second one, since the outcome $\mathbf{y}$ has not directly been used for score generation.

For the measurement of the prediction accuracies of these three prediction rules, we consider their discriminative ability. It measures how well the obtained prediction rule can distinguish between the two response classes $y = 0$ and $y = 1$.

Discrimination

To determine the discriminative ability we can proceed as follows with reference to Giancristofaro and Salmaso (2003).
First, we split our observations into two sub-groups, with one group containing all observations with positive outcome, and the other group containing all observations with $y = 0$. Afterwards, we use the prediction rule to predict the probabilities $\hat{P}(y = 1)$ for a positive outcome for every observation. When plotting the distributions of $\hat{P}(y = 1)$ for both sub-groups, we will see how well the prediction rule distinguishes between positive and negative outcomes. The discriminative ability is the better the less the two curves overlap each other. Figure 4.1 illustrates examples for good and bad discriminative abilities. The blue and red curves show

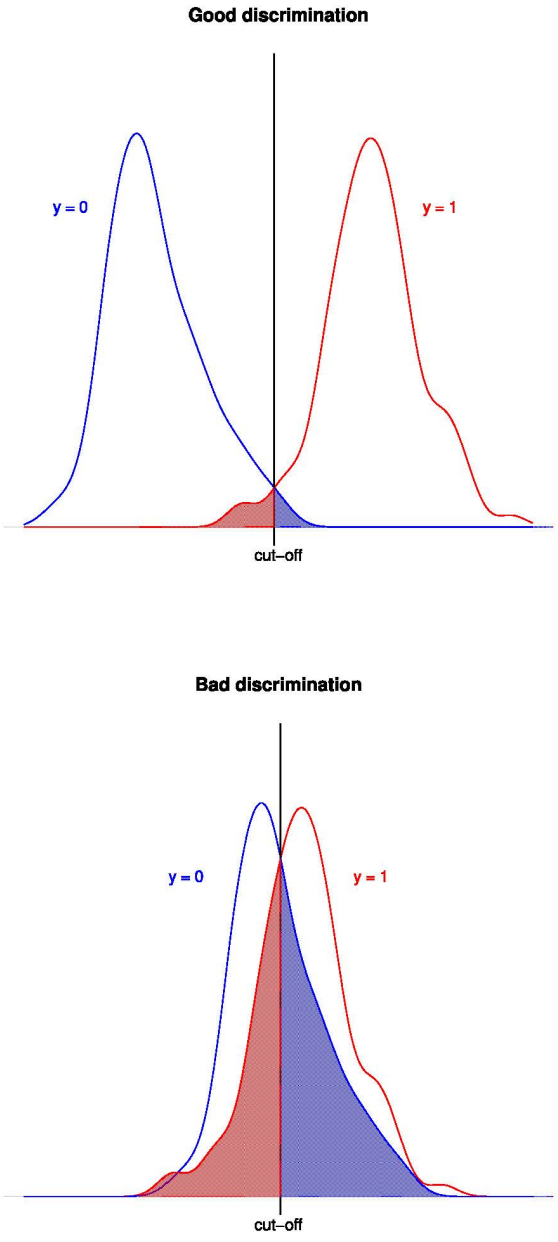

Figure 4.1: Examples of good and bad discrimination, modified according to Giancristofaro and Salmaso (2003). The red and blue areas display false negative and false positive classifications, respectively.

respectively the density functions of the observed negative and positive outcomes.

All observations with a predicted probability for a positive outcome greater or equal than a selected cut-off value are predicted to have $\hat{y} = 1$. Vice versa, observations with $\hat{P}(y = 1) <$ cut-off are classified to response class $\hat{y} = 0$:

$$\hat{y} = \begin{cases} 1 & \hat{P}(y = 1) \geq \text{cut-off} \\ 0 & \hat{P}(y = 1) < \text{cut-off.} \end{cases}$$

The natural consequence is the occurrence of misclassification which is illustrated by the colored areas in Figure 4.1. The blue areas display false positive classifications and the red areas false negative classifications. The best cut-off value is the probability where the chance for a wrong classification is minimal.
If a cut-off value is chosen, we can construct the classification table (fourfold table) which contains the frequencies of correct and incorrect classifications (see Table 4.1).
With the aid of Table 4.1 the (conditional) probabilities of correct and incorrect

	$y = 1$	$y = 0$
$\hat{y} = 1$	a	b
$\hat{y} = 0$	c	d

Table 4.1: Classification table

classifications can be computed.
Sensitivity denotes the probability of correct positive classifications, i.e.

$$\textit{sensitivity} = P(\hat{y} = 1 | y = 1) = \frac{a}{a + c}.$$

On the contrary,

$$\textit{specificity} = P(\hat{y} = 0 | y = 0) = \frac{d}{b + d}$$

measures the probability of correct negative classifications.
Both sensitivity and specificity are computed for every possible cut-off value.
When afterwards plotting *sensitivity* against $1 - \textit{specificity}$ i.e., the true positives against the false positives, we derive the so-called *receiver operating characteristic (ROC) curve*. The *area under the ROC curve* (*AUC*) is a measurement of the discriminative ability of the prediction model (Giancristofaro and Salmaso, 2003). The *AUC* ranges from 0.5 to 1, where 0.5 corresponds to a random classification – for example by coin tossing – and 1 corresponds to perfect discrimination. Thus, a prediction rule performs the better the closer its *AUC* is to 1.

4.3 Implementation in R

For the practical realization of these two validation strategies, we use already existing R-functions. To fit the multivariate regression models, we again use the function `brglm(·)` from the R-package of the same name. It computes the bias-reduced regression coefficients developed by Firth (1993) and also outputs the p-values derived by the Wald test. The *area under the receiver operating characteristic curve*, which represents the discriminative ability, is computed with the aid of the R-function `performance(·)` from the package `ROCR`.

5 Practical application

For the data-based comparison of pre-validated and non pre-validated molecular scores in binary classification both simulated and real data are used. Firstly, the simulation design and its implementation in R will be described. Later, the real data set used in this thesis is introduced.

5.1 Data simulation

The simulation of the data consists in artificially generating a data set with characteristics similar to real examples. It is an established procedure to test new methods, like the classification rules introduced in this thesis. The main advantage of this kind of data is that the truth is known. So we are able to compare the results with the truth and to figure out how well, for example, the classification rule works.

For the purposes of this thesis, it is necessary to simulate both clinical and omics data. Following the procedure of Oelker and Boulesteix (2013), we assume that the clinical predictors $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_q)$ and the molecular predictors $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)$ follow a normal distribution. Consequently, all predictors $(\mathbf{Z}, \mathbf{X}) \in \mathbb{R}^{n \times (q+p)}$ can be generated from a multivariate normal distribution $MVN(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where the mean is chosen to be zero for every predictor, $\boldsymbol{\mu} = (0, \ldots, 0)^\top \in \mathbb{R}^{(q+p)\times 1}$.
Before specifying the covariance matrix

$$\boldsymbol{\Sigma} = \left(\begin{array}{ccc|ccc} \sigma_{Z_1,Z_1} & \cdots & \sigma_{Z_1,Z_q} & \sigma_{Z_1,X_1} & \cdots & \sigma_{Z_1,X_p} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \sigma_{Z_q,Z_1} & \cdots & \sigma_{Z_q,Z_q} & \sigma_{Z_q,X_1} & \cdots & \sigma_{Z_q,X_p} \\ \hline \sigma_{X_1,Z_1} & \cdots & \sigma_{X_1,Z_q} & \sigma_{X_1,X_1} & \cdots & \sigma_{X_1,X_p} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \sigma_{X_p,Z_1} & \cdots & \sigma_{X_p,Z_q} & \sigma_{X_p,X_1} & \cdots & \sigma_{X_p,X_p} \end{array}\right) \in \mathbb{R}^{(q+p)\times(q+p)},$$

we will make some conjectures about the correlation structure which is closely related to $\boldsymbol{\Sigma}$. To be as close to truth as possible, we assume that some of the clinical and the molecular predictors are to be correlated among themselves and to each other.

The correlation matrix is a symmetric block matrix which can be written as

$$\mathbf{R} = \left(\begin{array}{c||c} \boldsymbol{\rho}_Z & \boldsymbol{\rho}_{ZX} \\ \hline\hline \boldsymbol{\rho}_{XZ} & \boldsymbol{\rho}_X \end{array} \right) \in \mathbb{R}^{(q+p)\times(q+p)}, \tag{5.1}$$

where the blocks $\boldsymbol{\rho}_Z$, $\boldsymbol{\rho}_X$ and $\boldsymbol{\rho}_{ZX} = \boldsymbol{\rho}_{XZ}^\top$ denote the correlation matrices of the clinical, the molecular and among the clinical and the molecular predictors, respectively. Appendix C contains a detailed description of the correlation matrix.
To simulate data sets with a pre-specified correlation structure, different techniques can be used. One possibility is to use the correlation matrix directly as covariance matrix. Therefore, it is necessary to make two restrictions. Firstly, the correlation matrix and the covariance matrix are identical if the standard deviations of two variables is 1. The consequence of this is that $diag(\boldsymbol{\Sigma}) = \mathbf{1}$, i.e. the variables are standardized.
The other important restriction refers to the property of positive (semi-) definiteness of both correlation and covariance matrices. Thus, $\mathbf{R}$ must be positive (semi-) definite. Because it is not very likely that an arbitrary created matrix $\mathbf{R}_{arb}$ is positive (semi-) definite, the most similar positive definite matrix to the given matrix is compiled.
For this purpose we use the algorithm described by Higham (2002) which computes the nearest correlation matrix $\mathbf{R}$ achieving the minimum of the distance $||\mathbf{R}_{arb} - \mathbf{R}||$ based on a weighted version of the Frobenius norm. Furthermore, $\mathbf{R}$ has to be symmetric. If these restrictions are complied, we can simulate the predictors from the distribution $MVN(\boldsymbol{\mu}, \mathbf{R})$.

Another possible way to simulate correlated predictors following a multivariate normal distribution is to use the Cholesky decomposition of the given correlation matrix $\mathbf{R} = \mathbf{U}^\top\mathbf{U}$. In order to do that, every predictor is simulated from a standard normal distribution $N(0, 1)$. Multiplication of the upper triangular matrix $\mathbf{U}$, derived by the Cholesky decomposition of the correlation matrix, with the matrix of the standard normal distributed predictors $(\mathbf{Z}, \mathbf{X})$, yields a transformed data set $((\mathbf{Z}, \mathbf{X})\mathbf{U}) \in \mathbb{R}^{n\times(q+p)}$ with the pre-specified correlation structure. For this approach the correlation matrix must also be symmetric and positive (semi-) definite with $diag(\mathbf{R}) = 1$.

Up to this point, both the clinical and the molecular predictors $\mathbf{Z}$ and $\mathbf{X}$ are

simulated. After the specification of the regression coefficients $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$, the linear predictor can be computed for every observation i, $i = 1, \ldots, n$. On basis of a logistic regression model, the probability for a positive response can be computed. The response variable is a Bernoulli random variable, where

$$P(y_i = 1|\mathbf{x_i}, \mathbf{z_i}) = \frac{\exp(\eta_i)}{1 + \exp(\eta_i)}.$$

The values of the binary outcome y_i are then generated with the aid of a Bernoulli distribution with probability π_i.
In the present thesis six different simulation settings are generated. They have been developed in collaboration with Dr. Riccardo De Bin from the *Department of Medical Informatics, Biometry and Epidemiology* of the *University of Munich*, who will also use similar simulation settings in future publications.

