Permutation Filtering: A Novel Concept for Significance Analysis of Large-Scale Genomic Data

Stefanie Scheid and Rainer Spang

Max Planck Institute for Molecular Genetics,
Computational Diagnostics Group,
Ihnestrasse 63-73, D-14195, Berlin, Germany
firstname.lastname@molgen.mpg.de

Abstract. Permutation of class labels is a common approach to build null distributions for significance analyis of microarray data. It is assumed to produce random score distributions, which are not affected by biological differences between samples. We argue that this assumption is questionable and show that basic requirements for null distributions are not met.

We propose a novel approach to the significance analysis of microarray data, called permutation filtering. We show that it leads to a more accurate screening, and to more precise estimates of false discovery rates. The method is implemented in the Bioconductor package *twilight* available on http://www.bioconductor.org.

1 Introduction

Screening thousands of candidate genes using some scoring function is a widely applied strategy in the analysis of microarrays. A typical scenario is the search for differentially expressed genes, where the sores can be fold changes or t-statistics. Screening inherently leads to a multiple testing problem, which requires the definition of a null distribution of scores. It is common practice to use simulated distributions obtained from randomizations of the original data [1]. With a set of samples (arrays) and corresponding class labels for the samples, one calculates scores for the original class labels, and compares them to the distribution of scores obtained from random shuffling of the class labels. Permutation approaches are popular because the correlation structure of gene expression levels is unknown, which makes the definition of a theoretical joint null distribution difficult. By randomly assigning the class labels to the samples and recomputing scores one circumvents this difficulty and generates a set of random scores, which serves as a null distribution for statistical inference. One transforms the scores obtained from the original class labels to empirical p-values by using the distribution of simulated scores from the permutation null model. Under the assumption that not a single gene is differentially expressed, one expects that this set of p-values is uniformly distributed. Several methods for estimating global or

A. Apostolico et al. (Eds.): RECOMB 2006, LNBI 3909, pp. 338–347, 2006.

local false discovery rates rely on the assumption that the p-value distribution for a set of non-differentially expressed genes is uniform [2, 3, 4, 5, 6, 7].

To borrow information across genes, empirical p-values are computed using a pooled set of scores from all genes on the array [8]. The combined use of class label permutations and score pooling leads to a conceptual problem. In real applications, one typically has both differentially and non-differentially expressed genes. While permutations produce a justifiable null distribution of scores for the non-differentially expressed genes, one expects that they produce wider score distributions for the differentially expressed genes. Wide score distributions are not only expected for genes that are differentially expressed between the class distinction of interest, but also for genes that are differentially expressed regarding some hidden non-random structure in the data, such as the gender of patients or experimental artefacts. As a consequence, the pooled set of scores is contaminated by signals resulting from differentially expressed genes and does not yield a pure null distribution.

In the next section we recall the notation for permutation approaches to multiple testing in microarray studies. In Section 3 we use a clinical data set to show that random permutations produce distributions, which do not meet basic requirements for a null distribution. As a way out of this dilemma, we describe in Section 4 the details of a novel approach to permutation tests termed *permutation filtering*. In Section 5 we show that permutation filtering produces valid null distributions, increases the accuracy of the screening, and leads to more precise estimates of false discovery rates.

2 Notation

Let matrix $\mathbf{X}$ be an $m \times n$ gene-expression matrix with genes in rows and samples in columns. Entry x_{ij} is the value of the ith gene observed for the jth sample with genes $i = 1, \ldots, m$ and samples $j = 1, \ldots, n$. In addition, we have a vector $\boldsymbol{c}_0 = (c_1, \ldots, c_n)$ with c_j being the class label of the jth sample. For simplicity of presentation we only consider binary class labels here. As a real world example, we shall later discuss a breast-cancer data set, where the class label is either one of two clinically defined risk groups.

Let $\boldsymbol{s}_0$ denote the vector of scores with entries $(s_{i0})_{i=1,\ldots,m}$. Let $\boldsymbol{c}$ be a random permutation of the entries of vector $\boldsymbol{c}_0$. Note that we shuffle only the class labels to preserve the correlation structure between the genes. We recompute the score of each gene based on $\boldsymbol{c}$ and derive a set of scores $\boldsymbol{s}$. Say, we do B permutations $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_B$ in total. This yields B random score vectors $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_B$. We join the original and the random scores into the $m \times (B + 1)$ score matrix $\mathbf{S}$ defined as:

$$\mathbf{S} := (\boldsymbol{s}_0\, \boldsymbol{s}_1 \ldots \boldsymbol{s}_B) = (s_{ij}) \text{ with } i = 1, \ldots, m \text{ and } j = 0, \ldots, B.$$

To compute the empirical p-value for score s_{i0}, we count how often a random score exceeds the observed score of the gene of interest:

$$p_{i0} = \frac{1}{m(B+1)} \sum_{k=1}^{m} \sum_{l=0}^{B} I\{|s_{kl}| \geq |s_{i0}|\} \tag{1}$$

with $I\{x\}$ being an indicator function that returns 1 if x is true and 0 otherwise. For simplicity of notation, we summarize the whole process in function $U_{\mathcal{C}}$, which maps a fixed vector of class labels $\boldsymbol{c}_0$ to the vector $\boldsymbol{p}_0 = (p_{i0})_{i=1,\ldots,m}$ of associate p-values

$$U_{\mathcal{C}}(\boldsymbol{c}_0) = \boldsymbol{p}_0 \tag{2}$$

where $\mathcal{C} = \{\boldsymbol{c}_0, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_B\}$ is the set of permutations on which we assess the significance of scores.

