Aus dem
Deutschen Krebsforschungszentrum in Heidelberg
Abteilung für Biostatistik
(Abteilungsleiterin: Prof. Dr. Annette Kopp-Schneider)

Robust Biclustering of Highdimensional Molecular Data by Sparse Singular Value Decomposition Incorporating Stability Selection

Inauguraldissertation
zur Erlangung des Doctor scientiarum humanarum (Dr.sc.hum.)
an der
Medizinischen Fakultät Heidelberg
der
Ruprecht-Karls-Universität

vorgelegt von

Martin Sill

aus
Bremerhaven
2011

Dekan: Prof. Dr. Claus R. Bartram
Doktormutter: Prof. Dr. Annette Kopp-Schneider

Contents

List of Tables

List of Figures

List of Abbreviations

BIC	Bayesian Information Criterion
CGH	Comparative Genomic Hybridization
ISA	Iterative Signature Algorithm
NMF	Non-negative Matrix Factorization
PM	Plaid Model
SNP	Single Nucleotid Polymorphism
SVD	Singular Value Decomposition
SSVD	Sparse Singular Value Decomposition

1 Introduction

1.1 The microarray technology

Since Mendel Darwin -¿ Mendel -¿ DNA -¿ Helix -¿ PCR -¿ Sanger Sequenzierung -¿ Microarrays expression protein chip on chip sage cgh snp -¿ qPCR -¿ RefSeq Highthroughput -¿ measure epigenome transcriptome proteome genome highdimensionl molecular data. the curse of dimensionality development of statistical methods -¿ classification supervised differentially expression clustering in clinical cancer research also clinical data integration by means of interactive graphics

1.2 Clustering

Clustering methods belong to the most commonly used statistical tools in the analysis of high dimensional data sets. If additional information about the sample class labels is lacking, other types of analysis like supervised classification methods or testing for differentially expressed genes can not be performed. In this case unsupervised clustering allows to reveal unknown structures that are possibly hidden in the gene expression data matrix. These structures may be characterized by groups of genes that are coregulated by a common transcription factor and thus belong to the same pathway or samples that share a similar gene expression pattern. One disadvantage of commonly used clustering algorithms like hierarchical clustering or k-means clustering is that the cluster assignment of objects are based on the complete feature space, e.g. in case of clustering the samples the resulting clusters are derived with respect to all genes. But groups of genes may only be coregulated within a subset of the samples and samples may share a common gene expression pattern only for a subset of genes. Such clusters that exist only in a subspace of the feature space can hardly be detected by these classical one-way clustering algorithms. To find such clusters other clustering concepts are needed.

1.3 Biclustering

In the past decade, the concept of biclustering has emerged in the field of gene expression analysis. Biclustering which is also known as coclustering or two-way clustering describes the simultaneous clustering of the rows and the columns of a data matrix. The first biclustering algorithm, the so called Block Clustering, has been developed by Hartigan (1972). Cheng and M. (2000) proposed the first biclustering algorithm for the analysis of high dimensional gene expression data. Since then, many different biclustering algorithms have been developed. Currently, their exists a diverse spectrum of biclustering tools that follow different strategies and algorithmic concepts. Among others, popular algorithms are the Coupled Two-Way Clustering (CTWC) by Getz et al. (2000), Order Preserving Sub Matrix (OPSM) algorithm by Ben-Dor et al. (2003), FLOC by Yang et al. (2003), Spectral biclustering by Kluger et al. (2003), xMotif by Kasif et al. (2003), the Iterative Signature Algorithm (ISA) by Bergmann et al. (2003), the Plaid Model by Lazzeroni and Owen (2000) and the improved Plaid Model (Turner et al. 2005), SAMBA by Tanay et al. (2004), biclustering by non-smooth non-negative matrix factorization by Carmona-Saez et al. (2006), the Bi-correlation clustering algorithm (BCCA) by Bhattacharya and K De (2009) and factor analysis for bicluster acquisition (FABIA)(Hochreiter et al. 2010). Prelic et al. (2006) developed a fast divide-and-conquer algorithm (Bimax) and conducted a systematic comparison of different biclustering algorithms. Santamaría et al. (2007) published an article on validation indices for the evaluation of biclustering results and the comparison for biclustering algorithms. Comprehensive reviews about the concept of biclustering and the different biclustering approaches have been written by Madeira and Oliveira (2004) and Mechelen et al. (2004).

In a more theoretical review Busygin et al. (2008) emphasized the mathematical concepts behind several biclustering algorithms and pointed out that the SVD represents a capable tool for finding biclusters. Furthermore, most existing biclustering algorithms use the SVD directly or have a strong association with it. To keep track of the huge diversity, regarding the mathematical properties of the existing biclustering algorithms, Busygin et al. (2008) suggest to relate new and existing biclustering algorithms to the SVD.

A major drawback of many biclustering methods is that they rely on random starting seeds and thus are inconsistent and results may vary even when the algorithm is applied

to the same data set. As often in unsupervised clustering it is difficult to judge the biclustering results regarding their stability. For one-way clustering several resampling approaches to validate the stability of the clustering results are known, e.g. multiscale bootstrap hierarchical clustering (Suzuki and Shimodaira 2006) and consensus clustering (Monti et al. 2003). In case of biclustering, similar methods that take the stability of the results into account are not yet available.

1.4 Objectives

Currently their exist a huge diversity of biclustering algorithms. That follow different

1. Development of a Biclustering approach that takes the stability of the resulting clusters into account.
2. Integrative explorative data

2 Material and Methods

This chapter outlines the development of a new biclustering method that aims to identify biclusters in high-dimensional microarray data taking the stability of the clustering result into account. The chapter is organized in four parts. In the first part, the sparse singular value decomposition (SSVD) proposed by Lee et al. (2010) and the stability selection by Meinshausen and Bühlmann (2010) is described. Then, the new developed biclustering method, the S4VD algorithm, which is a combination of these two approaches is introduced. Furthermore, the R-package *s4vd* which provides the S4VD algorithm and additional visualization functions is presented. In the second part, the design of the simulation study that compares the S4VD algorithm with other biclustering methods is illustrated. In this context, validation indices for the evaluation of the simulation results are described. In order to demonstrate the practical application of the S4VD algorithm, the third part delineates the evaluation of three microarray data sets. In the final part, the interactive visualization software SEURAT (Gribov et al. 2010) is presented. SEURAT is an open source software tool which provides interactive visualization capability for the integrated analysis of high-dimensional microarray data together with associated clinical and genomical data. Besides other clustering algorithms SEURAT offers several biclustering methods including the S4VD algorithm.

2.1 The S4VD Algorithm

Recently, Lee et al. (2010) proposed a sparse SVD method to find biclusters in high-dimensional gene expression data. Singular vectors of an SVD are interpreted as regression coefficients of a linear regression model. The SSVD algorithm alternately fits penalized regression models to the singular vector pair to obtain a sparse matrix decomposition. The sparseness of the resulting singular vectors depends on the choice of the penalization parameter. In this thesis, we propose to choose the penalization parameters by stability selection (Meinshausen and Bühlmann 2010). Stability selec-

tion is a subsampling procedure that can be applied to penalized regression models to select stable variables. In addition, stability selection offers the possibility to control type-one error rates (Dudoit et al. 2003), e.g. the per-family error rate (PFER) or the per-comparison wise error rate (PCER).

2.1.1 SVD and Biclustering

Let $\mathbf{X} = (x_{ij}) \in \mathbb{R}^{p \times n}$ be the gene expression matrix with indices $i = 1, \cdots, p$ and $j = 1, \cdots, n$. The number of genes p is usually by multiple greater than the number of samples n. The SVD of $\mathbf{X}$ can be written as:

$$\mathbf{X} \approx \mathbf{U}\mathbf{D}\mathbf{V}^T = \sum_{k=1}^{r} d_k \mathbf{u}_k \mathbf{v}_k^T, \tag{2.1}$$

where r is the rank of $\mathbf{X}$ and the columns of the matrix $\mathbf{U} = (\mathbf{u}_1, \cdots, \mathbf{u}_r)$ are the orthonormal left-singular vectors and the columns of $\mathbf{V} = (\mathbf{v}_1, \cdots, \mathbf{v}_r)$ are the orthonormal right-singular vectors. The elements of the diagonal matrix $\mathbf{D}$ are the corresponding positive singular values $d_1 \geq d_2 \geq \cdots d_r > 0$. Thus the SVD is the sum of rank one matrices $d_k \mathbf{u}_k \mathbf{v}_k^T$, herein after also called SVD-layers. According to Busygin et al. (2008) biclustering can be related to the SVD by considering an idealized data matrix. This matrix has a block diagonal structure where each block represents a bicluster and the elements outside these blocks are equal to zero:

$$\mathbf{X} = \begin{pmatrix} \mathbf{X}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{X}_2 & 0 & \cdots \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & \mathbf{X}_r \end{pmatrix}, \tag{2.2}$$

where $\mathbf{X}_k$, $k = 1, \cdots, r$ are submatrices of $\mathbf{X}$. If we decompose $\mathbf{X}$ by the SVD, then each submatrix $\mathbf{X}_k$ will be associated with a singular vector pair $(\mathbf{u}_k, \mathbf{v}_k)$ such that the non-zero coefficients in $\mathbf{u}_k$ represent the rows that belong to $\mathbf{X}_k$ and the non-zero coefficients $\mathbf{v}_k$ represent the columns that belong to $\mathbf{X}_k$. In the presence of noise and if the data matrix has no block diagonal structure, the SVD will still be able to detect the rows and columns of the submatrices as the prominent coefficients in the singular vector pair. These properties make the SVD a practical method for biclustering.