5.1.1 Simulation Settings

For every setting $n = 200$ observations are simulated. The number of clinical and molecular predictors is set to $q = 10$ and $p = 1000$, respectively. In every setting we discern between informative and non-informative predictors. It means that the clinical as well as the omics predictors may influence the response (regression coefficient $\neq 0$) or not (regression coefficient $= 0$).
The number of informative clinical predictors is specified to be 6, while the number of non-informative clinical predictors is to be 6. The regression coefficients of the clinical predictors are

$$\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_{10})^\top = (-2, -1.5, -1, 1, 1.5, 2, 0, 0, 0, 0)^\top$$

in each setting.
Within the molecular predictors, 20 of them are determined to influence the outcome while the other 980 are not related to $\mathbf{y}$ in the first four simulation settings. As for the clinical predictors, we also fix regression coefficients for the omics predictors in settings 1-4,

$$\boldsymbol{\beta} = (\beta_1, \ldots, \beta_{20}, \beta_{21}, \ldots, \beta_{1000})^\top = (0.75, \ldots, 0.75, 0, \ldots, 0)^\top.$$

Within the last two settings, $\boldsymbol{\beta}$ is equal to $\mathbf{0}$. It means that the molecular covariates do not influence the outcome at all. In general, we simulate six settings which vary in terms of the predictive ability of both the clinical and the omics predictors. See table 5.1 for an overview of the simulation settings.

		Predictive ability of clinical data	
		high	low
Predictive ability of molecular data	high	*setting 1*	*setting 3*
	low	*setting 2*	*setting 4*
	no	*setting 5*	*setting 6*

Table 5.1: Overview of the simulation settings.

Setting 1

In the first simulation scenario both the clinical and the molecular predictors affect the outcome strongly. With the aid of the correlation structure of the predictors we can additionally exert influence on the predictive ability. In this case, we want the clinical and the molecular predictors to be only low correlated to each other and among themselves. This will cause both predictor types to be crucial for predicting the response and thus, the omics predictors to supply a large added predictive value. As well as Oelker and Boulesteix (2013) we will use the values $\rho = 0.2$ and $\rho = 0.8$ for low and high correlations, respectively.

Setting 2

In the first simulation setting the 20 informative molecular predictors are supposed to have low predictive power. If the clinical predictors have high predictive ability, such as in this case, it is advisable to take a low correlation as basis. This has the consequence that the outcome mainly depends on the clinical predictors, which already explain a crucial part of the outcome variability. As opposed to this, the molecular predictors, which should provide a small contribution to outcome prediction, are simulated with high correlations among them and to the clinical predictors. As a result, they can merely explain a minor amount of the outcome variability.

Setting 3

In contrast to setting 2, the molecular predictors are supposed to have a high added predictive value. The clinical predictors, however, shall have just low predictive ability. To emphasize the predictive abilities, the clinical predictors are high correlated among themselves. As a result, the predictive power of the clinical part is narrowed. Vice versa, the omics predictors should have a small correlation

coefficient. Thus, added predictive value of the molecular predictors is increased.

Setting 4

In the fourth scenario, the clinical and the molecular predictors are highly correlated to each other and among themselves. Consequently, the added predictive value of the molecular data is small.

Setting 5

In contrast to the first four simulation settings, the molecular predictors should have no predictive power. Their correlation structure is neglected for this situation. It means that the molecular predictors are neither correlated among themselves nor to the clinical predictors. The other way round, the clinical predictors should strongly be related to $\mathbf{y}$. They are low correlated to each other.

Setting 6

The last simulation setting differs from Setting 5 with regard to the predictive ability of the clinical predictors. They are strongly correlated to each other and thus their prediction power is narrowed.

5.1.2 Implementation in R

For the practical realization of the simulation the R-function `simulation(·)` has been implemented. For the generation of a data set, the following parameters have to be pre-specified and provided to the function: the number of observations n, the number of both the clinical and the molecular predictors, the regression parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$, defining the influence of the predictors on the outcome, and the block correlation matrix $\mathbf{R} \in \mathbb{R}^{(q+p)\times(q+p)}$, containing the correlation structure that shall be achieved in the resulting data matrix.
The normal distributed data are either generated with aid of the R-function `rnorm(·)` from the package `stats`, or the function `mvrnorm(·)` from the package `MASS`, depending on whether Cholesky's decompositions is used or not.
To ensure that the inputted correlation matrix is at least positive semi-definite, its eigenvalues are checked to be greater than or equal to zero. Otherwise, the R-function `nearPD(·)` from the package `Matrix` computes the nearest positive definite matrix.
After the simulation of the clinical and molecular predictors is completed, the vector of the outcome $\mathbf{y}$ can be simulated using the function `rbinom(·)`.

Besides simulated data, we will also use real data to compare the pre-validated and the non pre-validated omics scores. The description of the real data can be found in the following section.

5.2 Breast cancer data

The real data set to be analyzed in the present thesis is taken from Hatzis et al. (2011). A huge advantage of this data is that it is freely accessibly online.
Originally, this data set served for predicting response and survival outcome from chemotherapy for newly diagnosed invasive breast cancer (Hatzis et al., 2011). The goal was to figure out whose clinical-pathologic risk at presentation favors the use of chemotherapy since it improves survival prognosis (Hatzis et al., 2011).
The collected data originate from a prospective multicenter study conducted from June 2000 to March 2010 at the M. D. Anderson Cancer Center in Houston, Texas (Hatzis et al., 2011) . They included a total of 310 patients in the training data and 198 patients in the validation data with newly diagnosed ERBB2 (HER2 or HER2/neu)- negative breast cancer treated with chemotherapy (Hatzis et al., 2011). For our purpose, we will only use the training data from Hatzis et al. (2011). After the exclusion of the missing values, we have 281 observations left. Furthermore, patients with indeterminate progesterone status are not further considered, since this group only includes 4 observations. Likewise, the two patients with tumor grade "T0" are excluded from further analysis. Altogether the data set consists of 275 patients.
With the aid of gene expression microarrays from Affymetrix, different predictive signatures for resistance and response to preoperative (neoadjuvant) chemotherapy have been developed (Hatzis et al., 2011).

For the usage of Hatzis' breast cancer data in the context of binary classification, the response of the tumor to neoadjuvant chemotherapy forms the new outcome. The residual cancer burden (RCB) developed by Symmans et al. (2007), helps with the quantification of residual tumor and is based upon the fact that the neoadjuvant chemotherapy influences the morphologic changes of the residual tumor (Schermann, 2014). Generally, there are four groups of residual cancer burden:

- RCB-0: no residual disease,
- RCB-I: minimal residual disease,
- RCB-II: moderate residual disease, and
- RCB-III: extensive residual disease.

It could be shown that residual cancer burden is highly associated to the tumor response, wherefore the binary outcome is built as $y = 0$ if residual cancer burden

is of RCB-0 or RCB-I which equals a high advantage of neoadjuvant chemotherapy and $y = 1$ if residual cancer burden is of RCB-II or RCB-III. Thus, $y = 1$ represents a poor prognosis for patients despite neoadjuvant chemotherapy (Schermann, 2014).
Summarized, the outcome variable can be outlined as

$$y = \begin{cases} 0 & \text{chemosensitivity} \quad \text{(no or minimal residual disease)} \\ 1 & \text{chemoresistance} \quad \text{(moderate or extensive residual disease)} \end{cases}$$

after neoadjuvant chemotherapy. For the prediction of the residual cancer burden six clinical predictors are used. The age of the patients, the progesterone receptor status, the estrogen receptor status, the tumor stage, the nodal status, and the tumor grade. The baseline characteristics of the clinical predictors are described in Table 5.2.
Furthermore, for every patient 22,283 probe sets (gene expression values) have been collected for outcome prediction.

In the next section the classification results of the simulated as well as the real data is represented.

5.3 Results

Altogether, seven regression models have been fit to the simulated data.
Three models each with a non pre-validated and a 5-fold pre-validated omics score have been fit using the *Lasso*, the *superPC* analysis without adjustment for the clinical predictors during variable selection, and the *superPC* analysis with adjustment for the clinical predictors during variable selection. The seventh model is the clinical model, only containing the clinical predictors. An overall view of all results are represented in Appendix D.
The value $G = 5$ for the number of pre-validation folds has been chosen since it is a common value for cross-validation, which is very similar to pre-validation (see, for instance Tibshirani and Efron, 2002). Leave-one-out ($G = n$) pre-validation would be deterministic and the variance estimates would be high. Small values for G would lead to too small training sets relative to the full training set (Tibshirani and Efron, 2002). Furthermore, $G = 5$ leads to good tradeoff between the complexity of pre-validation and computation time.

Simulation setting 1

Looking at the results of simulation setting 1 in Table 5.3, we can as anticipated see that the regression coefficients of the pre-validated scores (right part of the table) are all smaller than their non pre-validated counterparts.

	Training cohort		Validation cohort	
Age				
Mean(SD)	50.29	(10.68)	49.7	(11.04)
Progesterone receptor status				
Negative	148	(0.53)	50	(0.46)
Positive	129	(0.47)	58	(0.54)
Estrogen receptor status				
Negative	117	(0.42)	33	(0.31)
Positive	160	(0.58)	75	(0.69)
Tumor stage				
1	19	(0.07)	3	(0.03)
2	153	(0.55)	58	(0.54)
3	57	(0.21)	28	(0.26)
4	46	(0.17)	18	(0.17)
Nodal status				
0	84	(0.30)	41	(0.38)
1	127	(0.46)	46	(0.43)
2	37	(0.13)	15	(0.14)
3	29	(0.10)	6	(0.06)
Tumor grade				
1	18	(0.06)	9	(0.08)
2	113	(0.41)	36	(0.33)
3	146	(0.53)	63	(0.58)

Table 5.2: Baseline characteristics of the clinical predictors in Hatzis' breast cancer data.

It is also recognizable that the p-values of the Wald test are clearly higher than common significance levels like 1 % or 5 %. This result is according to our expectations since in the first simulation setting the molecular predictors own high prediction ability and are thus important for outcome prediction.
Furthermore, the results of the supervised principal component analysis show

				Without pre-validation	5-fold pre-validation
Lasso			β_{score}	2.09091	1.36930
			p_{score}	$9.75 \cdot 10^{-9}$	$2.69 \cdot 10^{-7}$
			AUC	0.95280	0.92069
superPC	without	adjustment	β_{score}	1.1550	0.45938
			p_{score}	$2.94 \cdot 10^{-9}$	$1.11 \cdot 10^{-5}$
			AUC	0.97560	0.89999
	with		β_{score}	1.77087	0.77207
			p_{score}	$1.88 \cdot 10^{-8}$	$1.67 \cdot 10^{-6}$
			AUC	0.98880	0.90929

Table 5.3: Comparison of pre-validated and non pre-validated molecular scores in logistic regression models on simulation setting 1.

higher regression coefficients if adjustment for the clinical predictors has been performed during the variable selection process. Looking at the values of the AUC, it occurs that the combined model including the pre-validated omics score clearly has lower values and thus, less discriminative ability. The clinical model has an AUC of the value 0.85839 i.e., it has lower discriminative ability than all of the combined prediction models.

