Using gene expression data to predict dose-response curves

Philippe Haldermans

promotor :
Prof. dr. Ziv SHKEDY, Dr. Willem
TALLOEN

Eindverhandeling voorgedragen tot het bekomen van de graad
Master of Statistics Biostatistics

universiteit hasselt

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The long-term promise that pharmacogenomics (the study of pharmacologically relevant genes) offers is likely to be the ability to stratify patients and diseases based on genotype and to develop better strategies for therapy and prevention based on these stratifications. Such knowledge is useful in the development of novel pharmaceutical products, and hence pharmaceutical industries has embraced genomics and greatly expanded their investment in genomics-based research (Amaratunga and Cabrera, 2004). One promising area of such research studies is oncology, that uses data from microarray data and dose-response curves for cancer patients.

Microarrays allow the monitoring of expression levels of thousands to tens of thousands of genes simultaneously in a given cell type. One can use this microarray data to generate gene expression profiles, which can discriminate between different known cell types or conditions. Many classification methods are developped for this purpose as is shown by Van Sanden *et al.* (2007) .

However, in many situations one is interested in predicting responses rather than classifying them. Potti *et al.* (2006) state that the development of gene expression profiles that can predict response to commonly used cytotoxic agents provides opportunities to better use these drugs, including using them in combination with existing targeted therapies.

Standard existing statistical methods usually deal with situations where the number of subjects (n) is (much) larger than the number of possible variables (p). With microarray data this is clearly not the case. The number of variables, typically in the tens of thousands, is much larger than the number of subjects, typically less than one hundred. Some methods already exist for prediction when the number of features is much larger than the number of subjects as is typical in microarray settings, i.e. Supervised Principal Components Analysis (Bair *et al.*, 2006), Lasso (Tibshrani, 1996) and Elastic Net (Zou and Hastie, 2005). In this thesis, we propose a new method based

on a weighted resampling scheme in combination with existing methods like Lasso and Elastic Net. We will apply this method on an oncology dataset and compare the results to the existing methods.

Chapter 2

Data

The data of this study consist of two major parts. The drug sensitivety data (dose-response) and the gene expression data (microarrays).

2.1 Dose-response data

Dose-response data was collected for a certain oncology compound. The compound was administered to 48 cell lines, each with a number of replicates. For each cell line, doses within a range of 10^{-10} to 10^{-5} were administered and values of the effect (expressed in percentage) were registered as response for each dose-level. Doses are usually converted to the negative log-scale, resulting in a range from 5 to 10. Effect is a (relative) response after calibration to get a biologically meaningful measure with values ranging between approximately -30 and 120.

2.2 Microarray data

The microarray gene expression data was collected on 50 different tumor cell lines each with 30,809 probe sets. For the sake of convenience, we will use the term gene or the more general term feature instead of probe set in this thesis. There were no repetitions, hence one microarray for each cell line.

There were some cell lines included in the microarray data but not in the dose-response study and vice versa. Hence, there are only 42 cell lines common to both the gene expression data and the dose-response data.

Chapter 3

Modelling of the dose-response curves

3.1 Logistic model formulation

In order to summarize the data we will fit a model to it. The model usally used to describe the relationship between the response and the administered dose is a three or four parameters logistic model (Pinheiro and Bates, 2000) :

$$y_{ij} = \frac{\theta_{1i}}{1 + (\frac{\log(d_{ij})}{\theta_{2i}})^{\theta_{3i}}} + \varepsilon_{ij}, \tag{3.1}$$

or

$$y_{ij} = \theta_{0i} + \frac{\theta_{1i}}{1 + (\frac{\log(d_{ij})}{\theta_{2i}})^{\theta_{3i}}} + \varepsilon_{ij}, \tag{3.2}$$

where y_{ij} is the response of cell line i at the jth dose level, d_{ij} is the administered dose and $\varepsilon_i = (\varepsilon_{i1}, ..., \varepsilon_{iJ})$ is the measurement error, which is assumed to be normally distributed with mean zero and covariance matrix Σ.

θ_{0i},θ_{1i}, θ_{2i} and θ_{3i} represent the cell line-specific parameters to be estimated. The parameter θ_{1i} is the maximum effect, θ_{2i} is the log-dose value at which the response is $\theta_{1i}/2$ and θ_{3i} is the slope. The additional fourth parameter θ_{0i} represents a vertical offset.

3.2 Summary measures

θ_{2i} is often referred to as IC50. IC50, or the half maximal inhibitory concentration measures how much of a particular substance/molecule is needed

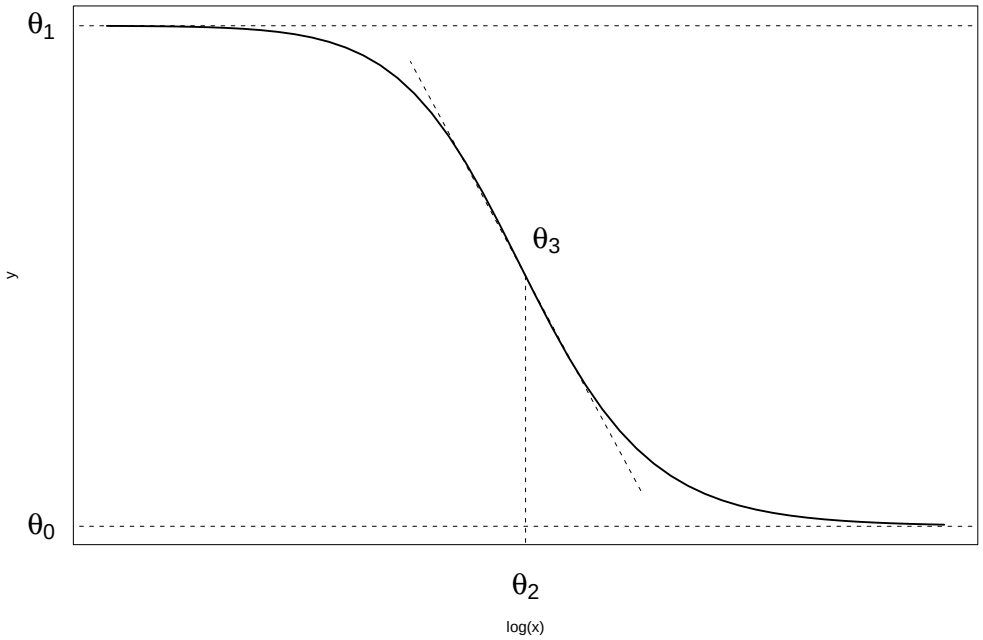


Figure 3.1: Illustration of the four parameter logistic model with a graphical representation of the different variables.

to inhibit some biological process by 50%. According to the FDA, IC50 represents the concentration of a drug that is required for 50% inhibition in vitro.