2.1.2 The SSVD Algorithm

In the following the SSVD method proposed by (Lee et al. 2010) is described. The idea is to interpret the singular vectors of a regular SVD as regression coefficients of a linear regression and use sparsity-inducing penalties to obtain sparse singular vector pairs. According to Eckart and Young (1936) the first SVD-layer gives us the best rank-one approximation of $\mathbf{X}$ with respect to the squared Frobenius norm, i.e.

$$(d_1, \mathbf{u}_1, \mathbf{v}_1) = \arg\min_{d,\mathbf{u},\mathbf{v}} \left\| \mathbf{X} - d\mathbf{u}\mathbf{v}^T \right\|_F^2, \tag{2.3}$$

where $\|\cdot\|_F^2$ indicates the squared Frobenius norm, which is the sum of squared elements of the matrix. Lee et al. (2010) showed how this rank-one approximation can be related to linear regression. Suppose $\mathbf{u}_1$ is fixed, then the minimization of (2.3) with respect to $(d_1, \mathbf{v}_1)$ is equivalent to a minimization with respect to $\tilde{\mathbf{v}}_1 = (d_1\mathbf{v}_1)$. Accordingly, the loss function can be written as minimization of the squared l^2-norm:

$$\left\| \mathbf{X} - \mathbf{u}_1\tilde{\mathbf{v}}_1^T \right\|_F^2 = \left\| \mathbf{y} - (I_n \otimes \mathbf{u}_1)\tilde{\mathbf{v}}_1 \right\|, \tag{2.4}$$

where $\mathbf{y} = (\mathbf{x}_1^T, \cdots, \mathbf{x}_n^T)^T \in \mathbb{R}^{pn}$ with $\mathbf{x}_j$ being the jth column of $\mathbf{X}$. Then the minimization of (2.4) can be interpreted as least squares problem with $\mathbf{y}$ as the response vector, $I_n \otimes \mathbf{u}_1$ as the design matrix and the $\tilde{\mathbf{v}}_1$ as vector of regression coefficients. The least squares estimator of $\tilde{\mathbf{v}}_1$ is:

$$\begin{aligned} \hat{\tilde{\mathbf{v}}}_1 = \left\{ (I_n \otimes \mathbf{u}_1)^T (I_n \otimes \mathbf{u}_1) \right\}^{-1} (I_n \otimes \mathbf{u}_1)^T \mathbf{y} = \\ (\mathbf{u}_1^T\mathbf{x}_1, \cdots, \mathbf{u}_1^T\mathbf{x}_n)^T = \mathbf{X}^T\mathbf{u_1}. \end{aligned} \tag{2.5}$$

In the same way we can derive the least squares estimator for the product of the first left singular vector multiplied with the first singular value $\tilde{\mathbf{u}}_1$. So without loss of generality with $\mathbf{v}_1$ fixed the minimization of (2.3) with respect to $\tilde{\mathbf{u}}_1 = (d_1\mathbf{u}_1)$ is given by the minimization of:

$$\left\| \mathbf{X} - \tilde{\mathbf{u}}_1\mathbf{v}_1^T \right\|_F^2 = \left\| \mathbf{z} - (I_n \otimes \mathbf{v}_1)\tilde{\mathbf{u}}_1 \right\|, \tag{2.6}$$

where $\mathbf{z} = (\mathbf{x}_1, \cdots, \mathbf{x}_p)^T \in \mathbb{R}^{pn}$ with $\mathbf{x}_i^T$ being the ith row of $\mathbf{X}$. Here $\mathbf{z}$ is the response vector and $(I_n \otimes \mathbf{v}_1)$ is the design matrix.

Finally, the least squares estimator of $\tilde{\mathbf{u}}_1$ is given by:

$$\begin{aligned}\hat{\tilde{\mathbf{u}}}_1 = \left\{(I_n \otimes \mathbf{v}_1)^T (I_n \otimes \mathbf{v}_1)\right\}^{-1} (I_n \otimes \mathbf{v}_1)^T \mathbf{z} = \\ (\mathbf{x}_1^T \mathbf{v}_1, \cdots, \mathbf{x}_p^T \mathbf{v}_1) = \mathbf{X}\mathbf{v_1}.\end{aligned} \tag{2.7}$$

In order to obtain sparse singular vector pairs, Lee et al. (2010) suggest to find the first SVD-layer that minimizes the Frobenius norm subject to sparsity-inducing penalty terms $P_1(d_1\mathbf{u}_1)$ and $P_2(d_1\mathbf{v}_1)$:

$$\left\|\mathbf{X} - d_1\mathbf{u}_1\mathbf{v}_1^T\right\|_F^2 + \lambda_{\mathbf{u}_1} P_1(d_1\mathbf{u}_1) + \lambda_{\mathbf{v}_1} P_2(d_1\mathbf{v}_1), \tag{2.8}$$

where $\lambda_{\mathbf{u}_1}$ and $\lambda_{\mathbf{v}_1}$ are tuning parameters. Possible penalty functions are the adaptive lasso penalties (Zou 2006). The corresponding penalized function is given by:

$$P_1(d_1\mathbf{u_1}) = d_1 \sum_{i=1}^{p} w_{1,i}|u_i|, \quad P_2(d_1\mathbf{v_1}) = d_1 \sum_{j=1}^{n} w_{2,j}|v_j|, \tag{2.9}$$

where $w_{1,i}$ and $w_{2,j}$ are weights that can be chosen according to Zou (2006), e.g. for $w_{1,i} = w_{2,j} = 1$ we obtain the lasso penalty. Thus the penalty functions are weighted sums of the absolute values of the elements of the first singular vector pair. Fixing $\mathbf{u}_1$ and using the adaptive lasso penalty the minimization of (2.8) becomes:

$$\begin{aligned}\left\|\mathbf{X} - d_1\mathbf{u}_1\mathbf{v}_1^T\right\|_F^2 + \lambda_{\mathbf{v}} \sum_{j=1}^{n} w_{2,j}|v_j| = \\ \|\mathbf{X}\|_F^2 + \sum_{j=1}^{n} \left\{\tilde{v}_j^2 - 2\tilde{v}_j(\mathbf{X}^T\mathbf{u}_1)_j + \lambda_{\mathbf{v}} w_{2,j}|\tilde{v}_j|\right\}.\end{aligned} \tag{2.10}$$

To solve this penalized regression and estimate the sparse right singular vector, Lee et al. (2010) proposed an algorithm that incorporates a simple component-wise thresholding rule. The component-wise minimizer of (2.10) is:

$$\hat{\tilde{v}}_{1,j} = \textit{sign}\left\{(\mathbf{X}^T\mathbf{u}_1)_j\right\} \left(|(\mathbf{X}^T\mathbf{u}_1)_j| - \lambda_{\mathbf{v}} w_{2,j}/2\right)_+. \tag{2.11}$$

This is the well known soft threshold estimator proposed by Tibshirani (1996). Then $\hat{\tilde{\mathbf{v}}}_{\mathbf{1}} = (\hat{\tilde{v}}_{1,1}, \cdots, \hat{\tilde{v}}_{1,n})^T$, is an estimate for the product of the first right singular vector multiplied with the first singular vector. In order to get an estimate for the first sparse right singular vector we have to update the first singular value. The first update

of d_1 is $d_{1,\mathbf{v}_1} = \left\|\hat{\tilde{\mathbf{v}}}_\mathbf{1}\right\|$ and accordingly the estimated sparse singular vector becomes $\hat{\mathbf{v}}_1 = \hat{\tilde{\mathbf{v}}}_\mathbf{1}/d_{1,\mathbf{v}_1}$. The penalized regression for the left singular vector can be solved in the same way. For fixed $\mathbf{v}_1$ and with the adaptive lasso penalty the loss function of (2.8) becomes:

$$\begin{aligned} \left\|\mathbf{X} - d_1\mathbf{u}_1\mathbf{v}_1^T\right\|_F^2 + \lambda_\mathbf{u}\sum_{i=1}^{p} w_{1,i}|u_i| = \\ \|\mathbf{X}\|_F^2 + \sum_{i=1}^{p}\left\{\tilde{u}_i^2 - 2\tilde{u}_i(\mathbf{X}\mathbf{v}_1)_i + \lambda_\mathbf{u} w_{1,i}|\tilde{u}_i|\right\}. \end{aligned} \tag{2.12}$$

The component-wise minimizer of (2.12) is:

$$\hat{\tilde{u}}_{1,i} = \textit{sign}\left\{(\mathbf{X}\mathbf{v}_1)_i\right\}\left(|(\mathbf{X}\mathbf{v}_1)_i| - \lambda_\mathbf{u} w_{1,i}/2\right)_+. \tag{2.13}$$

The updated singular value is $d_{1,\mathbf{u}_1} = \|\tilde{\mathbf{u}}_\mathbf{1}\|$, with $\hat{\tilde{\mathbf{u}}}_\mathbf{1} = (\hat{\tilde{u}}_{1,1}, \cdots, \hat{\tilde{u}}_{1,p})^T$. Finally, the estimated sparse left singular vector is $\hat{\mathbf{u}}_1 = \hat{\tilde{\mathbf{u}}}_\mathbf{1}/d_{1,\mathbf{u}_1}$.