Simulation setting 2

As well as in the first setting, all of the pre-validated omics scores have regression coefficients of larger (absolute) values than the non pre-validated molecular scores, as can be seen from Table 5.4. However, the different signs of the regression coefficients of the *superPC*-scores catch the eye.
For each of the three approaches, the non pre-validated omics scores significantly influence the response. The p-values of the *superPC*-scores are clearly larger in case of pre-validation has been performed. Both of them exceed commonly used significance levels. The pre-validated molecular score derived by *Lasso* regression generates less obvious results. Admittedly, the p-value decreases but it lies in the borderline between the two mentioned significance levels. It means that for the choice of 0.05 as significance level, the regression coefficient of the score differs significantly from zero, whereas it does not for the significance level 0.01.

				Without pre-validation	5-fold pre-validation
Lasso			β_{score}	4.119985	1.78445
			p_{score}	0.000729	0.040145
			AUC	0.99130	0.98289
superPC	without	adjustment	β_{score}	0.96547	-0.002267
			p_{score}	0.004812	0.994465
			AUC	0.98709	0.98099
	with		β_{score}	4.77079	-0.61718
			p_{score}	0.000259	0.276850
			AUC	0.99950	0.98279

Table 5.4: Comparison of pre-validated and non pre-validated molecular scores in logistic regression models on simulation setting 2.

The AUCs in this simulation setting are again smaller if pre-validation has been used to generate the omics score. Except for the model with the pre-validated *superPC*-score with adjustment, all combined models have a higher AUC than the clinical model (AUC=0.98099). However, the differences between the AUC of the clinical model and the combined models containing pre-validated molecular scores, is not huge.

Simulation setting 3

Within the scope of simulation setting 3 we can observe similar results as in setting 1, where the omics score also provides a large added predictive value. The results are shown in Table 5.5.
The values of the regression coefficients decrease while the p-values increase when the score has been derived by pre-validation. Both β-coefficients of the pre-validated *superPC*-scores are clearly significant, whereas the p-value of the pre-validated *Lasso*-score again lies in the borderline between 0.01 and 0.05. Without pre-validation each of the molecular scores shows significance.
Also in this setting, the AUC is reduced in the case of pre-validation has been performed during score generation. The discriminative ability of the clinical model (AUC=0.85343) is exceeded by each combined model.

				Without pre-validation	5-fold pre-validation
Lasso			β_{score}	0.74370	0.16428
			p_{score}	$2.79{\cdot}10^{-6}$	0.010838
			AUC	0.90085	0.86416
superPC	without	adjustment	β_{score}	1.05846	0.47376
			p_{score}	$1.04{\cdot}10^{-9}$	0.000155
			AUC	0.95599	0.88521
	with		β_{score}	1.59978	0.718874
			p_{score}	$2.33{\cdot}10^{-9}$	$7.58{\cdot}10^{-6}$
			AUC	0.97614	0.89995

Table 5.5: Comparison of pre-validated and non pre-validated molecular scores in logistic regression models on simulation setting 3.

Simulation setting 4

				Without pre-validation	5-fold pre-validation
Lasso			β_{score}	1.68618	1.19081
			p_{score}	0.000124	0.005002
			AUC	0.97949	0.99179
superPC	without	adjustment	β_{score}	0.7899	0.36192
			p_{score}	$1.13{\cdot}10^{-5}$	0.005157
			AUC	0.98920	0.97109
	with		β_{score}	1.23909	0.36265
			p_{score}	$5.68{\cdot}10^{-6}$	0.057260
			AUC	0.99130	0.96639

Table 5.6: Comparison of pre-validated and non pre-validated molecular scores in logistic regression models on simulation setting 4.

Simulation setting 4 indicates low predictive ability for both the clinical and the omics covariates. It can be observed from Table 5.6 that without pre-validation each regression coefficient differs significantly from zero. With pre-validation the coefficients decrease but only the β-coefficient of the *superPC*-score with adjustment looses its significance.
We can see that the model with the pre-validated *Lasso*-score has a higher *AUC*-value than its non pre-validated counterpart. The *superPC*-score lead to lower *AUC*s in case of pre-validation. All of the combined models outperform he discriminative ability of the clinical model (*AUC*=0.96239).

Simulation setting 5

				Without pre-validation	**5-fold pre-validation**
Lasso			β_{score}	0.05065	0.06716
			p_{score}	0.301241	0.160883
			AUC	0.9373	0.9384
superPC	without	adjustment	β_{score}	0.70669	-0.252203
			p_{score}	$1.21 \cdot 10^{-5}$	0.086206
			AUC	0.9626	0.9393
	with		β_{score}	1.23909	0.36265
			p_{score}	$1.11 \cdot 10^{-6}$	0.84399
			AUC	0.9914	0.9355

Table 5.7: Comparison of pre-validated and non pre-validated molecular scores in logistic regression models on simulation setting 5.

In the fifth simulation setting the differences between pre-validation and non pre-validation are not as clear as expected. The results are displayed in Table 5.7. Applying pre-validation to the *Lasso* yields a larger regression coefficient ans a smaller p-value in marked contrast to our expectation. It is noteworthy that the non pre-validated *Lasso* score is closer to the truth than its pre-validated counterpart. In line with setting 2, the signs of the *superPC*-scores reverse when pre-validation is performed. However, none of the regression coefficients of the pre-validated scores shows significance. The *AUC* of the clinical model is 0.9353 and thus very close to the *AUC*s of the combined models including the pre-validated

scores. It means that altogether no added predictive value can be revealed which is in common with the simulation design.

Simulation setting 6

				Without pre-validation	5-fold pre-validation
Lasso			β_{score}	-0.09667	-0.210018
			p_{score}	0.07981	0.005996
			AUC	0.96140	0.96620
superPC	without	adjustment	β_{score}	1.02174	0.067696
			p_{score}	$1.39 \cdot 10^{-5}$	0.60744
			AUC	0.98830	0.96030
	with	adjustment	β_{score}	3.12020	0.26675
			p_{score}	$1.05 \cdot 10^{-5}$	0.48099
			AUC	0.99330	0.96010

Table 5.8: Comparison of pre-validated and non pre-validated molecular scores in logistic regression models on simulation setting 6.

As in the setting before, the molecular data provides no predictive power and the clinical data owns only low prediction ability. Looking at Table 5.8 we again can observe that the *Lasso*-score behaves conversely than expected. The pre-validated version has a higher (absolute) coefficient value and a smaller associated p-value than the non-pre-validated molecular score. Both of the *superPC*-scores are significant in case of non pre-validation. However, when applying pre-validation both of them have p-values clearly higher than the common significance levels. The clinical model has an AUC-value of 0.95880 which is close to the results of the pre-validated scores. Also the model including the non pre-validated *Lasso*-score leads to similar results.

Altogether, in each of the simulation settings the *superPC* approach with adjustment for the clinical predictors yields larger (absolute) regression coefficients than without adjustment. This seems to be consistent with Bøvelstad's (2009) advice to adjust for the clinical predictors since then the *superPC*-score is stronger related to the outcome. On basis of the simulation results, we cannot generally confirm

Tibshirani and Efron's result that the clinical predictors strengthen in case of pre-validation has been performed. However, in the settings 5 and 6, when the molecular data has no influence on the response, the regression coefficients of the clinical predictors become larger when the omics score has been derived with the usage of pre-validation.
In the following, the results of the analysis of Hatzis' breast cancer data will be described.

Hatzis' breast cancer data

				Without pre-validation	**5-fold pre-validation**
Lasso			β_{score}	0.35718	0.04025
			p_{score}	0.0988	0.34821
			AUC	0.78032	0.77487
superPC	without	adjustment	β_{score}	1.12291	0.44676
			p_{score}	$2.43 \cdot 10^{-7}$	0.01195
			AUC	0.84077	0.78583
	with		β_{score}	1.02228	0.09555
			p_{score}	$4.68 \cdot 10^{-11}$	0.34871
			AUC	0.88865	0.77391

Table 5.9: Comparison of pre-validated and non pre-validated molecular scores in logistic regression models on Hatzis' breast cancer data.

When looking at Table 5.9 it is recognizable that all pre-validated omics scores have smaller regression coefficients with larger p-values than their non pre-validated versions. Both *superPC*-scores are significant without pre-validation. The non pre-validated *Lasso*-score as well as the pre-validated *Lasso*-score and the pre-validated *superPC*-score with adjustment are clearly not significant. The p-value of the pre-validated *superPC*-score obtained without adjustment for the clinical predictors during variable selection, lies in the borderline between 0.01 and 0.05. The area under the ROC curve is for all models including pre-validated molecular scores smaller than for the prediction models which contain the non pre-validated scores. Compared with the $AUC = 0.77183$ of the clinical model, the inclusion of the pre-validated omics score leads to higher AUC-values but the difference is not

huge.
In contrast to all simulation settings, we cannot observe that the adjustment for clinical predictors during the generation of the *superPC*-score yields larger regression coefficients. Furthermore, only few of the clinical predictors strengthen in case of pre-validation has been performed.
Considering these results the omics data does not seem to provide an added predictive value compared to the clinical data. This finding is also consistent with conclusions of De Bin, Sauerbrei and Boulesteix (2014).
The results can be found in detail in Appendix E.

6 Summary

High-dimensional molecular data such as microarray data display an actual and important research area. The handling of the $n << p$ *problem* and the combination of low- and high-dimensional predictors is a serious challenge. Also the validation of the added predictive value of the omics data – in form of a new generated molecular score – compared to standard clinical predictors on the identical data set that has already been used to build the score, is a non-trivial issue. Since microarray predictors tend to overfit the available data, the omics score might seem to be more relevant for outcome prediction than it actually is.
Goal of the present thesis was on the one hand the implementation of the *Lasso* and the *superPC* analysis in the scope of generating omics scores, and on the other hand the verification whether pre-validation is an appropriate approach to solve the problem of overfitting, and allows a fairer comparison between the different types of predictors. The results of both, the simulated and the real breast cancer data from Hatzis show that molecular scores which have been derived by pre-validation have smaller estimated coefficients in the multivariate regression model adjusted for the clinical predictors than their non pre-validated counterparts. Analogously, when applying a Wald test for the determination whether a regression coefficient significantly differs from zero, we can observe that the pre-validated omics scores are all less significant than the non-prevalidated ones. Also the measurement of the area under the receiver operating characteristic curve shows that, apart from one exception (*Lasso*-score in simulation setting 4), the prediction models containing pre-validated molecular scores have lower discriminative ability than the regression models including a non pre-validated score.
Special attention should be given to the results of simulation settings 5 and 6, where the molecular data has no predictive ability. With exception of the *Lasso*-score in setting 6, none of the pre-validated molecular scores is significant in the multivariate regression model, while most of the non pre-validated scores are. However, analysis of the AUC do not show such clear results. The clinical model does not lead to an higher AUC-value compared to the combined models, although the molecular data has no predictive ability in both simulation settings.
The results of this thesis are altogether consistent with Tibshirani and Efron's assertion that pre-validation is a suitable method to at least reduce the problem of overfitting during the assessment of the added predictive value. However, it should be noted that pre-validation cannot replace proper validation if independent val-

idation data is available. Also the application of a permutation test instead of a standard Wald test should be considered. Since the i.i.d. assumption in the generalized linear model is violated, the asymptotic distribution of the test statistic is not a t- or normal distribution (Oelker and Boulesteix, 2013, Höfling and Tibshirani, 2008).