However, this definition should be handled with care. θ_{2i} corresponds to the dose with response halfway between θ_{0i} and θ_{1i}. In fact, θ_{2i} will only be the dose at 50% response when θ_{0i} is equal to zero and θ_{1i} equal to 100. If one or both of these variables differ from these values θ_{2i} will not correspond to the 50% response dose. When θ_{2i} is used to determine the responsiveness of a cell line, misleading values could be observed. Figure 3.2 illustrates this. In this figure two distinct dose-response curves are shown. The upper curve represent a clear responsive cell line, while the lower curve is less responsive. However, this can not be seen from the θ_{2i} values, since these are equal for both curves.

To overcome this drawback, we suggest a slightly different approach to compute the IC50 values. Instead of taking θ_{2i} from the model, we will compute the value at which the response was 50% by inverting the function. This approach guarantees that we work with the true dose at 50% response.

When θ_{0i} is close to zero and θ_{1i} close to 100, both approaches will yield similar values. However when these variables deviate from these values, the estimations grow apart.

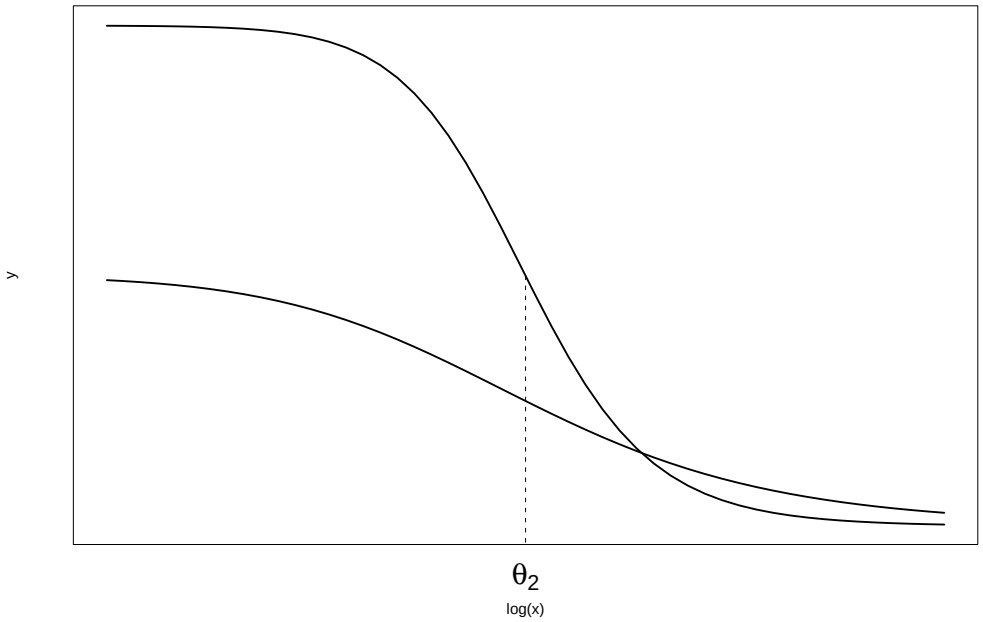


Figure 3.2: Plot of two dose-response curves with distinct patterns but equal IC50 values.

3.3 Univariate screening

In this section, we discuss a joint model for the gene expression and the IC50 score that allows us to identify which gene can serve as a biomarker. Let X_{ij} be the change from baseline of the jth gene expression $j = 1, \ldots, m$, of the ith subject, $i = 1, \ldots, n$, and denote the estimate for the IC50 values of the i'th cell line by θ_i. We define a gene-specific joint model in which the linear predictors of the IC50 scores and the gene expression are given by

$$\begin{array}{rcl} E(X_{ij}) & = & \mu_{X_{ij}}, \quad j = 1, \ldots, m \; ; \;\; i = 1, \ldots, n, \\ E(\theta_i) & = & \mu_{\theta_i}. \end{array} \tag{3.3}$$

Note that (3.3) is a gene-specific model and, in practice, is fitted for each gene separately, a procedure often termed "gene-by-gene" analysis. The pa-

rameters $\mu_{X_{ij}}$ are gene-specific fixed means. It is further assumed that the two outcomes are normally distributed:

$$\begin{pmatrix} X_{ij} \\ \theta_i \end{pmatrix} \sim \mathrm{N}\left(\begin{pmatrix} \mu_{X_{ij}} \\ \mu_{\theta_i} \end{pmatrix}, \Sigma_j = \begin{pmatrix} \sigma_{jj} & \sigma_{j\theta} \\ \sigma_{j\theta} & \sigma_{\theta\theta} \end{pmatrix}\right). \tag{3.4}$$

In the context of surrogate-marker evaluation in randomized clinical trials, Buyse and Molenberghs (1998) proposed the adjusted association as a measure of association, a coefficient derived from the covariance matrix of gene-specific joint model (2):

$$\rho_j = \frac{\sigma_{jY}}{\sqrt{\sigma_{jj}\sigma_{YY}}}. \tag{3.5}$$

Indeed, $\rho_j = 1$ indicates a deterministic relationship between the gene expression and the IC50, in the sense that, given the biomarker (gene expression), a perfect prediction of the IC50 score is possible.

Note that in this setting ρ_j can be equal to 1 even if the gene is not differently expressed. Thus, to select genes which can predict the response, there is no need for the gene to be differentially expressed.

Chapter 4

Prediction methods

4.1 Supervised principal component analysis

Principal component analysis (PCA) is concerned with explaining the variance-covariance structure of the set of variables through a few linear combinations of these variables, its general objective being data reduction and interpretation. For a dataset with large p (number of variables) and small n (number of samples), one wants to reduce the p original variables to k new variables, where k is a compromise between the minimum number of variables and the maximum amount of information kept.

Let $X = (X_1, X_2, ..., X_p)$ be a random vector with variance-covariance matrix Σ and correlation matrix. The ordered eigenvalue-eigenvector pairs for Σ are $(\lambda_j, e_j), j = 1, ..., p$. The principal components are then given by:

$$Z_j = e_j^T X = e_{j1}X_1 + ... + e_{jp}X_p \tag{4.1}$$

where the variance of Z_j is λ_j, and the covariance between any two principal components Z_j and Z_k is zero. Thus, principal components are those uncorrelated linear combinations Z_1, ... , Z_p containing the most information possible, i.e. have the largest variances.

Often only the first (or first few) principal components are used since this linear combination captures the direction of the largest variation in the dataset. However, due to the large amount of noise features, the first principal component might be very noisy, or even consist entirely out of noise.

The idea of supervised principal component is that rather performing PCA using all the genes in the dataset, we use only those that are the strongest correlated to the outcome.