The *degree of sparsity*, which is defined as the number of non-zero coefficients in the singular vector pair, depends on the choice of the penalty parameters. Lee et al. (2010) proposed to choose the optimal *degree of sparsity* by computing the complete penalization path and apply the penalty parameter that minimizes the Bayesian information criterion (BIC) (Schwarz 1978). In case of the penalized regression model estimating the right singular vector (2.10) the BIC is:

$$\text{BIC}(\lambda_{\mathbf{v}_1}) = \frac{\|\mathbf{z} - \hat{\mathbf{z}}\|^2}{np\hat{\sigma}^2} + \frac{log(np)}{np}\hat{df}(\lambda_{\mathbf{v}_1}), \tag{2.14}$$

where $\hat{df}(\lambda_{\mathbf{v}_1})$ is the *degree of sparsity* and $\hat{\sigma}$ is the least squares estimate of the error variance of the regression model. For the penalized regression model estimating the left singular vector (2.12) the BIC is:

$$\text{BIC}(\lambda_{\mathbf{u}_1}) = \frac{\|\mathbf{y} - \hat{\mathbf{y}}\|^2}{np\hat{\sigma}^2} + \frac{log(np)}{np}\hat{df}(\lambda_{\mathbf{u}_1}). \tag{2.15}$$

In the SSVD algorithm the two regressions with the corresponding parameter tuning are alternated until convergence is reached, which is if either $\|\mathbf{v}_1 - \hat{\mathbf{v}}_1\| < \epsilon$ or $\|\mathbf{u}_1 - \hat{\mathbf{u}}_1\| < \epsilon$, where $\epsilon > 0$ is an arbitrary convergence threshold. After convergence

the final estimate of the first singular value of the sparse SVD-layer is $\hat{d}_1 = \hat{\mathbf{u}}_1^T \mathbf{X} \hat{\mathbf{v}}_1$. The next sparse rank-one approximation can be obtained by subtracting the sparse SVD-layer and applying the SSVD method to the residual matrix $\mathbf{X} - \hat{d}_1 \hat{\mathbf{u}}_1 \hat{\mathbf{v}}_1^T$.

The SSVD Algorithm

---

1. Apply the standard SVD to $\mathbf{X}$. Let $\{d_1, \mathbf{u}_1, \mathbf{v}_1\}$ denote the first SVD triplet.
2. Update:
   a) Set $\hat{\tilde{u}}_{1,i} = \textit{sign}\{(\mathbf{X}\mathbf{v_1})_i\}\,(|(\mathbf{X}\mathbf{v_1})_i| - \lambda_{\mathbf{u}} w_{1,i}/2)_+$, where λ_u minimizes the *BIC*. Let $\hat{\tilde{\mathbf{u}}}_{\mathbf{1}} = (\hat{\tilde{u}}_{1,1}, \cdots, \hat{\tilde{u}}_{1,p})^T$, $d_{1,\mathbf{u}_1} = \left\|\hat{\tilde{\mathbf{u}}}_{\mathbf{1}}\right\|$, and $\hat{\mathbf{u}}_{\mathbf{1}} = \hat{\tilde{\mathbf{u}}}_{\mathbf{1}}/d_{1,\mathbf{u}_1}$.
   b) Set $\hat{\tilde{v}}_{1,j} = \textit{sign}\left\{(\mathbf{X}^T\hat{\mathbf{u}}_1)_j\right\}(|(\mathbf{X}^T\hat{\mathbf{u}}_1)_j| - \lambda_{\mathbf{v}_1} w_{2,j}/2)_+$, where λ_v minimizes the *BIC*. Let $\hat{\tilde{\mathbf{v}}}_{\mathbf{1}} = (\hat{\tilde{v}}_{1,1}, \cdots, \hat{\tilde{v}}_{1,n})^T$, $d_{1,\mathbf{v}_1} = \left\|\hat{\tilde{\mathbf{v}}}_{\mathbf{1}}\right\|$, and $\hat{\mathbf{v}}_{\mathbf{1}} = \hat{\tilde{\mathbf{v}}}/d_{1,\mathbf{v}_1}$.
   c) Set $\mathbf{v}_1 = \hat{\mathbf{v}}_1$, $\mathbf{u}_1 = \hat{\mathbf{u}}_1$ and repeat 2(a) and 2(b) until convergence.
3. Set $\hat{d}_1 = \hat{\mathbf{u}}_1^T \mathbf{X} \hat{\mathbf{v}}_1$.
4. To obtain the next layer apply steps 1 to 3 to the residual matrix $\mathbf{X} - \hat{d}_1 \hat{\mathbf{u}}_1 \hat{\mathbf{v}}_1^T$.

In practice we observed that choosing the regularization parameters according to the BIC results in singular vector pairs with a relative low *degree of sparsity*. In addition, the SSVD algorithm does not offer a stopping criterion and so the choice of the number of SVD-layers is arbitrary.

2.1.3 Stability Selection

In this thesis, we propose to choose the penalization parameters and to control the *degree of sparsity* of the resulting SVD-layers using stability selection (Meinshausen and Bühlmann 2010). The idea of stability selection is to combine resampling with variable selection methods, e.g. penalized regression models. For each variable its probability of being selected is estimated by resampling the data and calculating relative frequencies of being selected. Meinshausen and Bühlmann (2010) provide a theoretical framework

for controlling type-one error rates of falsely selecting variables based on the maximum of these selection probabilities over the range of regularization parameters.
Supposing interest lies in the inference of the true set of non-zero coefficients in the left singular vector $S_{\mathbf{u}_1} = \{i : u_i \neq 0\}$. The set of possible penalization parameters that can be applied within the adaptive lasso regression is $\Lambda_{\mathbf{u}_1}$. Each $\lambda_{\mathbf{u}_1} \in \Lambda_{\mathbf{u}_1}$ leads to a different estimated subset of indices of non-zero coefficients $\hat{S}_{\mathbf{u}_1}^{\lambda_{\mathbf{u}_1}} \subseteq \{1, \cdots, p\}$. Meinshausen and Bühlmann (2010) illustrate the stability selection with the so-called stability paths that show the selection probabilities of each coefficient along the range of penalization parameters. Given any $\lambda_{\mathbf{u}_1}$ the estimated set $\hat{S}_{\mathbf{u}_1}^{\lambda_{\mathbf{u}_1}}$ can be written as a function of the samples $J = \{1, \cdots, n\}$, e.g. $\hat{S}_{\mathbf{u}_1}^{\lambda_{\mathbf{u}_1}} = \hat{S}_{\mathbf{u}_1}^{\lambda_{\mathbf{u}_1}}(J)$. If $J^* \subset J$ is a subsample drawn without replacement, then the estimated selection probability is:

$$\hat{\Pi}_i^{\lambda_{\mathbf{u}_1}} = P(i \in \hat{S}_{\mathbf{u}_1}^{\lambda_{\mathbf{u}_1}}(J^*)). \tag{2.16}$$

The selection probability can be estimated by calculating the relative selection frequencies of i with regard to all subsamples. Given an arbitrary threshold $\pi_{thr} \in (0.5, 1)$ and the set of penalization parameters $\Lambda_{\mathbf{u}_1}$, the set of non-zero coefficients estimated with the stability selection is:

$$\hat{S}_{\mathbf{u}_1}^{stable} = \left\{ i : \max_{\lambda_{\mathbf{u}_1} \in \Lambda_{\mathbf{u}_1}} \hat{\Pi}_i^{\lambda_{\mathbf{u}_1}} \geq \pi_{thr} \right\}. \tag{2.17}$$

According to Meinshausen and Bühlmann (2010) the value of π_{thr} has a neglectible influence and they recommend to choose values in the range of $[0.6, 0.9]$. Let $\hat{S}^{\Lambda_{\mathbf{u}_1}} = \cup_{\lambda_{\mathbf{u}_1} \in \Lambda_{\mathbf{u}_1}} \hat{S}^{\lambda_{\mathbf{u}_1}}$ be the union of the estimated sets of selected coefficients with regard to all $\lambda_{\mathbf{u}_1} \in \Lambda_{\mathbf{u}_1}$. Then the average number of selected coefficients is $q_{\Lambda_{\mathbf{u}_1}} = E(|\hat{S}^{\Lambda_{\mathbf{u}_1}}(J^*)|)$. Let $N_{\mathbf{u}_1}$ denote the set of zero coefficients, then the number of falsely selected coefficients with stability selection is given by $V_{\mathbf{u}_1} = |N_{\mathbf{u}_1} \cap \hat{S}_{\mathbf{u}_1}^{stable}|$. Following Theorem 1 in Meinshausen and Bühlmann (2010) the expected number of falsely selected coefficients is bounded by:

$$E(V_{\mathbf{u}_1}) \leq \frac{1}{(2\pi_{thr} - 1)} \frac{q_{\Lambda_{\mathbf{u}_1}}^2}{p}. \tag{2.18}$$

Interpreting equation (2.18) the expected number of falsely selected coefficients decreases by either reducing the average number of selected coefficients $q_{\Lambda_{\mathbf{u}_1}}$ or by increasing the threshold π_{thr}. Supposing that π_{thr} is fixed the stability selection controls the desired error level of $E(V_{\mathbf{u}_1})$ as long as the average number of selected coefficients

is less then $e_{\Lambda_{\mathbf{u}_1}}$, where $e_{\Lambda_{\mathbf{u}_1}} = \sqrt{E(V_{\mathbf{u}_1})p(2\pi_{thr}-1)}$ is an upper bound for the average number of selected coefficients that can be controlled by reducing the length of the regularization path $\Lambda_{\mathbf{u}_1}$. In multiple testing the expected number of falsely selected variables is also known as the per-family error rate (PFER) and if divided by the total number of the variables it will become the per-comparison error rate (PCER) (Dudoit et al. 2003). The stability selection allows to control these type-one error rates.

2.1.4 The SSVD Algorithm with nested Stability Selection

Here we propose to replace the BIC based penalty parameter selection of the SSVD algorithm by the stability selection. This combined approach allows to control the expected number of falsely selected non-zero coefficients in the singular vector pair and therefore the *degree of sparsity* of the resulting SVD-layers. Furthermore, the error control also serves as stopping criterion for the improved SSVD algorithm and determines the number of reasonable layers.