Aside from that, someone should consider that we have many degrees of freedom in this thesis. Firstly, the binary classification has not necessarily be performed by a logistic regression model. For example, a probit regression would also be possible. Extension concerning the linearity of the predictor and interactions between covariates are not precluded.
Secondly, in the context of pre-validation, the number of pre-validation folds is optional. Also, for example, in the *superPC* approach, other selection methods for generating the top-list would be possible. Boulesteix and Slawski (2009) have implemented a lot of alternates in their Bioconductor package `geneSelector`. Additionally, besides the *Lasso* and *superPC* analysis, other conceivable methods can be used for score generation, like partial least squares, Ridge regression or random forests, to name just a few.
Moreover, in the process of data simulation many decisions need to be taken. The number of observations, the number of clinical and molecular predictors, the regression coefficients and the correlation structure is freely selectable.

Bibliography

Altman, D. and Royston, P. (2000). What do we mean by validating a prognostic model?, *Statistics in Medicine* **19**: 453–473.

Bair, E. and Tibshirani, R. (2004). Semi-supervised methods to predict patient survival from gene expression data, *Public Library of Science, Biology* **2**: 511–522.

Boulesteix, A.-L., Porzelius, C. and Daumer, M. (2008). Microarray-based classification and clinical predictors: On combined classifiers and additional predictive value, *Bioinformatics* **24**: 1698–1706.

Boulesteix, A.-L. and Sauerbrei, W. (2011). Added predictive value of high-throughput moleular data to clinical data and its validation, *Briefings in Bioinformatics* **12**: 215–229.

Boulesteix, A.-L. and Slawski, M. (2009). Stability and aggregation of ranked gene lists, *Technical Report 59*, Department of Statistics, University of Munich.

Boulesteix, A.-L., Strobl, C., Augustin, T. and Daumer, M. (2008). Evaluating microarray-based classifiers: An overview, *Cancer Informatics* **6**: 77–97.

Bøvelstad, H., Nygård, S. and Borgan, Ø. (2009). Survival prediction from clinico-genomic models – a comparative study, *BMC Bioinformatics* **10**.

Bühlmann, P. and van de Geer, S. (2011). *Statistics for High-Dimensional Data. Methods, Theory and Application*, 1st edn, Springer, Berlin.

Dalma-Weiszhausz, D., Warrington, J., Tanimoto, E. and Miyada, C. (2006). The Affymetrix GeneChip® platform: An overview, *Methods in Enzymology* **410**: 3–28.

De Bin, R., Herold, T. and Boulesteix, A.-L. (2014). Added predictive value of omics data: Specific issues related to validation illustrated by two case studies, *Technical Report 154*, Department of Statistics, University of Munich.

De Bin, R., Sauerbrei, W. and Boulesteix, A.-L. (2014). Investigating the prediction ability of survival models based on both clinical and omics data: Two case studies, *Technical Report 153*, Department of Statistics, University of Munich.

Duggan, D., Bittner, M., Chen, Y., Meltzer, P. and Trent, J. (1999). Expression profiling using cDNA microarrays, *Nature Genetics* **21**: 10–14.

Fahrmeir, L., Kneib, T., Lang, S. and Marx, B. (2013). *Regression. Models, Methods and Application*, 1st edn, Springer, Berlin.

Firth, D. (1993). Bias reduction of maximum likelihood estimates, *Biometrika* **80**: 27–38.

Giancristofaro, R. and Salmaso, L. (2003). Model performance analysis and model validation in logistic regression, *Statistica* **63**: 375–396.

Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The Elements of Statistical Learning. Data Mining, Inference, and Prediction*, 2nd edn, Springer, New York.

Hatzis, C., Pusztai, L., Valero, V., Booser, D., Esserman, L., Lluch, A., Vidaurre, T., Holmes, F., Souchon, E., Wang, H., Martin, M., Cotrina, J., Gomez, H., Hubbard, R., Chacón, J., Ferrer-Lozano, J., Dyer, R., Buxton, M., Gong, Y., Wu, Y., Ibrahim, N., Andreopoulou, E., Ueno, N., Hunt, K., Yang, W., Nazario, A., DeMichele, A., O'Shaughnessy, J., Hortobagyi, G. and Symmans, W. (2011). A genomic predictor of response and survival following taxane-anthracycline chemotherapy for invasive breast cancer, *The Journal of the American Medical Association* **305**: 1873–1881.

Heinze, G. (1999). The application of Firth's procedure to Cox and logistic regression, *Technical Report 10*, Department of Medical Computer Sciences, University of Vienna. (Updated in January 2001).

Higham, N. (2002). Computing the nearest correlation matrix – a problem from finance, *IMA Journal of Numerical Analysis* **22**: 329–343.

Höfling, H. and Tibshirani, R. (2008). A study of pre-validation, *The Annals of Applied Statistics* **2**: 643–664.

Jaenisch, R. and Bird, A. (2003). Epigenetic regulation of gene expression: how the genome integrates intrinsic and environmental signals, *Nature Genetics* **33**: 245–254.

Lai, C., Reinders, M., van't Veer, L. and Wessels, L. (2006). A comparison of univariate and multivariate gene selection techniques for classification of cancer datasets, *BMC Bioinformatics* **7**.

Lottaz, C., Kostka, D., Markowetz, F. and Spang, R. (2008). Computational diagnostics with gene expression profiles, *Methods in Molecular Biology* **453**: 281–296.

Mandal, A. (2014). Genetic inheritance. [Accessed January 27, 2014].
URL: *http://www.news-medical.net/health/Genetic-Inheritance.aspx*

National Center for Biotechnology Information (2014). Gene expression. [Accessed January 27, 2014].
URL: *http://www.ncbi.nlm.nih.gov/genome/probe/doc/ApplExpression.shtml*

National Human Genome Research Institute (2012). An overview of the human genome project. [Accessed January 27, 2014].
URL: *http://www.genome.gov/12011238*

Nguyen, D., Arpat, A., Wang, N. and Carroll, R. (2002). DNA microarray experiments: Biological and technological aspects, *Biometrics* **58**: 701–717.

Nikulin, V. and McLachlan, G. (2010). Penalized principal component analysis of microarray data, *in* F. Masulli, L. Peterson and R. Tagliaferri (eds), *Computational Intelligence Methods for Bioinformatics and Biostatistics - 6th International Meeting*, Springer, Berlin, pp. 82–96.

Ntzani, E. and Ioannidis, J. (2003). Predictive ability of DNA microarrays for cancer outcomes and correlates: an empirical assessment, *Lancet* **362**: 1439–1444.

Oelker, M.-R. and Boulesteix, A.-L. (2013). On the simultaneous analysis of clinical and omics data - a comparison of globalboosttest and pre-validation techniques, *in* P. Giudici, S. Ingrassia and M. Vichi (eds), *Proceedings of the 8th Scientific Meeting of the Classification and Data Analysis Group of the Italian Statistical Society*, pp. 259–268.

Peterson, L. (2013). *Classification Analysis of DNA Microarrays*, 1st edn, John Wiley & Sons, Inc., Hoboken, New Jersey.

Schermann, C. (2014). *Prospektive Evaluierung des Tumoransprechens mittels RCB (Resudial Cancer Burden) bei Frauen mit HER2 negativem Mammakarzinom nach neoadjuvanter Chemotherapie*, PhD thesis, Medizinische Universität Graz.

Science Creative Quarterly (2014). Spot your genes - an overview of the microarray. [Accessed May 8, 2014].
URL: *http://www.scq.ubc.ca/spot-your-genes-an-overview-of-the-microarray/*

Slawski, M., Daumer, M. and Boulesteix, A.-L. (2008). CMA - a comprehensive Bioconductor package for supervised classification with high dimensional data, *BMC Bioinformatics* **9**.

Symmans, W., Peintinger, F., Hatzis, C., Rajan, R., Kuerer, H., Valero, V., Assad, L., Poniecka, A., Hennessy, B., Green, M., Buzdar, A., Singletary, S., Hortobagyi, G. and Pusztai, L. (2007). Measurement of residual breast cancer burden to predict survival after neoadjuvant chemotherapy, *Journal of Clinical Oncology* **25**: 4414–4422.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso, *Journal of the Royal Statistical Society. Series B* **58**: 267–288.

Tibshirani, R. and Efron, B. (2002). Pre-validation and inference in microarrays, *Statistical Applications in Genetics and Molecular Biology* **1**: 1–18.

Tutz, G. (2013). Principal components analysis. [Accessed March 3, 2014]. **URL:** *http://www.statistik.lmu.de/institut/lehrstuhl/semsto/Lehre/Multivariate2013/PrincComp*

van Wieringen, W., Kun, D., Hampel, R. and Boulesteix, A.-L. (2009). Survival prediction using gene expression data: A review and comparison, *Computational Statistics & Data Analysis* **53**: 1590–1603.

Wikipedia (2014). Regulation of gene expression. [Accessed January 28, 2014]. **URL:** *http://en.wikipedia.org/wiki/Regulation_of_gene_expression*

List of Figures

List of Tables

Appendix

A Derivation of the log-likelihood function in logistic regression

Assumptions:

$y_i \overset{i.i.d.}{\sim} B(1, \pi_i)$,
where $\pi_i = P(y_i = 1) = \mathbb{E}(y_i) = h(\mathbf{x}_i^\top \boldsymbol{\beta})$

Density function:

$$f(y_i|\pi_i) = \pi_i^{y_i} \left(1 - \pi_i\right)^{1-y_i}$$

Likelihood function:

$$L(\boldsymbol{\beta}) \overset{i.i.d.}{=} \prod_{i=1}^{n} L_i(\boldsymbol{\beta}) = \prod_{i=1}^{n} \pi_i^{y_i} \left(1 - \pi_i\right)^{1-y_i}$$

Log-likelihood function:

$$\begin{aligned}
\ell(\boldsymbol{\beta}) &= \sum_{i=1}^{n} \log(L_i(\boldsymbol{\beta})) \\
&= \sum_{i=1}^{n} \left\{ y_i \log(\pi_i) - y_i \log(1 - \pi_i) + \log(1 - \pi_i) \right\} \\
&= \sum_{i=1}^{n} y_i \log\left(\frac{\pi_i}{1 - \pi_i}\right) + \log(1 - \pi_i)
\end{aligned}$$

From $\pi_i = \dfrac{\exp(\mathbf{x}_i^\top \boldsymbol{\beta})}{1 + \exp(\mathbf{x}_i^\top \boldsymbol{\beta})}$ follows

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left\{ y_i (\mathbf{x}_i^\top \boldsymbol{\beta}) - \log(1 + \exp(\mathbf{x}_i^\top \boldsymbol{\beta})) \right\}.$$

B Tuning of the penalization parameter

The error rate is best explained by using the the classification matrix for two possible outcomes $y = 0$ versus $y = 1$ below.

		Truth	
		$y = 0$	$y = 1$
Prediction	$\hat{y} = 0$	true negative TN	false negative FN
	$\hat{y} = 1$	false positive FP	true positive TP

According to this table the error rate ε i.e., the fraction of false predictions, can be estimated with reference to Slawski et al. (2008) (p. 10) as follows:

$$\varepsilon = \frac{FP + FN}{TN + TP + FN + FP},$$

where FP denotes the number of false positive predictions, TN the number of true negative predictions, et cetera.
The tuning parameter t can be estimated via the following cross-validation loop:

1. Divide the available observations into L subsets of approximate equal size.
2. Leave subset ℓ out and generate a classifier for every "candidate value" of t based on the Lasso.
3. Estimate the error rate for every candidate value on the left-out cases.
4. Repeat steps 2-3 for every $\ell \in L$.
5. Choose the candidate value with the smallest cross-validated error rate for t.