Bair *et al* (2006) propose a method for gene selection based on a certain threshold value of the regression coefficient $\widehat{\gamma}$. In our setting, we quantify the

magnitude of association using an R-square type measure, based on which we select a set of genes. Once genes that are associated with the response have been selected, they will be used for further analysis. For example, Bair *et al* (2006) proposed the supervised principal component analysis to predict the response by the first principal component analysis of the reduced expression matrix. Let $U(\mathbf{X})_\mathbf{i}$ be the first principal component for the reduced expression matrix and consider the following joint distribution for θ_i and $U(\mathbf{X})_\mathbf{i}$:

$$\theta_i = \delta_0 + \delta_1 U(\mathbf{X})_\mathbf{i} + \varepsilon_\mathbf{i}.$$

Note that $U(\mathbf{X})_\mathbf{i}$ is a latent variable.

$$\begin{pmatrix} U(\mathbf{X_i}) \\ \theta_i \end{pmatrix} \sim \mathrm{N}\left(\begin{pmatrix} \mu_{U_i} \\ \mu_{\theta_i} \end{pmatrix}, \Sigma_j = \begin{pmatrix} \sigma_{jj} & \sigma_{j\theta} \\ \sigma_{j\theta} & \sigma_{\theta\theta} \end{pmatrix} \right). \tag{4.2}$$

Bair *et al.* (2006) used a gene specific regression model as a means of selecting a subset of k genes ($k \leq m$) that forms the reduced expression matrix. Hence, within the supervised principal component approach the predictive model is based on the first principal component. However, this might be too restrictive. Perhaps, other genes that are less correlated to the response vector might play an important role in the gene signature as well. Indeed, SPCA can be seen as a gene selection scheme (1,...,1,0,...0) where the first K genes have probability 1 of being selected, while all other genes have a zero probability. In a later section we propose a slightly different approach in which the joint model (4.2) and the conditional model (**??**) are used in order to calculate weights for the genes for weighted resampleing and the predictive model is based on LASSO and elastic net models.

4.2 Lasso

In this section we will discuss a popular method for prediction in a high-dimensional setting called Least Absolute Selection and Shrinkage Operator (Lasso). Tibshirani (1996) proposed minimizing the residual sum of squares, subject to a constraint on the sum of absolute values of the regression coefficients, $\sum_{j=1}^{p} |\beta_j| \leq t$. This is equivalent to minimizing the sums of squares of residuals plus an l_1 penalty on the regression coefficients,

$$\|Y - X\beta\|_2^2 + \theta \sum_{j=1}^{p} |\beta_j|. \tag{4.3}$$

In contrast to ridge regression where all coefficients immediately become nonzero, LASSO coefficients only become nonzero one at a time. Hence the

l_1 penalty results in variable selection, as variables with coefficients of zero are effectively omitted from the model.

Another important difference occurs for the predictors that are most significant. Whereas an $l2$ penalty pushes all coefficients toward zero with a force proportional to the value of the coefficient, an $l1$ penalty exerts the same force on all nonzero coefficients. Hence the most important variables, that clearly should be in the model and where shrinkage toward zero is less desirable, an $l1$ penalty might be more appropriate. This is important for providing accurate predictions of future values.

However, the Lasso method also has it's limitations. The Lasso method can only select a number of features at most equal to the number of samples. Clearly, there will be situations where more features are required.

A second limitation of the Lasso method is it's inability to do grouped selection. When a group of highly correlation features exist, the method will only select one of the features and discard the others.

To overcome these limitations, the elastic net method is introduced.

4.3 Elastic net

The (naive) elastic net criterion is defined as follows for any fixed non-negative λ_1 and λ_2,

$$L(\lambda_1, \lambda_2, \beta) = |y - X\beta|^2 + \lambda_2|\beta|^2 + \lambda_1|\beta|_1, \tag{4.4}$$

where $|\beta|^2 = \sum_{j=1}^{p} \beta_j^2$ and $|\beta|_1 = \sum_{j=1}^{p} |\beta_j|$.

The (naive) elastic net estimator $\hat{\beta}$ is the minimizer of equation (4.4):

$$\hat{\beta} = \arg\min_{\beta} L(\lambda_1, \lambda_2, \beta) \tag{4.5}$$

This procedure can be viewed as a penalized least squares method. Let $\alpha = \lambda_2/\lambda_1 + \lambda 2$ then solving $\hat{\beta}$ in equation (4.4) is equivalent to the optimization problem

$$\hat{\beta} = \arg\min_{\beta} |y - X\beta|^2, \quad \text{subject to} (1-\alpha)|\beta|_1 + \alpha|\beta|^2 \leq t. \tag{4.6}$$

The function $(1-\alpha)|\beta|_1 + \alpha|\beta|^2$ is called the elastic net penalty, which is a convex combination of the lasso and ridge penalty. When $\alpha = 1$, the naive elastic net becomes simple ridge regression and when $\alpha = 0$ we have the lasso method.

The l_1 part of the penalty function ensures the sparsity of the model, while the l_2 part removes the limitations on the number of features that can be used in a model and even encourages the grouping of highly correlated features.

Zou and Hastie (2005) note that the naive elastic net estimator incurs an undesired double amount of shrinkage. To remove this double shrinkage they rescale the naive estimator as follows:

$$\hat{\beta}(\text{elastic net}) = (1 + \lambda_2)\hat{\beta}(\text{naive elastic net}). \tag{4.7}$$

This scaling transformation preserves the variable selection property of the naive elastic net and is the simplest way to undo shrinkage.

4.4 Weighted ensemble prediction

Both the lasso method and the elastic net approach suffer from a decrease in accuracy when the number of feature grow exponentially compared to the number of samples. Since both methods yield good results when the number of features is reasonably small compared to the number of samples (factor 10), it might be a good idea to select a group of important features first and then perform lasso or elastic net on this subset. The practical implementation of this technique will be discussed in this section.

We could now, similar to the supervised principal component analyis, select the top K ranked genes according to their individual correlation to the response. On this subset of K genes we could then apply the method of our choice, Lasso, Elastic net or any other suited method. However, in this ranking we only consider the working of individual genes. As is well known, genes don't usually work alone, but rather there is an interaction between several genes. Therefor it makes sense to investigate which genes have show a close connection to the response with respect to the other genes.

4.4.1 Multivariate screening

In this subsection we will discuss how to screen for genes that are related to the response with respect to other genes. The basis idea behind it comes from the concept of random forests.

A classification tree can be used to decide the classification of new entries, based on the available data. A random forest (Breiman, 2001) is an ensemble of many of such trees, based on a subset of the original data, where each tree is called a base classifier. Classes are assigned to test cases by majority vote: when given a test case, each tree assigns it a class according to its classifier;

this information is collated and overall the forest assigns it the most frequent class. The out-of-bag cases in any tree can be regarded as test cases for that tree as they were not used to build it and thus they can be used to assess the performance of the forest as a whole; this is done via the out-of-bag error rate, which is the proportion of times an out-of-bag case is misclassified. Thus only patterns truly present in the data would be detected consistently by a majority of the base classifiers and the majority votes turn out to be good indicators of class.