We aim to estimate the left singular vector $\hat{\mathbf{u}}_1$ and at the same time infer the true set of non-zero coefficients $S_{\mathbf{u}_1}$. For each possible $\lambda_{\mathbf{u}_1}$ we draw subsamples and estimate the selection probabilities $\hat{\Pi}_i^{\lambda_{\mathbf{u}_1}}$. Given a threshold π_{thr} and the desired type-one error $E(V_{\mathbf{u}_1})$, the regularization region $\Lambda_{\mathbf{u}_1}$ is defined so that $q_{\Lambda_{\mathbf{u}_1}} \leq e_{\Lambda_{\mathbf{u}_1}}$. Then the estimated set of non-zero coefficients is:

$$\hat{S}_{\mathbf{u}_1}^{stable} = \left\{ i : \max_{\lambda_{\mathbf{u}_1} \in \Lambda_{\mathbf{u}_1}} \hat{\Pi}_i^{\lambda_{\mathbf{u}_1}} \geq \pi_{thr} \right\} \tag{2.19}$$

To estimate $\hat{\tilde{\mathbf{u}}}_{\mathbf{1}}$ we apply the component-wise minimizer of Lee et al. (2010) with the smallest penalization value of the regularization path $\lambda_{\mathbf{u}_1}^{min}$.

$$\hat{\tilde{u}}_{1,i} = \textit{sign}\left\{(\mathbf{X}\mathbf{v}_1)_i\right\}\left(|(\mathbf{X}\mathbf{v}_1)_i| - \lambda_{\mathbf{u}_1}^{min} w_{1,i}/2\right)_+ \tag{2.20}$$

Like in the original SSVD approach, the first update of the singular value is $d_{1,\mathbf{u}_1} = \left\|\hat{\tilde{\mathbf{u}}}_{\mathbf{1}}\right\|$, with $\hat{\tilde{\mathbf{u}}}_{\mathbf{1}} = (\hat{\tilde{u}}_{1,1}, \cdots, \hat{\tilde{u}}_{1,n})^T$. The estimated sparse singular vector is $\hat{\mathbf{u}}_{\mathbf{1}} = \hat{\tilde{\mathbf{u}}}_{\mathbf{1}}/d_{1,\mathbf{u}_1}$. Without loss of generality we estimate the sparse right singular vector $\hat{\mathbf{v}}_1$ and infer the respective set of non-zero coefficients $S_{\mathbf{v}_1}$. The selection probabilities $\hat{\Pi}_j^{\lambda_{\mathbf{v}_1}}$ for each $\lambda_{\mathbf{v}_1}$ are estimated by drawing subsets of the genes $I^* \subset I$, where $I = \{1, \cdots, p\}$. Again, given the desired type-one error $E(V_{\mathbf{v}_1})$ and the threshold π_{thr} the regularization region is delimited such that $q_{\Lambda_{\mathbf{v}_1}} \leq e_{\Lambda_{\mathbf{v}_1}}$, where $e_{\Lambda_{\mathbf{v}_1}} = \sqrt{E(V_{\mathbf{v}_1})n(2\pi_{thr}-1)}$.

Consequently, the estimated set of non-zero coefficients in the right singular vector is:

$$\hat{S}^{stable}_{\mathbf{v}_1} = \left\{ j : \max_{\lambda_{\mathbf{v}_1} \in \Lambda_{\mathbf{v}_1}} \hat{\Pi}^{\lambda_{\mathbf{v}_1}}_j \geq \pi_{thr} \right\} \tag{2.21}$$

Given the smallest parameter of the penalization path $\lambda^{min}_{\mathbf{v}_1}$ the components of $\tilde{\mathbf{v}}_\mathbf{1}$ are:

$$\hat{\tilde{v}}_{1,j} = \textit{sign}\left\{ (\mathbf{X}^T\mathbf{u}_1)_j \right\} \left(|(\mathbf{X}^T\mathbf{u}_1)_j| - \lambda^{min}_{\mathbf{v}_1} w_{2,j}/2 \right)_+ \tag{2.22}$$

Finally let $\hat{\tilde{\mathbf{v}}}_\mathbf{1} = (\hat{\tilde{v}}_{1,1}, \cdots, \hat{\tilde{v}}_{1,n})^T$, the updated first singular value is $d_{1,\mathbf{v}_1} = \|\tilde{\mathbf{v}}_\mathbf{1}\|$ and estimated sparse singular vector is $\hat{\mathbf{v}} = \hat{\tilde{\mathbf{v}}}_\mathbf{1}/d_{1,\mathbf{v}_1}$.

These two penalized regression models with the nested stability selection are alternated until convergence, e.g. that is if either $\|\mathbf{v}_1 - \hat{\mathbf{v}}_1\| < \epsilon$ or $\|\mathbf{u}_1 - \hat{\mathbf{u}}_1\| < \epsilon$, where $\epsilon > 0$. After convergence the estimated singular value is $\hat{d}_1 = \hat{\mathbf{u}}_1^T\mathbf{X}\hat{\mathbf{v}}_1$ and finally those coefficients that are not in the two sets of stable coefficients $\hat{S}^{stable}_{\mathbf{u}_1}$ and $\hat{S}^{stable}_{\mathbf{v}_1}$ are set to zero. So the components of $\hat{\mathbf{u}}_1$ become $\hat{u}_{1,i} = \mathbf{1}(i \in \hat{S}^{stable}_{\mathbf{u}_1})\hat{u}_{1,i}$ and the components of $\hat{\mathbf{v}}_1$ become $\hat{v}_{1,j} = \mathbf{1}(j \in \hat{S}^{stable}_{\mathbf{v}_1})\hat{v}_{1,j}$, where $\mathbf{1}(\cdot)$ is an indicator function. The next sparse rank-one approximation can be obtained by subtracting the sparse SVD-layer and applying the S4VD method to the residual matrix. Alternatively non-overlapping biclusters can be detected by excluding either the rows or the columns (or both) that correspond to the non-zero coefficients in the singular vector pair and and apply the S4VD method to the submatrix. By incorporating the stability selection a stopping criterion can be defined. If in any iteration an estimated set of non-zero coefficients is an empty set, the sequential fitting of sparse rank-one layers will be interrupted.

The S4VD Algorithm

1. Apply the standard SVD to $\mathbf{X}$. Let $\{d_1, \mathbf{u}_1, \mathbf{v}_1\}$ denote the first SVD triplet. Choose the desired type-one errors $E(V_{\mathbf{v}_1})$ and $E(V_{\mathbf{u}_1})$ and the threshold π_{thr}.

2. Update:

   a) For each $\lambda_{\mathbf{u}_1}$ draw subsamples J^* and estimate $\hat{\Pi}_i^{\lambda_{\mathbf{u}_1}}$. Define $\Lambda_{\mathbf{u}_1}$ such that $q_{\Lambda_{\mathbf{u}_1}} \leq e_{\Lambda_{\mathbf{u}_1}}$ and estimate the set of non-zero coefficients $\hat{S}_{\mathbf{u}_1}^{stable}$.
   Set $\hat{\tilde{u}}_{1,i} = \textit{sign}\left\{(\mathbf{X}\mathbf{v}_1)_i\right\}\left(|(\mathbf{X}\mathbf{v}_1)_i| - \lambda_{\mathbf{u}_1}^{min} w_{1,i}/2\right)_+$
   Let $\hat{\tilde{\mathbf{u}}}_{\mathbf{1}} = (\hat{\tilde{u}}_{1,1}, \cdots, \hat{\tilde{u}}_{1,p})^T$, $d_{1,\mathbf{u}_1} = \left\|\hat{\tilde{\mathbf{u}}}_{\mathbf{1}}\right\|$, and $\hat{\mathbf{u}}_{\mathbf{1}} = \hat{\tilde{\mathbf{u}}}/d_{1,\mathbf{u}_1}$

   b) For each $\lambda_{\mathbf{v}_1}$ draw subsamples I^* and estimate $\hat{\Pi}_j^{\lambda_{\mathbf{v}_1}}$. Define $\Lambda_{\mathbf{v}_1}$ such that $q_{\Lambda_{\mathbf{v}_1}} \leq e_{\Lambda_{\mathbf{v}_1}}$ and estimate the set of non-zero coefficients $\hat{S}_{\mathbf{v}_1}^{stable}$.
   Set $\hat{\tilde{v}}_{1,j} = \textit{sign}\left\{(\mathbf{X}^T\hat{\mathbf{u}}_1)_j\right\}\left(|(\mathbf{X}^T\hat{\mathbf{u}}_1)_j| - \lambda_{\mathbf{v}_1}^{min} w_{2,j}/2\right)_+$
   Let $\hat{\tilde{\mathbf{v}}}_{\mathbf{1}} = (\hat{\tilde{v}}_{1,1}, \cdots, \hat{\tilde{v}}_{1,n})^T$, $d_{1,\mathbf{v}_1} = \left\|\hat{\tilde{\mathbf{v}}}_1\right\|$, and $\hat{\mathbf{v}}_{\mathbf{1}} = \hat{\tilde{\mathbf{v}}}/d_{1,\mathbf{v}_1}$

   c) Set $\mathbf{v}_1 = \hat{\mathbf{v}}_1$, $\mathbf{u}_1 = \hat{\mathbf{u}}_1$ and repeat 2(a) and 2(b) until convergence.

3. After convergence set $\hat{d}_1 = \hat{\mathbf{u}}_1^T\mathbf{X}\hat{\mathbf{v}}_1$.
   The components of $\hat{\mathbf{u}}_1$ become $\hat{u}_{1,i} = \mathbf{1}(i \in \hat{S}_{\mathbf{u}_1}^{stable})\hat{u}_{1,i}$.
   The components of $\hat{\mathbf{v}}_1$ become $\hat{v}_{1,j} = \mathbf{1}(j \in \hat{S}_{\mathbf{v}_1}^{stable})\hat{v}_{1,j}$.

4. To obtain the next layer apply steps 1 to 3 to the residual matrix $\mathbf{X} - \hat{d}_1\hat{\mathbf{u}}_1\hat{\mathbf{v}}_1^T$.

5. Stop steps 1 to 4 if either $\hat{S}_{\mathbf{v}_1}^{stable} = \emptyset$ or $\hat{S}_{\mathbf{u}_1}^{stable} = \emptyset$.