C Correlation matrix

$$\mathbf{R} = \left(\begin{array}{c|c} \boldsymbol{\rho}_Z & \boldsymbol{\rho}_{ZX} \\ \hline \boldsymbol{\rho}_{XZ} & \boldsymbol{\rho}_X \end{array}\right),$$

where

$$\boldsymbol{\rho}_Z = \underbrace{\left(\begin{array}{ccc|ccc} \rho_{Z_1,Z_1} & \cdots & \rho_{Z_1,Z_6} & \rho_{Z_1,Z_7} & \cdots & \rho_{Z_1,Z_{10}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \rho_{Z_6,Z_1} & \cdots & \rho_{Z_6,Z_6} & \rho_{Z_6,Z_7} & \cdots & \rho_{Z_6,Z_{10}} \\ \hline \rho_{Z_7,Z_1} & \cdots & \rho_{Z_7,Z_6} & \rho_{Z_7,Z_7} & \cdots & \rho_{Z_7,Z_{10}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \rho_{Z_{10},Z_1} & \cdots & \rho_{Z_{10},Z_6} & \rho_{Z_{10},Z_7} & \cdots & \rho_{Z_{10},Z_{10}} \end{array}\right)}_{\substack{\text{informative} \\ \text{clinical predictors}}\quad\quad\substack{\text{non-informative} \\ \text{clinical predictors}}} \begin{array}{l} \left.\vphantom{\begin{array}{c}a\\a\\a\end{array}}\right\} \text{informative clinical predictors} \\ \left.\vphantom{\begin{array}{c}a\\a\\a\end{array}}\right\} \text{non-informative clinical predictors} \end{array}$$

$$\boldsymbol{\rho}_X = \underbrace{\left(\begin{array}{ccc|ccc} \rho_{X_1,X_1} & \cdots & \rho_{X_1,X_{20}} & \rho_{X_1,X_{21}} & \cdots & \rho_{X_1,X_{1000}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \rho_{X_{20},X_1} & \cdots & \rho_{X_{20},X_{20}} & \rho_{X_{20},X_{21}} & \cdots & \rho_{X_{20},X_{1000}} \\ \hline \rho_{X_{21},X_1} & \cdots & \rho_{X_{21},X_{20}} & \rho_{X_{21},X_{21}} & \cdots & \rho_{X_{21},X_{1000}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \rho_{X_{1000},X_1} & \cdots & \rho_{X_{1000},X_{20}} & \rho_{X_{1000},X_{21}} & \cdots & \rho_{X_{1000},X_{1000}} \end{array}\right)}_{\substack{\text{informative} \\ \text{molecular predictors}}\quad\quad\substack{\text{non-informative} \\ \text{molecular predictors}}} \begin{array}{l} \left.\vphantom{\begin{array}{c}a\\a\\a\end{array}}\right\} \text{informative molecular predictors} \\ \left.\vphantom{\begin{array}{c}a\\a\\a\end{array}}\right\} \text{non-informative molecular predictors} \end{array}$$

and

$$\boldsymbol{\rho}_{ZX} = \left(\begin{array}{ccc|ccc} \rho_{Z_1,X_1} & \cdots & \rho_{Z_1,X_{20}} & \rho_{Z_1,X_{21}} & \cdots & \rho_{Z_1,X_{1000}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \rho_{Z_6,X_1} & \cdots & \rho_{Z_6,X_{20}} & \rho_{Z_6,X_{21}} & \cdots & \rho_{Z_6,X_{1000}} \\ \hline \rho_{Z_7,X_1} & \cdots & \rho_{Z_7,X_{20}} & \rho_{Z_7,X_{21}} & \cdots & \rho_{Z_7,X_{1000}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \rho_{Z_{10},X_1} & \cdots & \rho_{Z_{10},X_{20}} & \rho_{Z_{10},X_{21}} & \cdots & \rho_{Z_{10},X_{1000}} \end{array}\right)$$

informative clinical predictors

non-informative clinical predictors

informative molecular predictors

non-informative molecular predictors

The correlation matrix $\mathbf{R}$ will have the same structure for the first four simulation settings, but different values for the strength of correlations between the predictors. It is constructed as follows.

- Informative clinical predictors:
  - Z_1, Z_2 are correlated to informative omics predictors $X_1, \ldots, X_5$;
    - $\rightarrow$ Z_1, Z_2 are correlated among themselves;
    - $\rightarrow$ $X_1, \ldots, X_5$ are correlated among themselves;
  - Z_3, Z_4 are correlated to informative omics predictors $X_6, \ldots, X_{10}$;
    - $\rightarrow$ Z_3, Z_4 are correlated among themselves;
    - $\rightarrow$ $X_6, \ldots, X_{10}$ are correlated among themselves;
  - Z_5 is uncorrelated to other clinical predictors, but correlated to non-informative omics predictors $X_{21}, \ldots, X_{25}$;
    - $\rightarrow$ $X_{21}, \ldots, X_{25}$ are correlated among themselves;
  - Z_6 is correlated to non-informative clinical predictors Z_7, Z_8, and to informative omics predictors $X_{10}, \ldots, X_{20}$;
    - $\rightarrow$ $Z_6, \ldots, Z_8$ are correlated among themselves;
    - $\rightarrow$ $X_{10}, \ldots, X_{20}$ are correlated among themselves;
- Non-informative clinical predictors:
  - Z_9 is correlated to non-informative omics predictors $X_{26}, \ldots, X_{30}$;
    - $\rightarrow$ $X_{26}, \ldots, X_{30}$ are correlated among themselves;
  - Z_{10} is correlated to non-informative omics predictors $X_{31}, \ldots, X_{35}$;
    - $\rightarrow$ $X_{31}, \ldots, X_{35}$ are correlated among themselves;

- Informative molecular predictors:
  - X_1 is correlated to non-informative omics predictors $X_{36}, \ldots, X_{40}$;
    $\rightarrow$ $X_{36}, \ldots, X_{40}$ are correlated among themselves;
  - X_2 is correlated to non-informative omics predictors $X_{41}, \ldots, X_{45}$;
    $\rightarrow$ $X_{41}, \ldots, X_{45}$ are correlated among themselves;
  - ...
  - X_{15} is correlated to non-informative omics predictors $X_{106}, \ldots, X_{110}$;
    $\rightarrow$ $X_{106}, \ldots, X_{110}$ are correlated among themselves;

In simulation designs 5 and 6, the correlations regarding to the (non-informative) molecular predictors is neglected. That is, the molecular predictors are neither correlated among themselves nor to the clinical predictors. The clinical predictors retain the same correlation structure as described above with the exception that none of them is correlated to any molecular predictor.

D Results of the analysis of the simulated data

Setting 1

Lasso

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.04249	0.24034	0.177	0.859688
z1	-0.76003	0.25993	-2.924	0.003456
z2	-0.46998	0.24188	-1.943	0.052015
z3	-0.21053	0.23044	-0.914	0.360917
z4	1.01431	0.27254	3.722	0.000198
z5	0.94424	0.27188	3.473	0.000515
z6	1.32174	0.31408	4.208	$2.57 \cdot 10^{-5}$
z7	-0.04291	0.30935	-0.139	0.889689
z8	0.40820	0.26295	1.552	0.120566
z9	-0.11246	0.23620	-0.476	0.633974
z10	-0.29257	0.22680	-1.290	0.197066
score	2.09091	0.36459	5.735	$9.75 \cdot 10^{-9}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.05845	0.21154	0.276	0.782307
z1	-0.64501	0.22516	-2.865	0.004174
z2	-0.29824	0.20645	-1.445	0.148571
z3	-0.11581	0.20755	-0.558	0.576868
z4	0.89425	0.23873	3.746	0.000180
z5	0.77144	0.23061	3.345	0.000822
z6	1.38153	0.27864	4.958	$7.12 \cdot 10^{-7}$
z7	0.04164	0.26700	0.156	0.876063
z8	0.46689	0.23053	2.025	0.042838
z9	-0.01239	0.21138	-0.059	0.953269
z10	-0.15789	0.19564	-0.807	0.419631
score	1.36930	0.26619	5.144	$2.69 \cdot 10^{-7}$

SuperPC, without adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.1321	0.2803	-0.471	0.637386
z1	-1.1836	0.3435	-3.445	0.000571
z2	-0.4738	0.2848	-1.664	0.096191
z3	-0.2700	0.2816	-0.959	0.337629
z4	0.8559	0.3203	2.672	0.007545
z5	0.9586	0.3293	2.912	0.003596
z6	1.6405	0.4046	4.055	$5.01 \cdot 10^{-5}$
z7	-0.2708	0.3576	-0.757	0.448877
z8	0.1314	0.3165	0.415	0.677985
z9	0.4145	0.2737	1.515	0.129896
z10	-0.2096	0.2714	-0.772	0.439885
score	1.1550	0.1946	5.935	$2.94 \cdot 10^{-9}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.03656	0.19934	0.183	0.854482
z1	-0.60562	0.21345	-2.837	0.004550
z2	-0.32721	0.20704	-1.580	0.114018
z3	-0.11339	0.19515	-0.581	0.561219
z4	0.83867	0.23376	3.588	0.000334
z5	0.76962	0.21662	3.553	0.000381
z6	1.31246	0.26334	4.984	$6.23 \cdot 10^{-7}$
z7	0.02210	0.25559	0.086	0.931106
z8	0.30165	0.22087	1.366	0.172018
z9	0.02400	0.19545	0.123	0.902276
z10	-0.09041	0.18600	-0.486	0.626909
score	0.45938	0.10455	4.394	$1.11 \cdot 10^{-5}$

SuperPC, with adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.11065	0.33245	-0.333	0.739258
z1	-1.40397	0.39023	-3.598	0.000321
z2	-1.00104	0.38492	-2.601	0.009304
z3	-0.01714	0.32759	-0.052	0.958263
z4	1.74784	0.45793	3.817	0.000135
z5	1.82258	0.47115	3.868	0.000110
z6	2.56292	0.58666	4.369	$1.25 \cdot 10^{-5}$
z7	-0.40147	0.42981	-0.934	0.350267
z8	0.64259	0.37948	1.693	0.090391
z9	-0.02074	0.32200	-0.064	0.948645
z10	-0.28120	0.30889	-0.910	0.362635
score	1.77087	0.31495	5.623	$1.88 \cdot 10^{-8}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.04098	0.20536	0.200	0.841829
z1	-0.74096	0.23284	-3.182	0.001461
z2	-0.46388	0.21706	-2.137	0.032590
z3	-0.02111	0.21086	-0.100	0.920249
z4	0.87458	0.23704	3.690	0.000225
z5	0.75098	0.21659	3.467	0.000526
z6	1.40541	0.27907	5.036	$4.75 \cdot 10^{-7}$
z7	0.08026	0.26100	0.307	0.758470
z8	0.23377	0.22541	1.037	0.299684
z9	-0.13486	0.19303	-0.699	0.484773
z10	0.09421	0.19045	0.495	0.620836
score	0.77207	0.16117	4.790	$1.67 \cdot 10^{-6}$