When the number of possible features is huge and the percentage of truly informative features is small, a problem arises. The performance of the base classifiers degrades. This is because, if simple random sampling is used for selecting the subset of g eligible features at each node, almost all these subsets are likely to contain many non-informative features. This can be remedied by using weighted random sampling instead of simple random sampling as suggested by Amaratunga *et al.* (2008) .

The key to the modified algorithm is to score each feature based on how well it separates the two groups. Such a score can be generated by testing each feature for a group mean effect using a two-sample t test and calculating the p-value, small p-values indicating greater separation and large p-values indicating less. Once the weights have been determined, the random forest is run with the only modification being that when, at any node, the subset of g eligible features is selected, it is selected using weighted random sampling rather than simple random sampling. This way, highly informative gene have a higher probability to be selected for a tree, guaranteeing that most classification trees are based on informative genes, protecting the performance of the random forest.

In our setting, we will use the R^2 measure between individual features and the response as score statistic. How to use this score statistic for the sampling of subsets will be discussed in the following section.

4.4.2 Weights

What we are looking for is a way to convert the R^2 measures into weights that can be used for the subset selection. Different weight functions can be considered for the sampling of the subsets. Four different function will be considered here. The first and second function are based on p-values (p_i) coming from the correlation between individual genes and the IC50 values. The first function has the form:

$$w_i = \min(\frac{1}{p_i} - 0.99, 999) \tag{4.8}$$

and the second:

$$w_i = \min(-\log(p_i), 999) \tag{4.9}$$

The third function is based on q-values which are calculated from the p-values as: $q_i = min_{k \geq 1} min((G/k)p_k, 1)$, where $p_{(i)}$ and $q_{(i)}$ are the p-value and q-value associated with the feature with the i-th smallest p-value. The q-values provide false discovery rate (FDR)-adjusted measures of significance for the features and are in the same order as the p-values. The corresponding weight function is then given by:

$$w_i = \min(\frac{1}{q_i} - 0.99, 999) \tag{4.10}$$

As a reference function we also consider a weight function where are genes get equal weight of getting sampled. This can be achieved by:

$$w_i = 1 \tag{4.11}$$

4.4.3 Application of the prediction method to the subsets

Once the weights are computed, we can sample a subset of features and apply the preferred prediction method (i.e. Lasso, Elastic net, ...). This process can be repeated many times, each time with a new subset of genes. To avoid overfitting, only 90 % of the subjects will be used in each step. From each of the steps a record is kept of the features that were used in the prediction model. After these steps, a binomial test is used with the number of times a feature was sampled and the number of times a feature was used in the prediction model to see which features appeared above average times in the prediction models.

The features that are in this final subset are thus judged to be important and form the final subset on which the prediction method is used one last time to obtain the final prediction model.

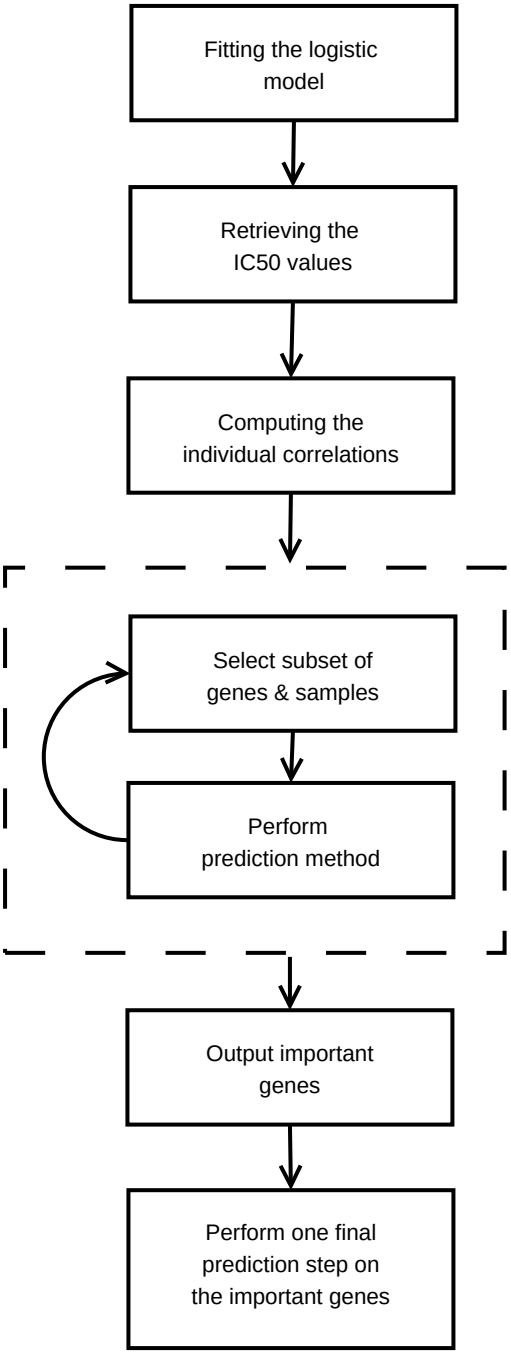


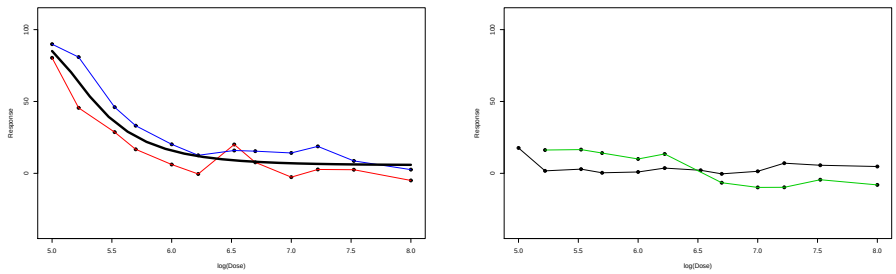
Figure 4.1: Flowchart of the steps involved in the Weighted Ensemble Prediction method.

Chapter 5

Results

5.1 Modelling of the dose-response curves

We will apply the logistic model described in the methodology to the provided dataset. Since several cell lines start at values below 0, the four-parameter logistic model seems the more appropriate choice. The fitting itself was done with the R-package `mrdrc` developped by Ritz *et al.* (2008). The package provides a self-starting function for the estimation of the logistic model, which removes the burden of providing start values.

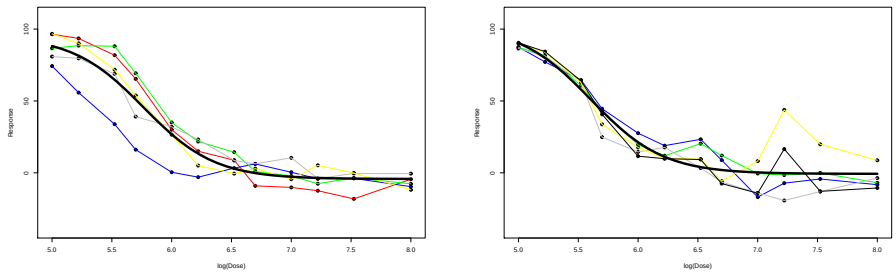


(a) Included curves (b) Excluded curves

Figure 5.1: Plot of the curves of the first subject. The left panel shows the curves that were used to model the dose-response curve. The solid black line represents the fitted curve. The right panel shows the two curves that were removed from the model fit.