The subsampling steps of the stability selection makes the S4VD algorithm computationally very demanding. To reduce the computation time, we implemented the pointwise error control suggested by Meinshausen and Bühlmann (2010). Details about the implementation of the S4VD algorithm are described in the supplementary material.

2.1.5 Pointwise error control

In each iteration of the proposed S4VD algorithm we perform two stability selections where for a stability selection the stability path is computed by subsampling for each possible penalization parameter. Thus the S4VD algorithm is computationally very demanding, especially for high dimensional data sets. To reduce the computation time, we implemented the pointwise error control suggested by Meinshausen and Bühlmann (2010). Suppose we are interested in estimating $\hat{\mathbf{u}}_\mathbf{1}$, we can define a single penalization parameter as penalization region $\Lambda_{\mathbf{u}_1} = \{\lambda_{\mathbf{u}_1}\}$ and draw subsamples J^* to calculate the average number of selected coefficients $q_{\Lambda_{\mathbf{u}_1}}$. Given this parameters the stability selection threshold can be calculated:

$$\pi_{thr} = \frac{1}{2}\left(\frac{q^2_{\Lambda_{\mathbf{u}_1}}}{E(V_{\mathbf{u}_1})p} + 1\right) \tag{2.23}$$

We define a region for the threshold $[\pi^{min}_{thr}, \pi^{max}_{thr}]$, e.g. [0.6,0.65], and implemented a simple search algorithm that seeks for a $\lambda_{\mathbf{u}_1}$ such that $\pi^{min}_{thr} \leq \pi_{thr} \leq \pi^{max}_{thr}$. So instead of calculating in each iteration the complete stability paths, this simple algorithm can be applied to find appropriate penalization parameters. In addition the penalization parameter can be used as starting value in the next iteration. This two changes reduce the computation time of the S4VD algorithm remarkably. To illustrate the idea of the stability selection and the pointwise error control, Figure 1 shows an example of the stability paths of the rows and the columns that correspond to a bicluster.

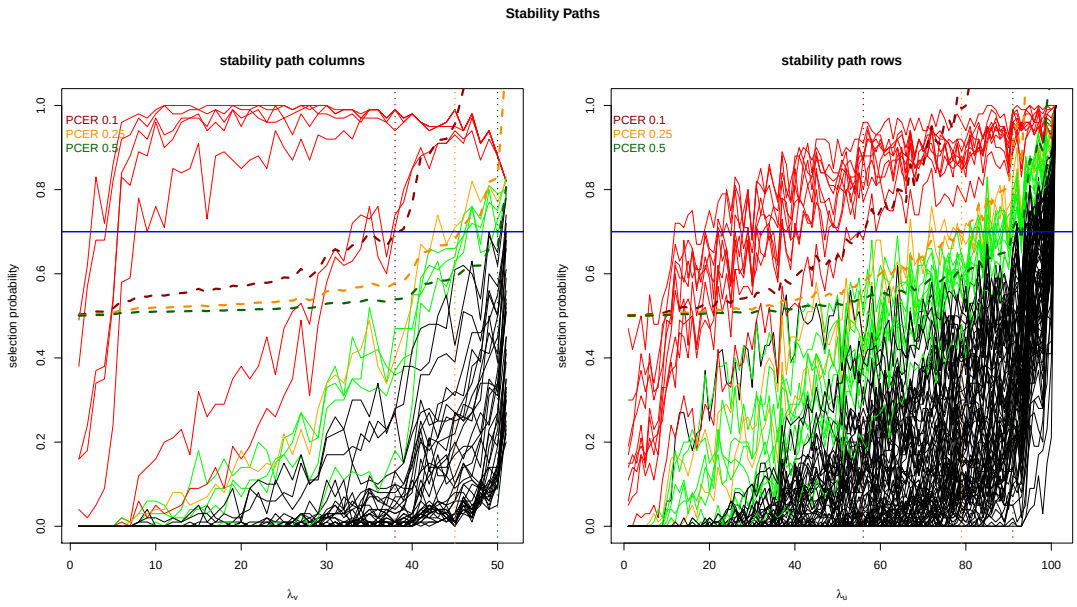

Figure 2.1 Stability paths for the rows and the columns corresponding to a bicluster identified with the S4VD algorithm. The dashed colored lines correspond to the stability selection threshold according to the pointwise error control. The continuous colored lines represent the estimated set of stable rows and columns with respect to different type one-error levels. The blue horizontal line corresponds to a stability selection threshold of 0.7.

2.1.6 The s4vd R-package

2.2 Simulation Study

2.2.1 Validation indices

2.2.1.1 Average relevance and recovery scores

Since a bicluster is defined by its row and column subsets, we apply the Jaccard index to the Cartesian product of the sets of row and column indices. If I^{G_1} and J^{G_1} are the subsets of row and column indices that define the first bicluster G_1 and I^{G_2} and J^{G_2} are respective subsets according to the second bicluster G_2, then the Jaccard index measuring the agreement between those two biclusters is:

$$Jac(G_1, G_2) = \frac{(I^{G_1} \times J^{G_1}) \cup (I^{G_2} \times J^{G_2})}{(I^{G_1} \times J^{G_1}) \cap (I^{G_2} \times J^{G_2})}$$

Suppose we have a biclustering result that corresponds to a set of biclusters $\mathbf{G} = \{G_1, \cdots, G_L\}$ with indices $l = 1, \cdots, L$ and we aim to compare this with a second biclustering result $\mathbf{F} = \{F_1, \cdots, F_M\}$ with indices $m = 1, \cdots, M$. We can summarize the pairwise Jaccard indices between these two bicluster sets with a match score:

$$M(\mathbf{G}, \mathbf{F}) = \frac{1}{L} \sum_{l=1}^{L} \max_{F_m \in \mathbf{F}} Jac(G_l, F_m) \tag{2.24}$$

If $\mathbf{G}$ is a bicluster set proposed by any biclustering algorithm and $\mathbf{F}$ is the artificial set of bicluster present in the data matrix, then $M(\mathbf{G}, \mathbf{F})$ measures the average relevance of the proposed biclusters with respect to the maximal Jaccard index between these biclusters and the artificial biclusters. On the contrary, $M(\mathbf{F}, \mathbf{G})$ measures the average recovery of the artificial biclusters by the proposed bicluster set.

2.2.1.2 Average proportions of falsely assigned rows and columns

If $\mathbf{F} = \{F_1, \cdots, F_M\}$ is the artificial set of biclusters in the data and $\mathbf{G} = \{G_1, \cdots, G_L\}$ is the bicluster set proposed by any biclustering algorithm, then the average number of falsely selected rows by this biclustering algorithm is:

$$V_I(\mathbf{G}, \mathbf{F}) = \frac{1}{L} \sum_{l=1}^{L} \min_{F_m \in \mathbf{F}} |I^{G_l} \notin I^{F_m}| \tag{2.25}$$

and the average number of falsely selected columns is:

$$V_J(\mathbf{G}, \mathbf{F}) = \frac{1}{L} \sum_{l=1}^{L} \min_{F_m \in \mathbf{F}} |J^{G_l} \notin J^{F_m}| \tag{2.26}$$

Then the average proportion of falsely assigned rows and columns are $V_I(\mathbf{G}, \mathbf{F})/p$ and $V_J(\mathbf{G}, \mathbf{F})/n$, respectively.

2.2.2 Selected algorithms

2.2.2.1 The Plaid Model

The Plaid Model is a statistically inspired modeling approach first proposed by Lazzeroni and Owen (2002). Assuming that the underlying data structure can be modeled as a

summation of K possibly overlapping layers with indices $k = 1, \cdots, K$ the model can be written as

$$\mathbf{X} \approx \theta_0 \sum_{k=1}^{K} \theta_k \rho_{i,k} \kappa_{j,k} + \epsilon_{ij}. \tag{2.27}$$

where $\rho_{i,k} = 1$ if the ith row belongs to the kth cluster and is zero otherwise, $\kappa_{j,k} = 1$ if the jth column belongs to the kth layer. θ_0 is the background effect of the complete data matrix, θ_k is the layer effect and ϵ_{ij} is an error term.
The Plaid Model is similar to a SVD with $d_k = \theta_k$, $\mathbf{u}_k = \rho_{i,k}$ and $\mathbf{v}_k^T = \kappa_{j,k}$. In contrast to the a regular SVD the layers of the Plaid Model are not restricted to be orthogonal, but $\rho_{i,k}$ and $\kappa_{j,k}$ are restricted to have values in $\{0, 1\}$.
In general the θ_ks can be extended to have the form $\theta_k = \mu_k + \alpha_{i,k} + \beta_{j,k}$, where μ_k is the mean effect of the kth layer and $\alpha_{i,k}$ and $\beta_{j,k}$ are the row and column effects, respectively. In this case each layer corresponds to a two-way ANOVA model. The algorithm first models the background layer and then sequentially fits each individual layer. This process stops if the prespecified number of layers K is reached or if any fitted layer is not significant, which is determined by a permutation test. The original algorithm updates the cluster membership parameters $\rho_{i,k}$ and $\kappa_{j,k}$ using alternating ordinary least squares (OLS). With each iteration the resulting non binary estimates $\hat{\rho}_{i,k}$ and $\hat{\kappa}_{j,k}$ are shifted gradually towards binary solutions. Turner et al. (2005) proposed to improve the original Plaid Model algorithm by using binary least squares, so that throughout the fitting process $\hat{\rho}_{i,k}$ and $\hat{\kappa}_{j,k}$ are restricted to have binary values. The improved Plaid Model algorithm showed a better performance regarding the relevance and the recovery of biclusters and in the computation time. An implementation of the improved Plaid Model of Turner et al. (2005) can be found in the R-package *biclust* (Kaiser et al., 2008). The parameter settings used in the simulation study are displayed in Table 1.