Clinical model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.022292	0.183758	0.121	0.903442
z1	-0.525239	0.193095	-2.720	0.006526
z2	-0.232260	0.188395	-1.233	0.217637
z3	-0.006647	0.183096	-0.036	0.971042
z4	0.772416	0.204234	3.782	0.000156
z5	0.688903	0.199851	3.447	0.000567
z6	1.245375	0.232310	5.361	$8.28 \cdot 10^{-8}$
z7	0.357658	0.222872	1.605	0.108546
z8	0.459904	0.197327	2.331	0.019771
z9	-0.143343	0.178033	-0.805	0.420734
z10	-0.026256	0.165133	-0.159	0.873668

Setting 2

Lasso

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.318312	0.372822	-0.854	0.393221
z1	-0.110318	0.406392	-0.271	0.786039
z2	0.843967	0.417882	2.020	0.043422
z3	-0.011554	0.491621	-0.024	0.981250
z4	1.308445	0.598572	2.186	0.028820
z5	2.040548	0.573397	3.559	0.000373
z6	2.801141	1.159962	2.415	0.015741
z7	-0.587921	1.067398	-0.551	0.581772
z8	-0.571177	0.808422	-0.707	0.479857
z9	0.420072	0.349538	1.202	0.229444
z10	0.008594	0.280819	0.031	0.975586
score	4.119985	1.219508	3.378	0.000729

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.40746	0.34373	-1.185	0.235859
z1	0.04488	0.35946	0.125	0.900640
z2	0.59583	0.35882	1.661	0.096803
z3	0.44137	0.43116	1.024	0.305988
z4	1.54266	0.47601	3.241	0.001192
z5	1.35848	0.38240	3.553	0.000382
z6	3.29285	0.97472	3.378	0.000729
z7	0.74121	0.93849	0.790	0.429651
z8	0.76136	0.69712	1.092	0.274774
z9	0.27295	0.30353	0.899	0.368514
z10	0.00913	0.25868	0.035	0.971845
score	1.78445	0.86951	2.052	0.040145

SuperPC, without adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.08909	0.32496	0.274	0.783958
z1	-0.27381	0.38137	-0.718	0.472786
z2	0.04520	0.34584	0.131	0.896008
z3	0.12180	0.50347	0.242	0.808836
z4	0.85421	0.55347	1.543	0.122739
z5	1.34592	0.40676	3.309	0.000937
z6	2.32403	1.07185	2.168	0.030140
z7	0.43384	0.90620	0.479	0.632114
z8	0.17502	0.76367	0.229	0.818732
z9	0.23636	0.31886	0.741	0.458519
z10	0.14615	0.26633	0.549	0.583187
score	0.96547	0.34245	2.819	0.004812

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.088580	0.294121	-0.301	0.763285
z1	0.156963	0.330707	0.475	0.635052
z2	0.270047	0.304156	0.888	0.374617
z3	0.925657	0.487317	1.899	0.057499
z4	1.856048	0.539904	3.438	0.000587
z5	1.231106	0.352854	3.489	0.000485
z6	4.504160	1.090238	4.131	$3.61 \cdot 10^{-5}$
z7	2.312755	0.950741	2.433	0.014992
z8	1.950174	0.757715	2.574	0.010060
z9	0.081463	0.291023	0.280	0.779539
z10	0.109967	0.247601	0.444	0.656948
score	-0.002267	0.326776	-0.007	0.994465

SuperPC, with adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.26398	0.48327	-0.546	0.584902
z1	0.03333	0.57382	0.058	0.953685
z2	0.84266	0.59535	1.415	0.156948
z3	2.09549	0.94949	2.207	0.027316
z4	3.47637	0.99853	3.481	0.000499
z5	2.20744	0.67771	3.257	0.001125
z6	8.78544	2.12398	4.136	$3.53 \cdot 10^{-5}$
z7	3.57839	1.22181	2.929	0.003403
z8	3.36487	0.95649	3.518	0.000435
z9	-0.24357	0.54209	-0.449	0.653203
z10	-0.05414	0.40143	-0.135	0.892712
score	4.77079	1.30602	3.653	0.000259

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.03641	0.29496	-0.123	0.901757
z1	0.20671	0.32914	0.628	0.529987
z2	0.25466	0.30525	0.834	0.404145
z3	0.91697	0.35411	2.590	0.009611
z4	1.83078	0.44349	4.128	$3.66 \cdot 10^{-5}$
z5	1.27652	0.36303	3.516	0.000438
z6	4.62370	0.90412	5.114	$3.15 \cdot 10^{-7}$
z7	2.36651	0.65881	3.592	0.000328
z8	1.97538	0.47136	4.191	$2.78 \cdot 10^{-5}$
z9	0.03255	0.28606	0.114	0.909405
z10	0.11873	0.24241	0.490	0.624289
score	-0.61718	0.56756	-1.087	0.276850

Clinical model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.09166	0.29409	-0.312	0.755290
z1	0.15721	0.32626	0.482	0.629914
z2	0.27141	0.30558	0.888	0.374453
z3	0.93476	0.36024	2.595	0.009464
z4	1.87947	0.45327	4.146	$3.38 \cdot 10^{-5}$
z5	1.24613	0.35627	3.498	0.000469
z6	4.56033	0.89684	5.085	$3.68 \cdot 10^{-7}$
z7	2.34648	0.64898	3.616	0.000300
z8	1.97200	0.47692	4.135	$3.55 \cdot 10^{-5}$
z9	0.08208	0.28635	0.287	0.774376
z10	0.11150	0.24504	0.455	0.649105

Setting 3

Lasso

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.13588	0.19978	-0.680	0.496420
z1	-0.77277	0.33855	-2.283	0.022452
z2	-0.39344	0.31508	-1.249	0.211775
z3	-0.03491	0.34263	-0.102	0.918839
z4	0.47403	0.32139	1.475	0.140226
z5	0.69350	0.21403	3.240	0.001195
z6	1.54399	0.45947	3.360	0.000778
z7	-0.54616	0.42708	-1.279	0.200959
z8	0.23528	0.41388	0.568	0.569705
z9	-0.13588	0.19194	-0.708	0.478984
z10	0.15297	0.19031	0.804	0.421501
score	0.74370	0.15871	4.686	$2.79 \cdot 10^{-6}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.11104	0.18606	-0.597	0.550643
z1	-0.65040	0.31606	-2.058	0.039602
z2	-0.27964	0.29820	-0.938	0.348379
z3	-0.05842	0.31157	-0.187	0.851274
z4	0.47570	0.29971	1.587	0.112466
z5	0.56998	0.19573	2.912	0.003590
z6	1.39222	0.41384	3.364	0.000768
z7	-0.28828	0.37774	-0.763	0.445353
z8	0.16212	0.37022	0.438	0.661464
z9	-0.10284	0.17740	-0.580	0.562102
z10	0.20940	0.17448	1.200	0.230095
score	0.16428	0.06448	2.548	0.010838

SuperPC, without adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.03554	0.24030	0.148	0.882417
z1	-1.24688	0.46261	-2.695	0.007032
z2	-0.01197	0.40081	-0.030	0.976169
z3	-0.33816	0.42417	-0.797	0.425324
z4	0.70420	0.41855	1.682	0.092476
z5	0.74521	0.27136	2.746	0.006030
z6	2.19757	0.57851	3.799	0.000145
z7	-0.71724	0.49175	-1.459	0.144689
z8	-0.32259	0.49748	-0.648	0.516701
z9	0.07278	0.23801	0.306	0.759773
z10	0.16504	0.22513	0.733	0.463500
score	1.05846	0.17342	6.103	$1.04 \cdot 10^{-9}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.01618	0.19201	0.084	0.932855
z1	-0.85292	0.34225	-2.492	0.012698
z2	-0.16331	0.31499	-0.518	0.604129
z3	-0.03504	0.32719	-0.107	0.914709
z4	0.43963	0.31530	1.394	0.163225
z5	0.60993	0.20477	2.979	0.002895
z6	1.70854	0.44488	3.840	0.000123
z7	-0.47949	0.39309	-1.220	0.222537
z8	0.01217	0.38248	0.032	0.974607
z9	-0.04334	0.18412	-0.235	0.813904
z10	0.22919	0.18075	1.268	0.204794
score	0.47376	0.12523	3.783	0.000155

SuperPC, with adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.19460	0.28998	0.671	0.502174
z1	-1.85091	0.56887	-3.254	0.001139
z2	-0.09849	0.43609	-0.226	0.821313
z3	-0.23410	0.47994	-0.488	0.625706
z4	1.14692	0.50387	2.276	0.022833
z5	1.25336	0.35229	3.558	0.000374
z6	2.96283	0.73747	4.018	$5.88 \cdot 10^{-5}$
z7	-0.92074	0.54675	-1.684	0.092178
z8	-0.28659	0.59085	-0.485	0.627646
z9	-0.06738	0.27059	-0.249	0.803353
z10	0.32675	0.26606	1.228	0.219413
score	1.59978	0.26785	5.973	$2.33 \cdot 10^{-9}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.025703	0.199300	0.129	0.897384
z1	-0.924588	0.348223	-2.655	0.007927
z2	-0.196820	0.315128	-0.625	0.532251
z3	0.002161	0.334559	0.006	0.994846
z4	0.454078	0.320861	1.415	0.157014
z5	0.722984	0.214914	3.364	0.000768
z6	1.812805	0.462253	3.922	$8.79 \cdot 10^{-5}$
z7	-0.398837	0.389789	-1.023	0.306208
z8	-0.100955	0.402137	-0.251	0.801778
z9	-0.096367	0.188545	-0.511	0.609275
z10	0.271766	0.185307	1.467	0.142492
score	0.718874	0.160584	4.477	$7.58 \cdot 10^{-6}$

Clinical model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.06399	0.18125	-0.353	0.724051
z1	-0.68046	0.31419	-2.166	0.030331
z2	-0.21243	0.29808	-0.713	0.476056
z3	-0.01756	0.30340	-0.058	0.953853
z4	0.46131	0.29191	1.580	0.114035
z5	0.56590	0.19354	2.924	0.003456
z6	1.44754	0.40188	3.602	0.000316
z7	-0.12585	0.35802	-0.352	0.725192
z8	0.06621	0.35539	0.186	0.852205
z9	-0.09352	0.17658	-0.530	0.596383
z10	0.23791	0.16859	1.411	0.158197

Setting 4

Lasso

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.33611	0.29583	-1.136	0.255889
z1	-0.67402	0.53090	-1.270	0.204230
z2	0.45765	0.50014	0.915	0.360164
z3	-0.49747	0.52386	-0.950	0.342301
z4	1.77878	0.60817	2.925	0.003447
z5	1.26020	0.34174	3.688	0.000226
z6	1.50799	0.62410	2.416	0.015680
z7	0.85214	0.73392	1.161	0.245604
z8	0.77826	0.57826	1.346	0.178344
z9	0.08865	0.28753	0.308	0.757857
z10	0.17741	0.24807	0.715	0.474512
score	1.68618	0.43937	3.838	0.000124

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.25806	0.27150	-0.950	0.341865
z1	-0.39063	0.47074	-0.830	0.406639
z2	0.24087	0.45166	0.533	0.593825
z3	-0.28590	0.47805	-0.598	0.549801
z4	1.62200	0.53478	3.033	0.002421
z5	1.10447	0.30359	3.638	0.000275
z6	1.58672	0.58993	2.690	0.007152
z7	1.04448	0.68368	1.528	0.126581
z8	0.90086	0.54325	1.658	0.097261
z9	-0.02299	0.25749	-0.089	0.928846
z10	0.17849	0.23051	0.774	0.438728
score	1.19081	0.42424	2.807	0.005002