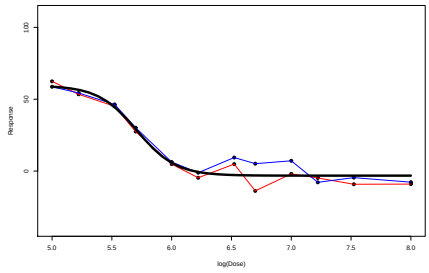
As indicated by the people that provided the data, the setup of the experiments was as follows. A few dose-response curves were made per cell line. If the curves of this cell lines were all similar the experiment was complete. However, when there was a deviating pattern detected, a few more

curves were made until it became clear what the dominant pattern was. To avoid possible influence of these deviating curves, they were removed from the analysis. An example of deviated curves that were removed from the data is shown in Figure 5.1.

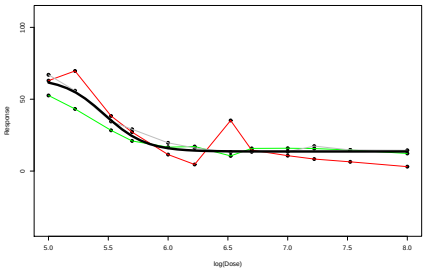


(a) Curve 2

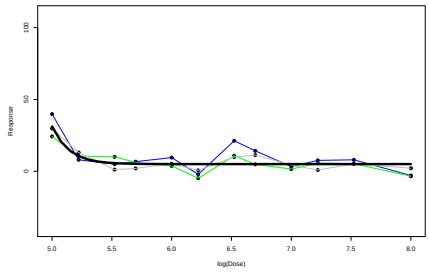
(b) Curve 5



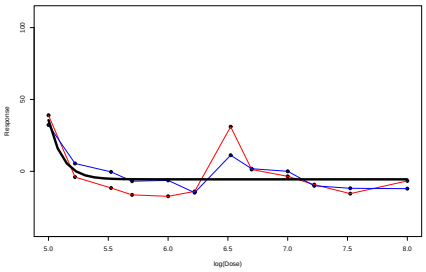
(c) Curve 9



(d) Curve 21



(e) Curve 38



(f) Curve 47

Figure 5.2: Plots of some of the dose-response curves. The top row shows clear responsive curves, the middle row show slow responding curves and the bottom row consists of non responsive curves. The solid line represents the fitted logistic model.

As is illustrated in Figure 5.2, different patterns can be observed. Some cell lines can clearly be categorized as either responsive or non-responsive. These are the cell lines that are used in classification studies. For example,

19 of these cell lines where used in a study where the goal was to classify the cell lines using microarray data. Obviously a lot of information is lost in this study. Not only is the information for these 19 cell lines greatly reduced, but even more severe is the fact that all other cell lines are discarded all together. Figure 5.3 shows the plot of the IC50 values of the 19 responsive and non-responsive cell lines. It can be seen that all cell lines that were catergorized as non-responsive indeed all have lower IC50 values compared to the responsive cell lines. This confirms the validity of the model.

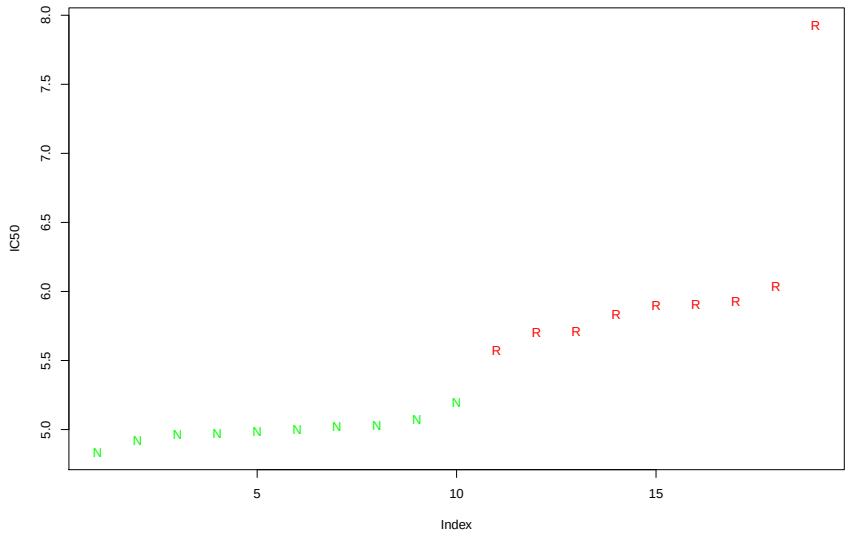


Figure 5.3: Plot of the IC50 values of the different cell lines. "R" indicates the responders, "N" the non responders.

5.2 Univariate screening for important genes

Once the IC50 values are computed, we can proceed to see which genes have the highest correlation and are the most promising genes with respect to predicting the IC50 values. Figure 5.4 shows a histogram of the p-values corresponding to the individual correlations. When there is no signal in the data, we expect to see uniformly distributed p-values. Luckily, this is not the case here, we can clearly see a peak at the smal p-values, indicating that there might be a gene signature in the data.

Figure 5.5 shows plots of the 9 genes with the highest correlation with respect to the IC50 values. Immediately one sees that these correlations

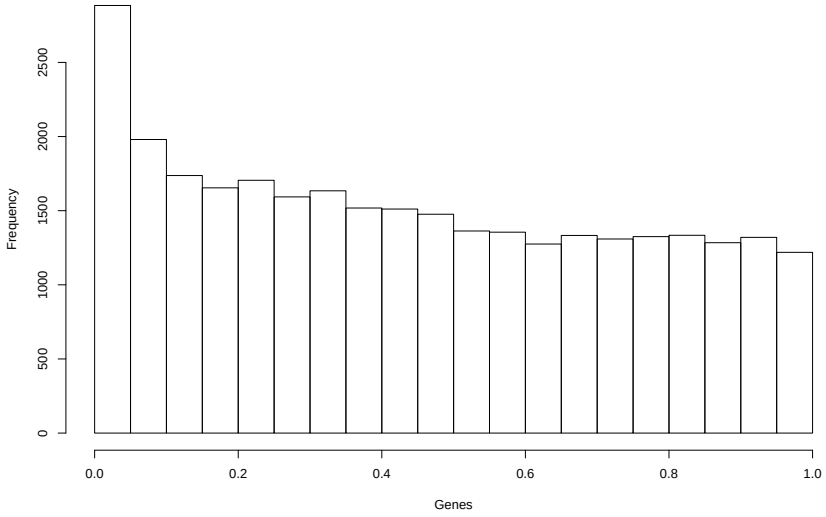


Figure 5.4: Histogram of the p-values corresponding to the individual correlations between the genes and the response.

are greatly driven by one particular outlier, i.e. cell line 29. One could worry that this might negatively affect the selection of important genes for the prediction model. However, the weigthed ensemble prediction method seems not affected by this situation. Anaysis was done with and without the cell line and using more robust correlation measures, but results did not change significantly, with exception of the analysis where the Spearman rank correlation was used, which gave worse results.