2.2.2.2 The Iterative Signature Algorithm (ISA)

The ISA performs an iterative heuristic search for biclusters for which the expression values of the corresponding genes are most similar over the conditions and vice versa. As input data the algorithm employs two normalized forms of the data matrix. Where the first matrix $\mathbf{X}_G$ has been standardized with respect to the rows and second matrix $\mathbf{X}_C$ has been standardized with respect to the columns. In addition an arbitrary gene threshold g_{thr} and condition threshold c_{thr} and a binary starting vector that represents

the genes $\mathbf{g}$ are needed. The algorithm scores the conditions by the linear transformation $\mathbf{X}_G\mathbf{g}$ and identifies those conditions that have a higher score than the condition threshold. Given the resulting scored condition vector $\mathbf{c}$, genes that have a higher score than the gene threshold are identified. In this case the scores are according to the linear transformation $\mathbf{X}_C^T\mathbf{c}$. These two steps are alternated until between any two iterations the set of identified genes does not change. According to Bergmann et al. (2003) the ISA is a generalization of the SVD and will give similar results when applied without the thresholds. For the simulation study we used the implementation of the ISA algorithm available in the R-package *isa2* (Csardi, 2010). In addition, prior to calculating the validation indices we excluded any non-robust or non-unique biclusters by applying the additional filtering functions available in this R-package.

2.3 Evaluation of example data sets

2.3.1 Lung cancer data set

Here we analyzed the same subset of the lung cancer gene expression data set (Bhattacharyee et al., 2001) that was used by Lee et al. (2010) to illustrate the SSVD algorithm. This data set contain 56 samples and gene expression values of $12\,625$ genes measured using the Affymetrix 95av2 GeneChip. The samples comprise 20 pulmonary carcinoid samples (Carcinoid), 13 colon cancer metastasis samples (Colon), 17 normal lung samples (Normal) and 6 small cell lung carcinoma samples (SmallCell). Lee et al. (2010) applied the SSVD method to this gene expression matrix and decomposed it into the first three sparse SVD-layers. For each of the resulting SVD-layers the *degree of sparsity* was relatively low, e.g. all singular vectors that correspond to the samples contained no non-zero coefficients and the singular vectors that correspond to the genes contained $3\,205$, $2\,511$ and $1\,221$ non-zero coefficients. Scatterplots of the sample singular vectors showed a clear grouping of the samples into the different cancer subtypes. In addition, Lee et al. (2010) formed 27 gene sets according to the sign of the coefficients in the three gene singular vectors. The mean expression profiles of these gene sets showed clear differences between the cancer subtypes. However, despite these results a direct interpretation of each singular vector pair is not possible. To obtain SVD-layers with a higher *degree of sparsity* that can be interpreted as single biclusters, we applied the S4VD algorithm controlling a PCER of 0.5 for falsely select-

ing coefficients in the sample singular vector and a PCER of 0.01 for falsely selecting coefficients in the gene singular vector. Furthermore, our interest lies in clusters that correspond to the distinct subclasses which show no overlap with regard to the samples, e.g. each sample is assigned to only one bicluster. Therefore, after a sparse SVD-layer is fitted, we exclude the corresponding columns from the data matrix and applied the S4VD algorithm to the resulting submatrix.

2.3.2 Ependymoma data set

2.3.3 Protein array data set

2.3.4 Gene Set Enrichment Analysis

To examine whether the genes corresponding to a bicluster reflect genes that belong to the similar functional groups, we conducted a gene set enrichment analysis. To this end, Fisher's exact test was applied to test the association of the bicluster gene sets with Gene Ontology (GO) terms of the biological process Ontology. Moreover, we used the capabilities of the R-package *topGO* (Alexa, 2006) to perform the testing and to correct the resulting p-values for the directed acyclic graph (DAG) structure of the Ontology. Therefore we applied the so called *elim* algorithm that takes the structure of the DAG into account, by removing all genes that are annotated to a significantly enriched child node from all its ancestor nodes.

2.4 Interactive Graphics

2.4.1 SEURAT

3 Results

The aim of this research is to determine whether the here proposed S4VD approach is able to find biological relevant and stable biclusters in gene expression data. To this end, we applied the S4VD to a well known lung cancer gene expression data set (Bhattacharyee et al., 2001) . Furthermore, to examine the influence of increasing levels of noise regarding the performance of the S4VD algorithm, we performed a simulation study. The S4VD algorithm was compared with the SSVD method, the improved Plaid Model (Turner et al., 2005) and the ISA (Bergmann et al., 2003). The ISA and the Plaid Model are known to be closely related to the SVD.

3.1 Simulation study

In the first part of the simulations we generated 100 artificial data matrices comprising $p = 1000$ genes and $n = 100$ samples, where each entry of the data matrix is set to 0. In each dataset we randomly assigned 100 genes and 10 samples to a bicluster that shows constant upregulated gene expression represented by a value of 1 in the data matrix. Normally distributed noise $\mathrm{N}(0, \sigma)$ was added to each entry of the data matrix. We examined different noise levels in the range of $\sigma = (0, 0.1, \cdots, 1)$. In the second part of the simulation study 100 data matrices of the same dimension were generated. This time four biclusters were included where each consists of 100 genes and 10 samples. Constant up- and down-regulation was represented by values of 1,-1,0.5 and -0.5. For both scenarios the performance of the S4VD algorithm was examined in comparison to the original SSVD algorithm, the improved Plaid Model (PM; Turner et al., 2005) and the ISA (Bergmann et al., 2003). Since the SSVD algorithm does not include a stopping criterion, we considered only the first SVD-layer as result in the first scenario and the first four SVD-layers as the biclustering result in the second scenario. The clustering results were validated by application of an external validation index based on the Jaccard coefficient. In addition, the stability of the clustering results was

assessed through the average proportion of falsely selected rows and columns. Details on the validation indices, the remaining biclustering algorithms and their relation to the SVD are provided in the supplementary material.

3.1.1 Scenario 1

The simulation results of the first scenario are shown in Figure 2. For low noise levels up to $\sigma = 0.3$ all biclustering algorithms except the SSVD show an almost perfect performance with relevance and recovery scores mostly equal to one and no falsely selected rows and columns. For noise levels of 0.1 to 0.7 all biclusters proposed by the SSVD algorithm are too large and on average a proportion around 0.015 of the rows and 0.012 of the columns are falsely assigned. This results in relevance and recovery scores around 0.8. In case of larger noise levels the SSVD algorithm often fails to converge and thus the relevance scores and the number falsely assigned rows and columns approach zero. For noise levels above 0.3 the first bicluster detected by the Plaid Model usually consists of a strict subset of those rows and columns that belong to the true artificial bicluster in the data. Thus the performance of the Plaid Model regarding the relevance and the recovery decreases with noise. Furthermore, the algorithm starts to fit the noise and proposes a number of further small biclusters. This explains why the relevance score is inferior compared to the recovery score. Most of these small biclusters correspond to parts of the true artificial bicluster and hence the proportions of falsely assigned rows and columns are close to zero. Beginning with a noise level of 0.5 the ISA proposes an increasing number of biclusters of which only one shows a strong agreement with the true bicluster. Even after applying the additional filtering functions available in the *isa2* R-package (Csardi et al., 2010) some nonsense biclusters remain. Thus both scores start to decrease with noise but are superior to the Plaid Model. The number of falsely assigned rows and columns increases with the noise level indicating that some of the detected biclusters correspond to fitted noise. Regardless of the noise level the S4VD algorithm always detects a single bicluster that agrees with the true bicluster. For noise levels above 0.6 the proposed bicluster becomes smaller and represents only a part of the true bicluster. Therefore both scores start to decrease with noise but are superior to that of all other biclustering methods considered in the simulation study. Due to the stability selection the S4VD algorithm rarely assigns false rows and columns to the proposed bicluster and does not detect any additional nonsense clusters. Thus the average proportions of falsely assigned rows

and columns are almost zero for all noise levels.

3.1.2 Scenario 2

The results of the second part of the simulation study are shown in Figure 3. For noise levels below 0.3 the ISA and the S4VD showed relevance and recovery scores around 1 and the according average proportions of falsely assigned objects are near zero. This indicates that both algorithms are able to correctly detect all of the four artificial biclusters present in the data. The Plaid Model algorithm in some cases perfectly revealed the hidden structure, but in other situations depending on the randomly chosen starting values and the noise level the algorithm falsely assigns rows and columns to the biclusters. The stopping criterion of the algorithm depends on a permutation test that can fail to reject unimportant biclusters that correspond to noise. On the other hand for higher noise levels the permutation test also tends to reject biclusters early in the fitting process so that only three or less biclusters are detected. Thus the resulting relevance and recovery scores are highly variable and decrease with noise. Regarding low noise levels, the SSVD algorithm mostly identifies the correct biclusters but usually falsely assigns some additional rows and columns. This behavior maintains for higher noise levels, but additionally the number of correctly identified biclusters becomes less. The performance of the ISA decreases due to an increasing number of identified irrelevant biclusters, starting with noise levels above 0.2. For noise level 0.5 the medians of both similarity scores are approximately 0.5 and the relevance scores show a high variability. For noise levels above 0.5 the two embedded biclusters generated to have a constant up- and down-regulation of 0.5 and -0.5 are masked by noise, and hence, the ISA as well as the S4VD algorithm tend to miss these clusters. This results in a slight increase of their relevance scores while the recovery scores decrease. Moreover, the relevance scores for both algorithms show a high variability at noise level 0.5. In summary, the S4VD algorithm outperforms all other biclustering algorithms considered in the simulation study regarding the similarity scores and the number of falsely assigned rows and columns for all simulation scenarios.