SuperPC, without adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.1116	0.3325	0.336	0.7372
z1	-0.3029	0.5264	-0.575	0.5651
z2	0.2014	0.5197	0.388	0.6984
z3	-0.8674	0.6377	-1.360	0.1738
z4	1.6194	0.7047	2.298	0.0216
z5	1.1797	0.4153	2.841	0.0045
z6	1.3580	0.7957	1.707	0.0879
z7	0.3818	0.8895	0.429	0.6678
z8	0.6690	0.6603	1.013	0.3110
z9	0.2710	0.3269	0.829	0.4071
z10	0.2806	0.2911	0.964	0.3351
score	0.7899	0.1799	4.390	$1.13 \cdot 10^{-5}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.03177	0.26787	0.119	0.905581
z1	-0.24722	0.46007	-0.537	0.591016
z2	0.07646	0.44549	0.172	0.863733
z3	-0.38928	0.46378	-0.839	0.401269
z4	1.49214	0.51959	2.872	0.004082
z5	1.15204	0.30470	3.781	0.000156
z6	1.51533	0.60367	2.510	0.012067
z7	1.07469	0.67680	1.588	0.112309
z8	0.85358	0.55099	1.549	0.121336
z9	0.09666	0.26220	0.369	0.712394
z10	0.23620	0.23653	0.999	0.317988
score	0.36192	0.12939	2.797	0.005157

SuperPC, with adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.09813	0.34769	-0.282	0.777754
z1	-0.65784	0.61551	-1.069	0.285172
z2	0.43134	0.57686	0.748	0.454620
z3	0.13352	0.58448	0.228	0.819308
z4	2.73819	0.76479	3.580	0.000343
z5	1.50923	0.44381	3.401	0.000672
z6	1.57377	0.76757	2.050	0.040332
z7	1.61507	0.80045	2.018	0.043622
z8	1.26184	0.67808	1.861	0.062756
z9	0.37316	0.36851	1.013	0.311245
z10	0.29423	0.29243	1.006	0.314333
score	1.23909	0.27305	4.538	$5.68 \cdot 10^{-6}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.24154	0.26132	-0.924	0.355324
z1	-0.22441	0.44785	-0.501	0.616304
z2	0.05785	0.43672	0.132	0.894619
z3	0.12019	0.41812	0.287	0.773765
z4	1.66862	0.47929	3.481	0.000499
z5	0.97146	0.29197	3.327	0.000877
z6	1.90066	0.58296	3.260	0.001113
z7	1.29111	0.62651	2.061	0.039322
z8	0.96741	0.52995	1.825	0.067933
z9	0.11918	0.26180	0.455	0.648955
z10	0.20686	0.21952	0.942	0.346025
score	0.36265	0.19074	1.901	0.057260

Clinical model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.15320	0.25095	-0.610	0.541539
z1	-0.10367	0.42931	-0.241	0.809190
z2	0.01355	0.41297	0.033	0.973830
z3	0.05617	0.40965	0.137	0.890932
z4	1.70649	0.46924	3.637	0.000276
z5	1.03004	0.28642	3.596	0.000323
z6	1.98483	0.57523	3.450	0.000560
z7	1.66752	0.61207	2.724	0.006442
z8	1.32588	0.50831	2.608	0.009097
z9	-0.03197	0.24894	-0.128	0.897822
z10	0.18210	0.21336	0.853	0.393387

Setting 5

Lasso

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.05924	0.22285	0.266	0.790354
z1	-1.49025	0.27410	-5.437	$5.42 \cdot 10^{-8}$
z2	-1.22899	0.28090	-4.375	$1.21 \cdot 10^{-5}$
z3	-0.52856	0.22980	-2.300	0.021442
z4	0.95391	0.26613	3.584	0.000338
z5	1.27598	0.27750	4.598	$4.26 \cdot 10^{-6}$
z6	1.48059	0.29795	4.969	$6.72 \cdot 10^{-7}$
z7	-0.05725	0.27037	-0.212	0.832307
z8	-0.15816	0.24801	-0.638	0.523645
z9	-0.25045	0.20871	-1.200	0.230155
z10	-0.05686	0.20934	-0.272	0.785910
score	0.05065	0.04899	1.034	0.301241

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.05497	0.22335	0.246	0.805605
z1	-1.49560	0.27429	-5.453	$4.96 \cdot 10^{-8}$
z2	-1.21392	0.28114	-4.318	$1.58 \cdot 10^{-5}$
z3	-0.53854	0.22886	-2.353	0.018614
z4	0.96526	0.26581	3.631	0.000282
z5	1.27315	0.27646	4.605	$4.12 \cdot 10^{-6}$
z6	1.54387	0.30779	5.016	$5.28 \cdot 10^{-7}$
z7	-0.03223	0.27046	-0.119	0.905137
z8	-0.12601	0.25073	-0.503	0.615262
z9	-0.29603	0.21144	-1.400	0.161502
z10	-0.04170	0.21188	-0.197	0.843991
score	0.06716	0.04790	1.402	0.160883

SuperPC, without adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.14503	0.25358	0.572	0.567367
z1	-1.10605	0.29945	-3.694	0.000221
z2	-1.02042	0.30027	-3.398	0.000678
z3	-0.38661	0.26206	-1.475	0.140131
z4	0.79932	0.29560	2.704	0.006851
z5	0.88844	0.28424	3.126	0.001774
z6	1.21647	0.32006	3.801	0.000144
z7	-0.19607	0.31006	-0.632	0.527142
z8	-0.06327	0.28230	-0.224	0.822654
z9	-0.15877	0.23828	-0.666	0.505214
z10	-0.06638	0.22812	-0.291	0.771050
score	0.70669	0.16151	4.375	$1.21 \cdot 10^{-5}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.019219	0.226287	0.085	0.932315
z1	-1.537303	0.281614	-5.459	$4.79 \cdot 10^{-8}$
z2	-1.325769	0.290774	-4.559	$5.13 \cdot 10^{-6}$
z3	-0.540766	0.232793	-2.323	0.020182
z4	0.987115	0.272667	3.620	0.000294
z5	1.368888	0.283169	4.834	$1.34 \cdot 10^{-6}$
z6	1.549643	0.309140	5.013	$5.37 \cdot 10^{-7}$
z7	-0.005321	0.262913	-0.020	0.983854
z8	-0.194767	0.249967	-0.779	0.435879
z9	-0.282099	0.210255	-1.342	0.179692
z10	-0.074753	0.211040	-0.354	0.723181
score	-0.252203	0.146992	-1.716	0.086206

SuperPC, with adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.37881	0.38824	0.976	0.329208
z1	-2.88013	0.57617	-4.999	$5.77 \cdot 10^{-7}$
z2	-2.40592	0.58271	-4.129	$3.65 \cdot 10^{-5}$
z3	-1.07622	0.40744	-2.641	0.008255
z4	1.72074	0.49496	3.477	0.000508
z5	2.32981	0.49434	4.713	$2.44 \cdot 10^{-6}$
z6	3.10514	0.65258	4.758	$1.95 \cdot 10^{-6}$
z7	0.06504	0.43714	0.149	0.881719
z8	-0.21016	0.42072	-0.500	0.617420
z9	-0.58047	0.29694	-1.955	0.050599
z10	-0.73167	0.34758	-2.105	0.035288
score	3.48984	0.71632	4.872	$1.11 \cdot 10^{-6}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.04486	0.22539	0.199	0.84222
z1	-1.47341	0.27123	-5.432	$5.56 \cdot 10^{-8}$
z2	-1.25514	0.28133	-4.461	$8.14 \cdot 10^{-6}$
z3	-0.53187	0.22940	-2.318	0.02042
z4	0.97766	0.26545	3.683	0.00023
z5	1.32117	0.27868	4.741	$2.13 \cdot 10^{-6}$
z6	1.48693	0.29991	4.958	$7.12 \cdot 10^{-7}$
z7	-0.02617	0.26910	-0.097	0.92252
z8	-0.16136	0.25081	-0.643	0.51999
z9	-0.26193	0.21288	-1.230	0.21854
z10	-0.05936	0.20644	-0.288	0.77369
score	-0.06299	0.32010	-0.197	0.84399

Clinical model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.05541	0.22197	0.250	0.802888
z1	-1.48823	0.27291	-5.453	$4.95 \cdot 10^{-8}$
z2	-1.26881	0.28200	-4.499	$6.82 \cdot 10^{-6}$
z3	-0.53980	0.22905	-2.357	0.018437
z4	0.98132	0.26598	3.689	0.000225
z5	1.32595	0.27845	4.762	$1.92 \cdot 10^{-6}$
z6	1.50443	0.30142	4.991	$6.00 \cdot 10^{-7}$
z7	-0.01854	0.26786	-0.069	0.944810
z8	-0.17564	0.24872	-0.706	0.480076
z9	-0.27356	0.20680	-1.323	0.185885
z10	-0.06174	0.20726	-0.298	0.765786

Setting 6

Lasso

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.15388	0.25185	0.611	0.54120
z1	-1.38330	0.49330	-2.804	0.00504
z2	-1.53228	0.47832	-3.203	0.00136
z3	-1.26615	0.46675	-2.713	0.00667
z4	1.32798	0.48062	2.763	0.00573
z5	1.68280	0.34145	4.928	$8.29 \cdot 10^{-7}$
z6	1.65327	0.57347	2.883	0.00394
z7	0.10970	0.52496	0.209	0.83447
z8	0.26634	0.51368	0.518	0.60412
z9	-0.01623	0.23194	-0.070	0.94421
z10	0.08002	0.23167	0.345	0.72979
score	-0.09667	0.05519	-1.752	0.07981

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.053355	0.260590	0.205	0.837771
z1	-1.155869	0.509650	-2.268	0.023331
z2	-1.948020	0.556712	-3.499	0.000467
z3	-1.467208	0.489608	-2.997	0.002729
z4	1.665782	0.522880	3.186	0.001444
z5	1.791102	0.359534	4.982	$6.3 \cdot 10^{-7}$
z6	1.736589	0.589805	2.944	0.003236
z7	0.305695	0.546495	0.559	0.575907
z8	0.085362	0.535544	0.159	0.873359
z9	-0.078525	0.247634	-0.317	0.751168
z10	0.006057	0.241498	0.025	0.979991
score	-0.210018	0.076425	-2.748	0.005996

SuperPC, without adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.01193	0.33727	-0.035	0.971794
z1	-1.36701	0.66764	-2.048	0.040608
z2	-1.47153	0.62841	-2.342	0.019198
z3	-1.07439	0.59695	-1.800	0.071892
z4	0.96639	0.57826	1.671	0.094680
z5	1.49787	0.41639	3.597	0.000322
z6	1.28508	0.73188	1.756	0.079113
z7	0.20126	0.65219	0.309	0.757633
z8	0.89622	0.69102	1.297	0.194652
z9	-0.16829	0.29317	-0.574	0.565933
z10	-0.08910	0.28960	-0.308	0.758342
score	1.02174	0.23513	4.345	$1.39 \cdot 10^{-5}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.136608	0.249357	0.548	0.58380
z1	-1.398810	0.485898	-2.879	0.00399
z2	-1.477466	0.476619	-3.100	0.00194
z3	-1.150635	0.449992	-2.557	0.01056
z4	1.196240	0.458929	2.607	0.00914
z5	1.686582	0.349057	4.832	$1.35 \cdot 10^{-6}$
z6	1.644784	0.577631	2.847	0.00441
z7	0.008917	0.511990	0.017	0.98610
z8	0.382793	0.507295	0.755	0.45050
z9	-0.004641	0.228406	-0.020	0.98379
z10	0.107201	0.228027	0.470	0.63827
score	0.067696	0.131772	0.514	0.60744