This might be explained by the fact that due to the repeated sampling in the weighted ensemble method important genes are still selected often enough in the different subsets even if they are dominated by some less important genes. This also illustrates the importance of not only looking at individual correlations only, but also to groups of genes, since results of individual correlation might be misleading as such.

5.3 Leave-one-out cross-validation

In order to evaluate the prediction methods, we use a leave-one-out cross validation scheme. This means that we use an outer loop in which we exclude one subject each time, which we will try to predict using only the other subjects. This way we can objectively compare the new method with other

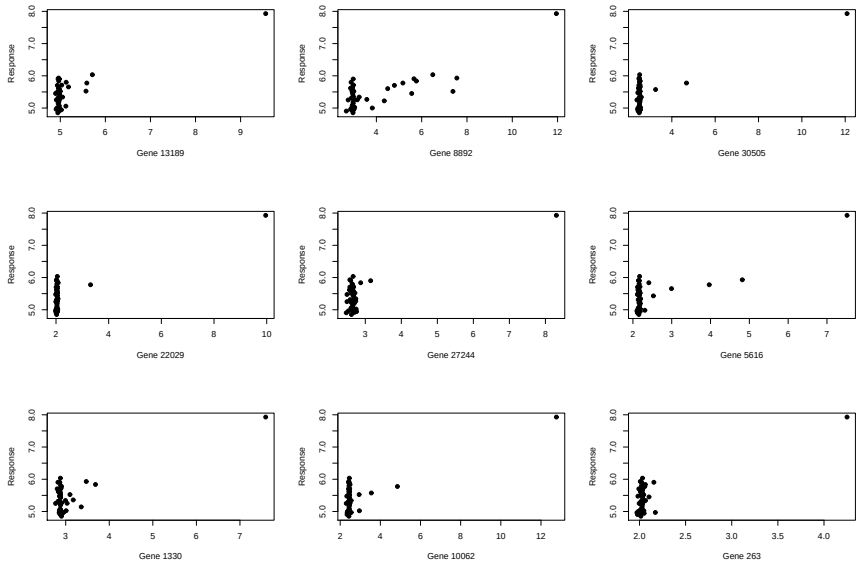


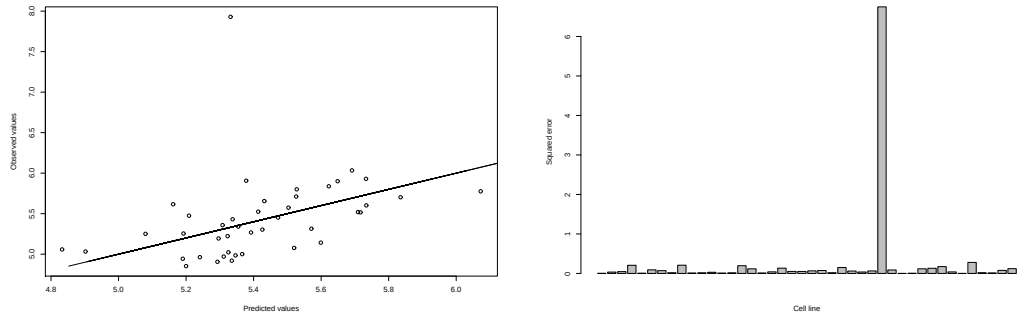
Figure 5.5: Plot of the 9 genes with the highest correlation with respect to the response.

methods and compare between different setting.

In the basic setting we use standard Pearson correlation for the univariate screening, a loop of 500 lasso steps, using each time 90 % of the subjects with one final lasso step at the end. As result we will rapport both the correlation and mean squared error (MSE) between the observed IC50 values and the ones predicted by the method. The lasso step were implemented with the R-package `glmnet` developed by Friedman *et al.* (2009).

Figure 5.6(a) gives the plot of the observed versus the predicted IC50 values, while Figure 5.6(b) shows the individual squared contributions to the total error. Immediately becomes clear that all possible results will be clouded by the one outlying cell line that we also mentionned in the previous section. Since we are interested in an overall idea how good the IC50 values are predicted rather than only the one, we will from now on rapport correlation and mean squared error based on all the cell lines except the outlying one. Note that does not mean that we remove this cell line from the dataset, only that we focus for results on the other cell lines.

Applying the weighted ensemble prediction method with the basic settings described above, we get a correlation of 0.621 and a MSE of 0.070.



(a) LOO predicted values vs observed values (b) Barplot of squared errors

Figure 5.6: Comparison between observed and predicted values. Panel (a) shows a plot of predicted versus observed values. The solid line represents the y=x line. Panel (b) gives a barplot of the squared errors for the individual cell lines.

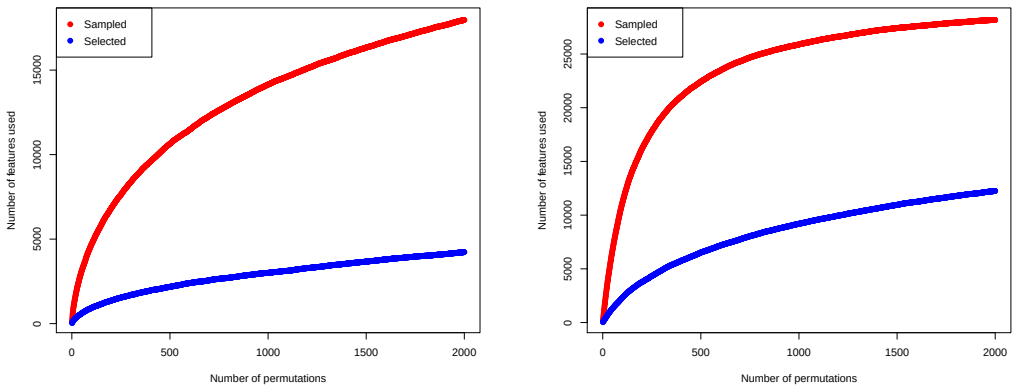
5.4 Weight functions

In this section we will investigate the effect of the different weight function discussed in Section 4.4.2. Figure 5.7 shows the number of genes that are sampled and selected by the four respective weight functions. The red line gives the total number of genes that are sampled by that particular weight function. As can be seen in panel (a), the least number of genes are sampled with weights defines as in (4.8). Even after 1000 permutations, only 50% of the available number of genes are selected. Consequently, a small number of genes receive high weights, and hence have a high chance of getting sampled. Most genes have low weights and will only be sampled infrequently.