3.2 Lung cancer data set

Three biclusters have been obtained and are shown in the heatmap in Figure 1. The first bicluster corresponds to a subset of 550 genes and a subset of 28 samples including 14 Normal samples and 14 Carcinoid samples. The second bicluster comprises 12 Colon samples and one falsely assigned Carcinoid sample together with a subset of 506 genes. The third bicluster consists of 6 SmallCell samples and 344 genes. All other samples and genes have not been assigned to any bicluster. To illustrate that the selected genes represent genes that are associated with the cancer subtypes we performed a geneset enrichment analysis. Tables of all significantly enriched Gene Ontology (GO) terms ($p < 0.01$) as well as a description of the analysis can be found in the supplementary material. Bhattacharjee et al. (2001) identified several possible marker genes for the different cancer subtypes. A list of eight of these genes together with the corresponding selection probabilities with respect to the three bicluster are shown in Table 1. *TGF-β receptor II*, *tetranectin*, *retinoic acid receptor responder 3* and *ficolin 3* are known to be highly expressed in normal lung tissue compared to carcinoid tissue and thus have high selection probabilities for the first bicluster. This coincides with the GO analysis, e.g. two of the 62 GO-terms that are significantly enriched by the genes corresponding to the first bicluster are *TGFβ receptor signaling pathway* (GO:0007179) and *response to retinoic acid* (GO:0032526). *Integrin,α6* as well as *v-myc* (*c-myc*) are usually overexpressed in colon cancer. These genes have high selection probabilities with respect to the second bicluster. In addition, among the 61 significantly enriched GO-terms corresponding to the second bicluster is the term *endothelial cell migration* (GO:0043542) which coincides with the fact that the associated samples correspond to colon cancer metastases. The *cell-cycle inhibitor protein p18* and *thymosin-β* are markers for small cell carcinomas and show high selection probabilities in the third bicluster. Among the 97 GO-terms significantly enriched in the third bicluster are many cell cycle associated terms, e.g. *cell division* (GO:0051301), *mitotic spindle organization* (GO:0007052) and *cell cycle checkpoint* (GO:0000075). Furthermore, for the first bicluster as well as for the third bicluster the GO-term *positive regulation of Notch signaling pathway* (GO:0045747) is significantly enriched. Alterations of the Notch signaling cascade are known to be associated with several human cancer types.

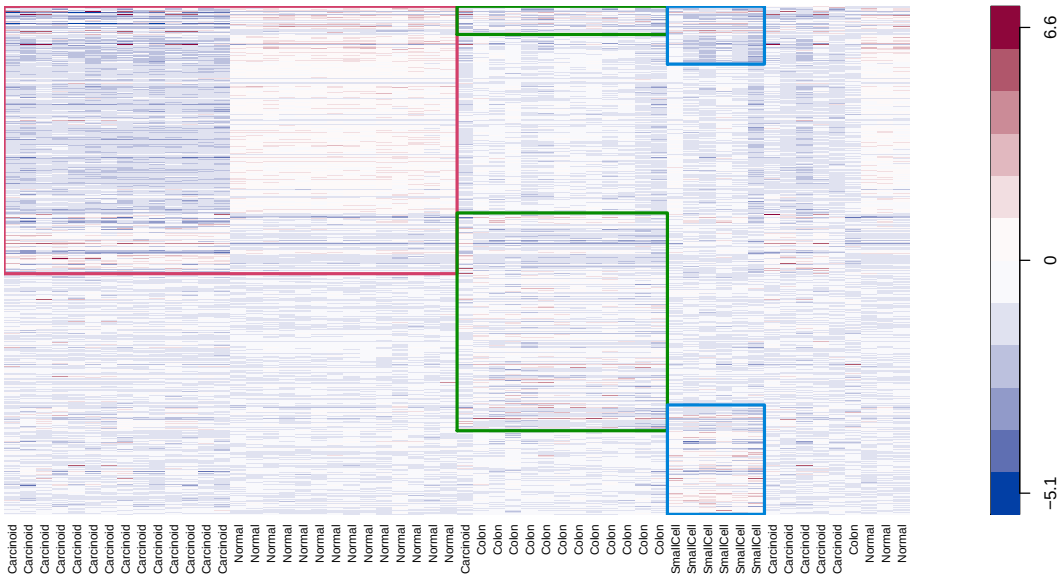

Figure 3.1 Heatmap showing the biclusters identified in the lung cancer data set. Note that the heatmap shows only those genes that have been selected in at least one bicluster. The colored rectangles indicate the genes and samples that correspond to the three biclusters (red corresponds to Bicluster 1, green to Bicluster 2 and blue to Bicluster 3).

3.3 Ependymoma data set

3.4 Protein array data set

Table 3.1 Selection probabilities of lung cancer subclass marker genes

Gene	Bicluste
Retinoic acid receptor responder 3	0
Transforming growth factor, β receptor II (70/80kDa)	1
C-type lectin domain family 3, member B (tetranectin)	1
Ficolin (collagen/fibrinogen domain containing) 3 (Hakata antigen)	1
v-myc myelocytomatosis viral oncogene homolog	0
Integrin, α 6	0
Cyclin-dependent kinase inhibitor 2C (p18)	0
Thymosin β	0

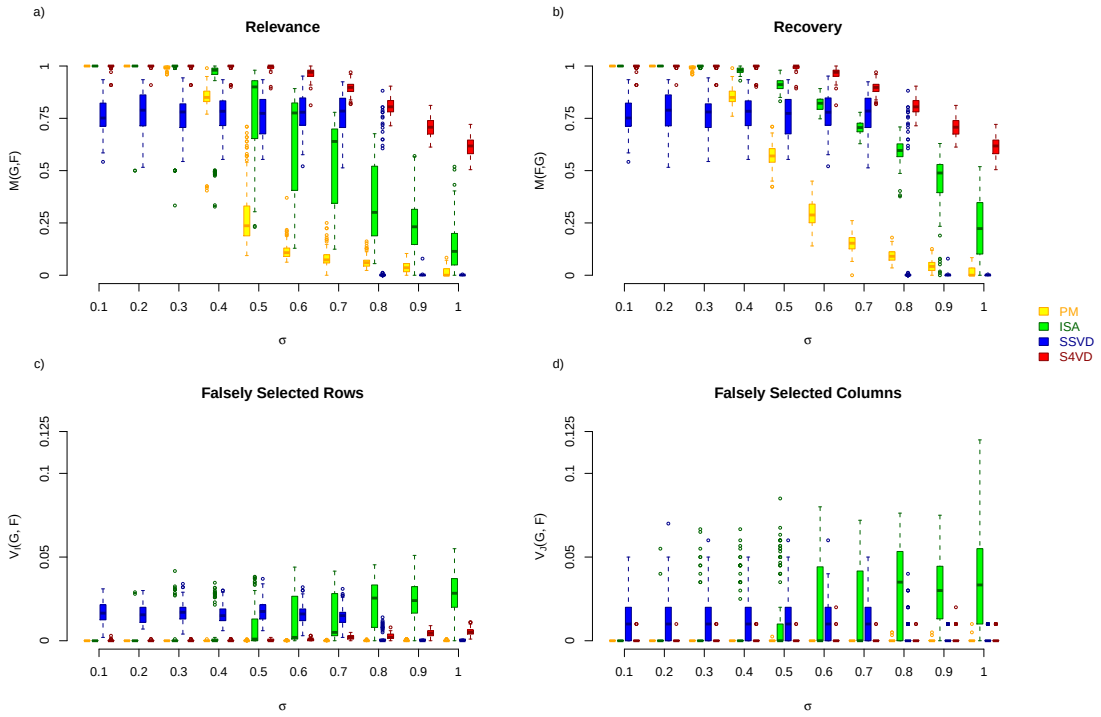

Figure 3.2 Simulation results of the first scenario. The relevance score $M(G, F)$, recovery score $M(F, G)$ and the average proportions of falsely assigned rows $V_I(G, F)$ and columns $V_J(G, F)$ are described in the supplementary material. The boxplots show the distribution of these validation indices with respect to the 100 simulated data sets. σ indicates the considered noise level.

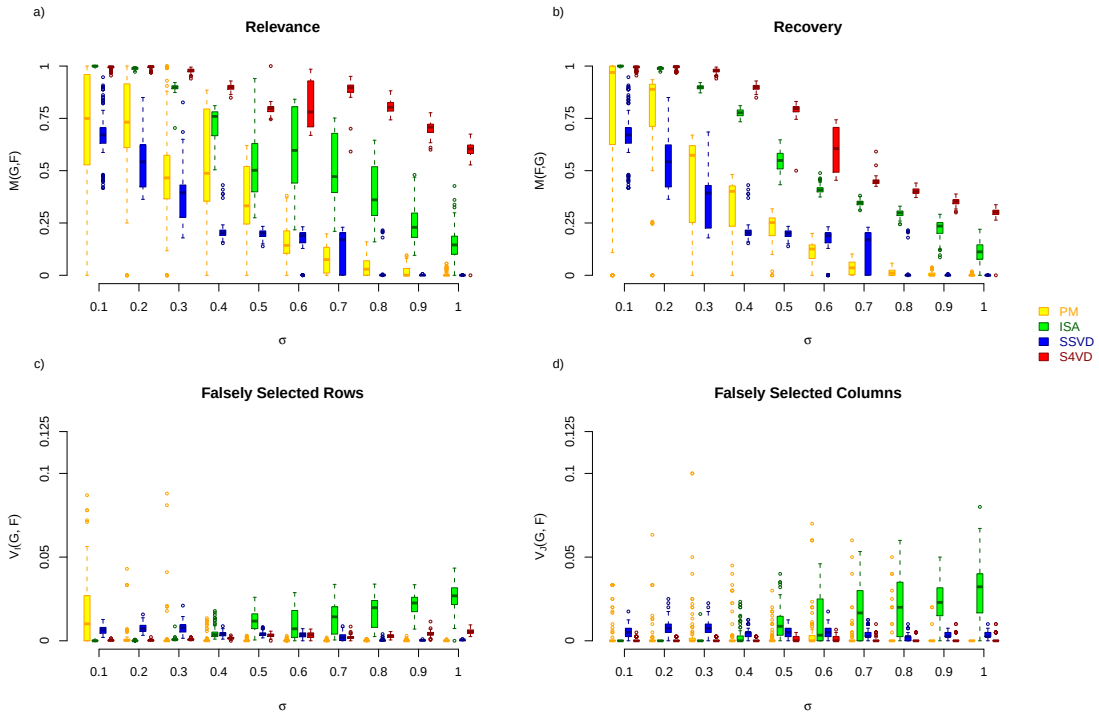

Figure 3.3 Simulation results of the second scenario. The relevance score $M(G, F)$, recovery score $M(F, G)$ and the average proportions of falsely assigned rows $V_I(G, F)$ and columns $V_J(G, F)$ are described in the supplementary material. The boxplots show the distribution of these validation indices with respect to the 100 simulated data sets. σ indicates the considered noise level.