SuperPC, with adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.03139	0.37473	0.084	0.933244
z1	-2.35045	0.84844	-2.770	0.005600
z2	-3.01706	0.85955	-3.510	0.000448
z3	-1.95224	0.76330	-2.558	0.010538
z4	1.72421	0.75132	2.295	0.021738
z5	2.40155	0.53641	4.477	$7.57 \cdot 10^{-6}$
z6	3.32800	1.00855	3.300	0.000968
z7	0.48242	0.71389	0.676	0.499190
z8	0.62727	0.80800	0.776	0.437557
z9	-0.05259	0.29316	-0.179	0.857623
z10	0.45406	0.37884	1.199	0.230701
score	3.12020	0.70800	4.407	$1.05 \cdot 10^{-5}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.12454	0.25045	0.497	0.61901
z1	-1.38228	0.48400	-2.856	0.00429
z2	-1.53080	0.48345	-3.166	0.00154
z3	-1.14472	0.45049	-2.541	0.01105
z4	1.21206	0.46372	2.614	0.00895
z5	1.71218	0.35001	4.892	$9.99 \cdot 10^{-7}$
z6	1.67072	0.57887	2.886	0.00390
z7	-0.03710	0.51100	-0.073	0.94212
z8	0.42911	0.51047	0.841	0.40057
z9	-0.01430	0.22642	-0.063	0.94962
z10	0.08855	0.22757	0.389	0.69721
score	0.26675	0.37853	0.705	0.48099

Clinical model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.134412	0.250183	0.537	0.59109
z1	-1.416797	0.484356	-2.925	0.00344
z2	-1.490707	0.476718	-3.127	0.00177
z3	-1.140743	0.449685	-2.537	0.01119
z4	1.182022	0.458241	2.579	0.00989
z5	1.702642	0.349198	4.876	$1.08 \cdot 10^{-6}$
z6	1.638541	0.575852	2.845	0.00444
z7	0.009583	0.511925	0.019	0.98507
z8	0.382295	0.507403	0.753	0.45119
z9	-0.015820	0.230082	-0.069	0.94518
z10	0.105006	0.229398	0.458	0.64714

The following table shows the percentage of true informative molecular variables selected for building the omics score. In brakes stands the absolute number of selected genes in the Lasso approach. For the supervised principal component analysis the number of genes used to build the principal components is fixed to 25.

	w/o PV	**fold 1**	**fold 2**	**fold 3**	**fold 4**	**fold 5**
Setting 1						
Lasso	0.30 (47)	0.25 (36)	0.21 (43)	0.22 (54)	0.25 (48)	0.24 (41)
superPC.	0.48	0.48	0.52	0.44	0.4	0.44
superPC adj.	0.36	0.32	0.28	0.32	0.24	0.36
Setting 2						
Lasso	0.39 (23)	0.47 (19)	0.38 (21)	0.44 (18)	0.39 (23)	0.45 (20)
superPC.	0.64	0.6	0.6	0.6	0.6	0.6
superPC adj.	0.0	0.0	0.0	0.04	0.04	0.0
Setting 3						
Lasso	0.13 (98)	0.09 (107)	0.09 (106)	0.12 (109)	0.12 (106)	0.12 (103)
superPC.	0.44	0.48	0.4	0.44	0.4	0.36
superPC adj.	0.44	0.44	0.48	0.28	0.32	0.44
Setting 4						
Lasso	0.43 (21)	0.47 (19)	0.41 (17)	0.53 (17)	0.47 (15)	0.30 (20)
superPC.	0.6	0.6	0.6	0.6	0.6	0.6
superPC adj.	0.32	0.24	0.16	0.24	0.12	0.24
Setting 5						
Lasso	0.0 (154)	0.0 (128)	0.0 (133)	0.0 (129)	0.0 (121)	0.0 (127)
superPC.	0.0	0.0	0.0	0.0	0.0	0.0
superPC adj.	0.0	0.0	0.0	0.0	0.0	0.0
Setting 6						
Lasso	0.0 (146)	0.0 (118)	0.0 (125)	0.0 (110)	0.0 (124)	0.0 (106)
superPC.	0.0	0.0	0.0	0.0	0.0	0.0
superPC adj.	0.0	0.0	0.0	0.0	0.0	0.0

E Results of the analysis of Hatzis' breast cancer data

Lasso

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-15.47602	8.53086	-1.814	0.0697
age	0.02062	0.01426	1.447	0.1480
progesterone receptor status (negative)	0.42210	0.40629	1.039	0.2988
estrogen receptor status (positive)	0.82660	0.42138	1.962	0.0498
tumor stage (T2)	0.61783	0.57313	1.078	0.2810
tumor stage (T3)	0.77328	0.62181	1.244	0.2136
tumor stage (T4)	1.53867	0.68509	2.246	0.0247
nodal status (N1)	0.88404	0.35539	2.488	0.0129
nodal status (N2)	1.05891	0.51287	2.065	0.0390
nodal status (N3)	0.72300	0.55005	1.314	0.1887
tumor grade (2)	-0.13939	0.75723	-0.184	0.8540
tumor grade (3)	-1.11832	0.77580	-1.442	0.1494
score	0.35718	0.21635	1.651	0.0988

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-3.02340	1.97001	-1.535	0.12485
age	0.02173	0.01421	1.529	0.12624
progesterone receptor status (negative)	0.45545	0.40410	1.127	0.25970
estrogen receptor status (positive)	0.76428	0.41722	1.832	0.06697
tumor stage (T2)	0.76328	0.55538	1.374	0.16934
tumor stage (T3)	0.89985	0.60566	1.486	0.13735
tumor stage (T4)	1.55776	0.67308	2.314	0.02065
nodal status (N1)	0.93492	0.35525	2.632	0.00849
nodal status (N2)	1.04473	0.51624	2.024	0.04300
nodal status (N3)	0.65676	0.54747	1.200	0.23029
tumor grade (2)	-0.11584	0.75753	-0.153	0.87847
tumor grade (3)	-1.05566	0.77119	-1.369	0.17104
score	0.04025	0.04290	0.938	0.34821

SuperPC, without adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-1.88089	1.31250	-1.433	0.1518
age	0.01178	0.01531	0.769	0.4416
progesterone receptor status (negative)	0.35258	0.43260	0.815	0.4151
estrogen receptor status (positive)	-0.68927	0.52568	-1.311	0.1898
tumor stage (T2)	0.64012	0.62327	1.027	0.3044
tumor stage (T3)	0.72313	0.67398	1.073	0.2833
tumor stage (T4)	1.46904	0.75243	1.952	0.0509
nodal status (N1)	1.03988	0.38067	2.732	0.0063
nodal status (N2)	1.05348	0.55570	1.896	0.0580
nodal status (N3)	0.87044	0.57684	1.509	0.1313
tumor grade (2)	0.19863	0.80705	0.246	0.8056
tumor grade (3)	-0.22185	0.83007	-0.267	0.7893
score	1.12291	0.21750	5.163	$2.43 \cdot 10^{-7}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-1.78392	1.24261	-1.436	0.15111
age	0.01810	0.01446	1.252	0.21059
progesterone receptor status (negative)	0.39614	0.40499	0.978	0.32799
estrogen receptor status (positive)	0.07913	0.50172	0.158	0.87468
tumor stage (T2)	0.81538	0.57442	1.419	0.15576
tumor stage (T3)	0.97208	0.62445	1.557	0.11954
tumor stage (T4)	1.67004	0.69735	2.395	0.01663
nodal status (N1)	0.92912	0.35780	2.597	0.00941
nodal status (N2)	0.90229	0.51527	1.751	0.07993
nodal status (N3)	0.61652	0.54947	1.122	0.26185
tumor grade (2)	0.05564	0.76551	0.073	0.94206
tumor grade (3)	-0.69890	0.78681	-0.888	0.37439
score	0.44676	0.17773	2.514	0.01195

SuperPC, with adjustment

Without pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-4.31393	1.50687	-2.863	0.00420
Age	0.02196	0.01625	1.352	0.17650
Progesterone receptor status (negative)	0.09498	0.48316	0.197	0.84416
Estrogen receptor status (positive)	0.48004	0.49922	0.962	0.33626
tumor stage (T2)	1.15670	0.69187	1.672	0.09455
tumor stage (T3)	1.23403	0.75091	1.643	0.10031
tumor stage (T4)	2.15727	0.85019	2.537	0.01117
nodal status (N1)	1.29342	0.41750	3.098	0.00195
nodal status (N2)	1.37676	0.59661	2.308	0.02102
nodal status (N3)	1.06972	0.63738	1.678	0.09329
tumor grade (2)	1.02808	0.88670	1.159	0.24627
tumor grade (3)	0.74855	0.91689	0.816	0.41427
score	1.02228	0.15534	6.581	$4.68 \cdot 10^{-11}$

With 5-fold pre-validation

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-1.79182	1.25193	-1.431	0.15236
Age	0.02055	0.01419	1.449	0.14740
Progesterone receptor status (negative)	0.44378	0.40150	1.105	0.26904
Estrogen receptor status (positive)	0.69463	0.42217	1.645	0.09990
tumor stage (T2)	0.79790	0.56402	1.415	0.15716
tumor stage (T3)	0.88973	0.61195	1.454	0.14596
tumor stage (T4)	1.62638	0.68476	2.375	0.01754
nodal status (N1)	0.91158	0.35360	2.578	0.00994
nodal status (N2)	0.96988	0.50968	1.903	0.05705
nodal status (N3)	0.71034	0.54407	1.306	0.19169
tumor grade (2)	0.01445	0.76428	0.019	0.98491
tumor grade (3)	-0.85893	0.79378	-1.082	0.27922
score	0.09555	0.10196	0.937	0.34871

Clinical model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-1.59035	1.22952	-1.293	0.1958
Age	0.02110	0.01415	1.491	0.1359
Progesterone receptor status (negative)	0.46566	0.40164	1.159	0.2463
Estrogen receptor status (positive)	0.77623	0.41506	1.870	0.0615
tumor stage (T2)	0.77686	0.55889	1.390	0.1645
tumor stage (T3)	0.89264	0.60841	1.467	0.1423
tumor stage (T4)	1.58264	0.67711	2.337	0.0194
nodal status (N1)	0.90394	0.35277	2.562	0.0104
nodal status (N2)	0.98365	0.50836	1.935	0.0530
nodal status (N3)	0.70581	0.54508	1.295	0.1954
tumor grade (2)	-0.08394	0.75555	-0.111	0.9115
tumor grade (3)	-1.04042	0.77004	-1.351	0.1767

F Electronic appendix

Declaration of Authorship

I hereby confirm that I have authored this master's thesis independently and without use of others than the indicated resources.

Munich, 12th of May, 2014

Eva-Marie Christina Endres