The other weight functions yield less outspoken weights, meaning that the probability of getting sampled is spread over more genes, in stead of a few genes that get (very) high probabilities. Weights from (4.11) are the extreme case here, since all genes get the same probability of getting sampled.

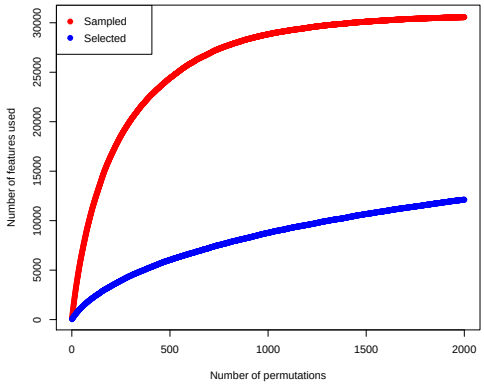
In every permutation, the Lasso method is applied to the sampled genes. The resulting set of genes used by the Lasso method to fit the model, are indicated here as the selected genes. These are shown on the respective plots as blue lines. As can be expected, the basic setting uses the smallest number of different genes in the resulting sets, while this number is the highest for the weight function with equal weights for all genes.

Table 5.1 shows us that there is not much difference between the results for the first two weight functions. One could conclude that the important choice here is to choose to use the p-values, rather than the choice of the

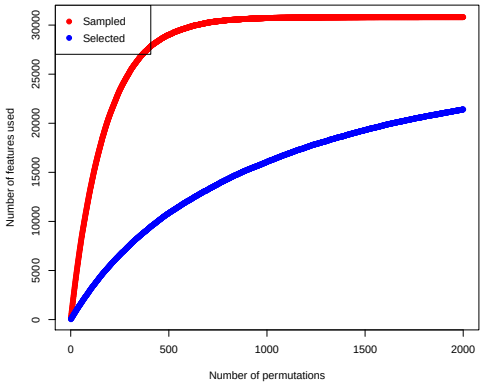


(a) Basic weights

(b) Log weights



(c) Qval weights



(d) Random weights

Figure 5.7: Plots of the number of sampled and selection genes for each of the weight functions.

Table 5.1: L-o-o correlation and MSE for the different weight functions.

	Corr	MSE
Basic weight function	0.609	0.073
Log p-values	0.586	0.075
Weights based on q-values	0.307	0.128
Equal weights	0.276	0.128

conversion function.

Differences come when we move to the q-values and equal weights, which show clear lower values for both correlation and MSE. The only real difference was noticed for the weight function with equal weights. This function yields quite lower values for both correlation and MSE. In the case of the q-values, the q-values tend to keep only very few gene with a higher weights, while all other receive the same baseline weight. This means that we get close to the situation with all equal weights.

Equal weights in its turn implies that all genes are only selected a few times, including the important genes and thus separation of important and unimportant genes is very difficult. This might be solved by increasing the number of lasso steps inside the loop. We tried this for an increasing number of step with results shown in Table 5.2. We see a slight improvement for a higher number of steps, but still more steps are needed to approach the results of the first two weight functions. This becomes of course too time consuming, indicating the importance of the choice for appropriate weights.

Table 5.2: L-o-o correlation and MSE for equal weights function for increasing number of permutations.

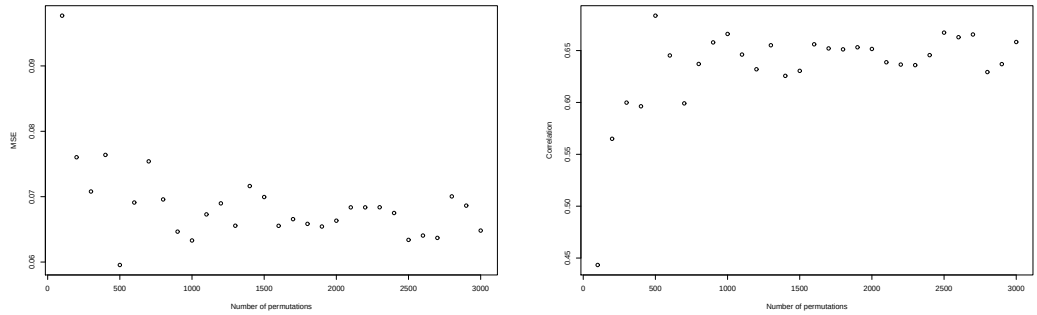
	Corr	MSE
500 perms	0.276	0.128
1000 perms	0.391	0.160
1500 perms	0.374	0.120
2000 perms	0.493	0.106

5.5 Number of step in the loop

In the previous section we showed that increasing the number of permutations for the case where all weights where equal improved the predictions in terms of both correlation and MSE. A logical question thus is, what happens to the correlation and MSE for the basic setting when we increase the number of permutations.

The result for values ranging from 100 to 3000 is shown in Figure 5.8. Only minor changes are seen, especially when looking at values starting from 500 upwards. The minor changes are caused by the inherent randomness of the sampling of both features and subjects inside the loop. Since more step

is equal to a longer computation time, we tend to choose a lower number of steps, i.e. something in the range of 500 to 1000.



(a) MSE (b) Correlation

Figure 5.8: Plot of the MSE and Correlation in function of the number of permutations.

5.6 Influence of the correlation inside/outside the loop

Thusfar we compute the individual correlations once and converted the p-values of these correlations to the weights which we use for the sampling of features in the loop. Since we use only 90 % of the subjects each step of the loop, the top features with respect to correlation to the response could differ from those that would come out on top when we look at the subset of subjects in each step seperately. Therefor we suggest to recompute the correlation each step of the loop, to obtain each time the optimal weigths for that particular subjects.

Table 5.3 shows the comparison between the situations where the individual correlations are computed once outside the loop or each time inside the loop respectively. As can be seen, the prediction improves considerably when correlation is recomputed each time, so this will become our default strategy for the remainder of the analysis.

5.7 Weighted Ensemble Elastic Net

In this section we will investigate how result could be affected by the prediction method used. As was discussed in Section 4.3, the Elastic Net can be

Table 5.3: L-o-o correlation and MSE for correlation out-/inside the Lasso loop.

	Corr	MSE
One correlation	0.609	0.073
Correlation in each step	0.713	0.055

seen as a generalization of the Lasso method. In a similar way we could see Weigthed Ensemble Elastic Net as a generalization of Weighted Ensemble Lasso. We investigated the effect of a change in the parameter alpha on the results. Alpha equal to 1 gives us the original Lasso method alpha equal to zero stands for ridge regression, while values in between 0 and 1 can be associated with a combination of both methods.