4 Discussion

In this paper we propose a new biclustering algorithm which combines the SSVD algorithm suggested by Lee et al. (2010) with the stability selection of Meinshausen and Bühlmann (2010). In brief, the model selection based parameter tuning of the penalized regression models of the SSVD algorithm is replaced by a subsampling based variable selection that controls type-one error rates. The S4VD approach here presented allows to control the *degree of sparsity* of the resulting SVD-layers by choosing desired type-one error levels. The stability selection estimates the selection probabilities of the rows and columns to belong to a bicluster. Depending on the chosen type-one error levels the results are robust biclusters represented by rows and columns that have high selection probabilities. If the noise level is getting too high the stopping criterion leads to an interruption of the S4VD algorithm preventing from fitting additional SVD-layers that correspond to noise. So far, the S4VD method is the only biclustering approach that takes the cluster stability regarding perturbations of the data into account.

We applied the S4VD algorithm to evaluate a lung cancer microarray data set and showed that the resulting biclusters represent tumor subclasses together with coregulated genes. Genes that were known as markers, showed high selection probabilities in the respective biclusters. A gene set enrichment analysis revealed that the genes associated with identified biclusters belong to significantly enriched cancer related biological processes. In a simulation study the S4VD algorithm was compared with the SSVD algorithm, the improved Plaid Model (Turner et al., 2005) and the ISA (Bergmann, 2003). The S4VD algorithm showed the best performance regarding the recovery of biclusters and was less susceptible to noisy data compared to the other methods.

However, the subsampling steps of the stability selection make the S4VD algorithm computationally demanding. We presented a simple improvement that strongly reduces the computation time.

5 Summary

References

Ben-Dor A, Chor B, Karp R, Yakhini Z (2003) Discovering local structure in gene expression data: The Order-Preserving submatrix problem. Journal of Computational Biology 10: 373–384

Bergmann S, Ihmels J, Barkai N (2003) Iterative signature algorithm for the analysis of large-scale gene expression data. Phys Rev E Stat Nonlin Soft Matter Phys 67: 031902

Bhattacharjee A, Richards WG, Staunton J, Li C, Monti S, Vasa P, Ladd C, Beheshti J, Bueno R, Gillette M, Loda M, Weber G, Mark EJ, Lander ES, Wong W, Johnson BE, Golub TR, Sugarbaker DJ, Meyerson M (2001) Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses. Proceedings of the National Academy of Sciences of the United States of America 98: 13790–13795

Bhattacharya A, K De R (2009) Bi-correlation clustering algorithm for determining a set of co-regulated genes. Bioinformatics 25: 2795–2801

Busygin S, Prokopyev O, Pardalos PM (2008) Biclustering in data mining. Comput Oper Res 35: 2964–2987

Carmona-Saez P, Pascual-Marqui R, Tirado F, Carazo J, Pascual-Montano A (2006) Biclustering of gene expression data by non-smooth non-negative matrix factorization. BMC Bioinformatics 7: 78

Cheng Y, M CG (2000) Biclustering of expression data. Proc Int Conf Intell Syst Mol Biol 8: 93–103

Dudoit S, Shaffer JP, Boldrick JC (2003) Multiple hypothesis testing in microarray experiments. Statistical Science 18: 71–103

Eckart C, Young G (1936) The approximation of one matrix by another of lower rank. Psychometrika 1: 211–218

Getz G, Levine E, Domany E (2000) Coupled two-way clustering analysis of gene microarray data. Proc Natl Acad Sci U S A 97: 12079–12084

Gribov A, Sill M, L?ck S, R?cker F, D?hner K, Bullinger L, Benner A, Unwin A (2010) Seurat: visual analytics for the integrated analysis of microarray data. BMC Med Genomics 3: 21

Hartigan JA (1972) Direct clustering of a data matrix. Journal of the American Statistical Association 67: 123–129

Hochreiter S, Bodenhofer U, Heusel M, Mayr A, Mitterecker A, Kasim A, Khamiakova T, Van Sanden S, Lin D, Talloen W, Bijnens L, Göhlmann H, Shkedy Z, Clevert D (2010) Fabia: factor analysis for bicluster acquisition. Bioinformatics 26: 1520–1527

Kasif S, Murali TM, Murali TM, Kasif S (2003) Extracting conserved gene expression motifs from gene expression data. In: Pac. Symp. Biocomput, 77–88

Kluger Y, Basri R, Chang JT, Gerstein M (2003) Spectral biclustering of microarray data: coclustering genes and conditions. Genome Res 13: 703–716

Lazzeroni L, Owen A (2000) Plaid models for gene expression data. Statistica Sinica 12: 61–86

Lee M, Shen H, Huang JZ, Marron JS (2010) Biclustering via sparse singular value decomposition. Biometrics 66: 1087–1095

Madeira SC, Oliveira AL (2004) Biclustering algorithms for biological data analysis: A survey. IEEE/ACM Trans Comput Biol Bioinformatics 1: 24–45

Mechelen IV, Bock HH, Boeck PD (2004) Two-mode clustering methods: a structured overview. Stat Methods Med Res 13: 363–394

Meinshausen N, Bühlmann P (2010) Stability selection. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 72: 417–473

Monti S, Tamayo P, Mesirov J, Golub T (2003) Consensus clustering: A Resampling-Based method for class discovery and visualization of gene expression microarray data. Machine Learning 52: 91–118

Prelic A, Bleuler S, Zimmermann P, Wille A, B?hlmann P, Gruissem W, Hennig L, Thiele L, Zitzler E (2006) A systematic comparison and evaluation of biclustering methods for gene expression data. Bioinformatics 22: 1122–1129

Santamaría R, Quintales L, Therón R (2007) Methods to bicluster validation and comparison in microarray data. In: Proceedings of the 8th international conference on Intelligent data engineering and automated learning, IDEAL'07, 780–789. Springer-Verlag, Berlin, Heidelberg

Schwarz G (1978) Estimating the dimension of a model. The Annals of Statistics 6: 461–464

Suzuki R, Shimodaira H (2006) Pvclust: an r package for assessing the uncertainty in hierarchical clustering. Bioinformatics 22: 1540–1542

Tanay A, Sharan R, Kupiec M, Shamir R (2004) Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data. Proceedings of the National Academy of Sciences of the United States of America 101: 2981–2986

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

Turner HL, Bailey TC, Krzanowski WJ, Hemingway CA (2005) Biclustering models for structured microarray data. IEEE/ACM Trans Comput Biol Bioinform 2: 316–329

Yang J, Wang H, Wang W, Yu P (2003) Enhanced biclustering on expression data. In: Proceedings of the 3rd IEEE Symposium on BioInformatics and BioEngineering, BIBE '03, 321–. IEEE Computer Society, Washington, DC, USA

Zou H (2006) The adaptive lasso and its oracle properties. Journal of the American Statistical Association 101: 1418–1429

Own Publications Related to this Dissertation

1. **Mustermann S**, Benner A and Kopp-Schneider A (2010) On the Identification of Predictive Biomarkers: Detecting Treatment-by-Gene Interaction in High-Dimensional Data, Computational Statistics and Data Analysis, Second Special Issue on Computational Statistics for Clinical Research (publication in preparation)

2. **Mustermann S** and Benner A (2010) glmperm: A Permutation of Regressor Residuals Test for Inference in Generalized Linear Models. The R Journal, Vol. 2/1, 39-45.

Further Own Publications

4. Hielscher T, Zucknick M, **Mustermann S**, and Benner A (2010) On the prognostic value of survival models with application to gene expression signatures. Statistics in Medicine, 29(7-8), 818-829.

Curriculum Vitae

PERSONAL INFORMATION

Family name:	Mustermann
First name:	Stephanie
Date of birth:	March 4th 1983
Place of birth:	Berlin
Marital status:	married
Father:	Heinz Mustermann
Mother:	Gertrud Mustermann, née Musterfrau

EDUCATION

2008-present:	Doctoral student at the Division of Biostatistics of the German Cancer Research Center (DKFZ) in Heidelberg
Dezember 2007	Diplom in Biology at the Humbolt University in Berlin
2005-2007	Studies at the Humbolt University in Berlin
June 2003	Abitur at the Raphael Gymnasium in Berlin
1996-2003	Studies at Raphael Gymnasium in Berlin

WORK EXPERIENCE

2007-2008	Consultant at Idiots GmbH in Hamburg

Acknowledgement

I am deeply indebted to my supervisor Mr **xxx** whose help, stimulating suggestions and encouragement helped me in all times of research for this thesis.

Especially I am obliged to Ms **xxx** and Mr **xxx** for all their help, support, interest and valuable hints.

My friends Ms **xxx** and Ms **xxx** as well as my brother Mr **xxx** looked closely at the final version of the thesis for English style and grammar, correcting both and offering suggestions for improvement.

I wish to express my deepest gratitute to Mr **xxx** and to my family for being supportive and sympathetic in difficult times.

Research leading to this thesis was supported by Eli Lilly & Co. for the statistical analysis of the S080 Companion Study. I am grateful to Eli Lilly & Co. for providing the data.