As can be seen from Figure 5.9 the best results are obtained with values of alpha close to 1. When alpha goes down towards zero the correlation drops and MSE rises. Similar to regular elastic net, a lower value of alpha stands for methods closely related to ridge regression and hence more non-zero coefficients. The evolution of the number of non-zero coefficients from ridge regression towards lasso is given in Figure 5.9(c).

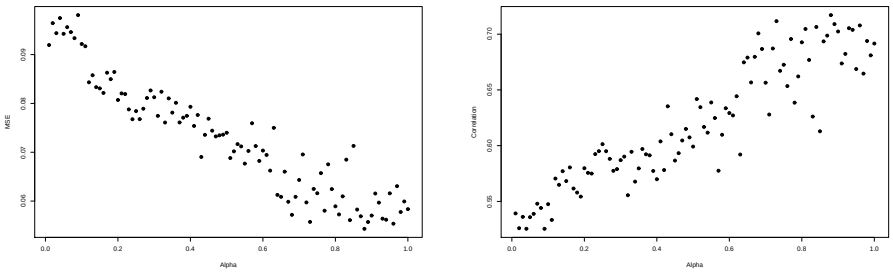
5.8 Comparison to other methods

In a final step we will compare the weigthed ensemble prediction method to the popular Supervised Principal Component Analysis and regular Lasso method. Note that the Lasso steps were implemented using the R-package `lars` developed by Hastie *et al.* (2007) in stead of `glmnet`. This was due to the memory issues that occurred when the dimension of the feature matrix exceeded 15000 features.

As can be seen in Table 5.4, the Weighted Ensemble Prediction outperforms both SPCA and Lasso.

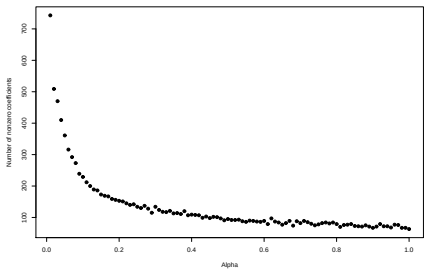
Table 5.4: L-o-o correlation and MSE for different prediction methods.

	Corr	MSE
SPCA	0.510	0.082
Lasso	0.670	0.067
Weighted Ensemble Prediction	0.713	0.055



(a) MSE

(b) Correlation



(c) Nonzero coefficients

Figure 5.9: Plots of respectively the MSE and correlation in function of the parameter alpha.

When we compare for example the Weighted Ensemble Lasso versus regular Lasso, it becomes clear that feature selection before applying the Lasso method can improve the accuracy of the prediction. This could be explained by the fact that the important genes are not swamped in a sea of noise features, which makes it easier for the method to fit the correct genes.

From the comparison with the SPCA method we learn that only univariate screening might not be sufficient to filter out the important genes. It seems plausible that it is more important to look at which set of genes shows a good connection to the response rather than to limit the search for individual genes.

Chapter 6

Discussion and conclusions

DNA microarrays were developed as a mean of monitoring thousands of genes at once. Nowadays, the study of microarray data is becoming an important research area in pharmaceutical industries for the discovery and development of novel pharmaceutical products.

The goal of this study was to find a gene signature that could accurately predict the dose-response relationship. In order to quantify the dose-response relationship we fitted a four-parameter logistic model and obtained the corresponding IC50 values.

Several existing linear prediction methods could be considered including Supervised Principal Component Analysis and Lasso. A possible drawback of these methods is the fact that the important genes might be masked by the vast number of noisy uninformative genes. However, as can be seen from the results coming from the leave-one-out cross-validation set-up, results are remarkably good. This might indicate that the present gene signature is strong and can be picked up even with noise present. In the remainder of the analysis, we tried to improve results with a Weighted Ensemble approach.

The rationale behind the Weighted Ensemble method was to reduce the number of genes before performing the actual prediction method. The goal was to reduce the number of noisy uninformative genes, so the method could accurately detect the requested gene signature. Many schemes could be used to do feature selection, ranging from discarding features whose range falls below a certain threshold, removing features that display little to no variation among cell lines to selecting features based on their individual correlation with the response. However, all these selection methods don't take into account the interaction of groups of genes that might be activated together, i.e. gene signatures.

Weighted Ensemble Lasso proved this view by further improving the accuracy of the Lasso model, resulting in a lower MSE and higher correlation.

In this example, moving from Lasso towards Ridge Regression as prediction method did not improve the resuls, indicating that Lasso was the optimal choice for this dataset.

Thusfar, we only considered combinations of either Lasso or Elastic Net as prediction method in both the feature selection step and the final prediction step. One could extend this concept by using one prediction method for feature selection and another for the final step. Or one might think of different prediction methods altogether.

Bibliography

Alonso, A. and Molenberghs, G. (2007). Surrogate marker evaluation from an information theory perspective. *Biometrics*, 63:180–186.

Amaratunga, D., Cabrera, J., and Lee, Y.-S. (2008). Enriched random forests. *Bioinformatics*, 24(18):2010–2014.

Bair, E., Hastie, T., Debashis, P., and Tibshirani, R. (2006). Prediction by supervised principal components. *American Statistics Association*, 101(473):119–137.

Breiman, L. (2001). Random forests. *Machine learning*, 45:5–32.

Buyse, M. and Molenberghs, G. (1998). Criteria for the validation of surrogate endpoints in randomized experiments. *Biometrics*, 54:1014–1029.

Friedman, J., Hastie, T., and Tibshirani, R. (2009). *glmnet: Lasso and elastic-net regularized generalized linear models*. R package version 1.1-3.

Hastie, T. and Efron, B. (2007). *lars: Least Angle Regression, Lasso and Forward Stagewise*. R package version 0.9-7.

Pinheiro, J. and Bates, D. (2000). *Mixed-effects models in S and S-PLUS*. Springer.

Potti, A., Dressman, H., Bild, A., Riedel, R., Chan, G., Sayer, R., Cragun, J., Cottrill, H., Kelley, M., Petersen, R., Harpole, D., Marks, J., Berchuck, A., Ginsburg, G., Febbo, P., Lancaster, J., and Nevins, J. (2006). Genomic signatures to guide the use of chemotherapeutics. *Nature Medicine*, 12(11):1294–300.

Ritz, C., Tarp-Johansen, M. J., and Martinussen, T. (2008). *Model-robust concentration-response analysis*. R package version 1.0-2.

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

Van Sanden, S., Lin, D., and Burzykowski, T. (2007). Performance of classification methods in a microarray setting: A simulation study. *Biocybernetics and Biomedical Engineering*, 27:15–28.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B*, 67:301–320.

Auteursrechtelijke overeenkomst

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:
Using gene expression data to predict dose-response curves

Richting: **Master of Statistics-Biostatistics**
Jaar: **2010**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

Haldermans, Philippe

Datum: **29/01/2010**