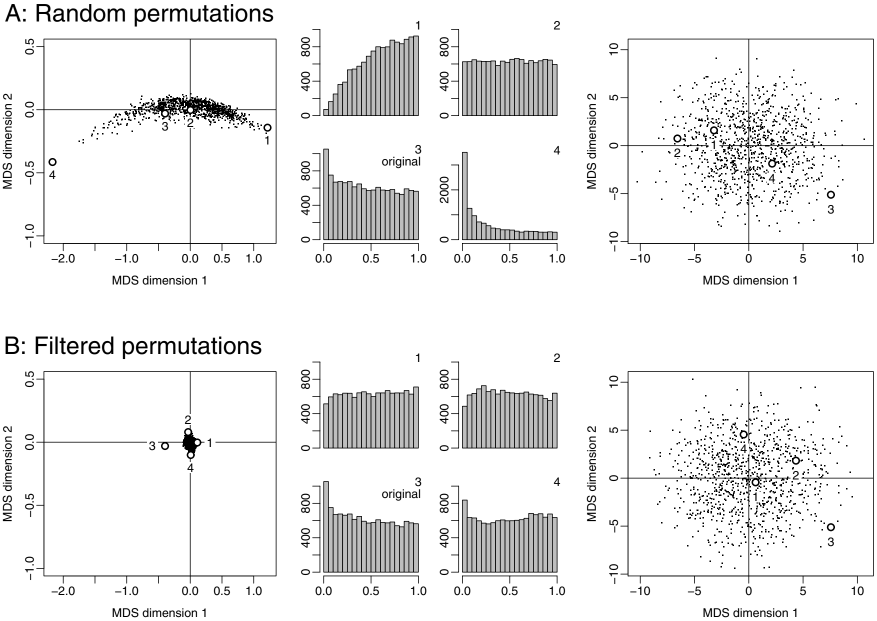
3 Random Permutations Can Produce Invalid Null Distributions

In this section we show that random permutations can produce score distributions, which do not meet basic requirements of a null distribution. We use a clinical microarray study comprising a total of 89 samples from breast-cancer patients measured on Affymetrix GeneChip® HGU95Av2 arrays, which code for $m = 12625$ transcripts/genes [9]. We applied the following preprocessing steps. The background was calculated similar as in the Affymetrix® software Microarray Suite 5.0 [10]. The only difference was that we did not use a correction to avoid negative values. After background correction, we normalized on probe level using a variance-stabilizing procedure [11]. Perfect match probes within a probe set were summarized by the median-polish method [12]. For each probe set, an additive model with probe set, chip and overall effect was fitted using a robust median-polish procedure. Mismatch probes were not taken into account at all.

We compare two risk groups, that is 18 patients with high risk of relapse to 19 low-risk patients ($n = 37$). Again, $\boldsymbol{c}_0$ is a binary vector of length n of class labels where "1" corresponds to the high-risk and "0" to the low-risk class. We score each gene i by computing absolute z-scores as described in [13]. The z-scores are defined as regularized t-statistics with a positive fugde factor added to their denominators. The fudge factor prevents genes with small variances from having high scores. We set the fudge factor to the median value of the pooled standard deviations across genes.

We draw $B = 1000$ random permutations of the original labeling $\boldsymbol{c}_0$, compute the matrix $\mathbf{S}$ of z-scores and empirical p-values $\boldsymbol{p}_0 = U_{\mathcal{C}}(\boldsymbol{c}_0)$. Each permutation is assumed to destroy all biological signals in the data, such that the resulting set of scores consists of random scores, which are not driven by biological signals at all. Deviations in the scores obtained from the original (not permuted) class labels give evidence for differentially expressed genes.

Next we introduce a key requirement for a valid null distribution. We let each permutation of class labels $\boldsymbol{c}_b$ in turn play the role of the original class labels and calculate $\boldsymbol{p}_b := U_{\mathcal{C}}(\boldsymbol{c}_b)$. Hence, we use the function $U_{\mathcal{C}}$ not only for assigning a vector of p-values to the original class labels, but also to each permuted vector of class labels. If the permutation process truly has destroyed all biological signal one would expect to observe uniform distributions of p-values. In panel A of


A: Random permutations

B: Filtered permutations

Fig. 1. A: Random permutation does not always produce valid null distributions. The multi-dimensional scaling plot on the left-hand side shows distributional distances between 1000 sets of p-values resulting from random permutations. Euclidean distances between the CDFs of the p-value sets were used. The four numbered examples show that permutations on the right side in the MDS plot have increasing densities, permutations on the left side have decreasing densities, and only permutations close to the origin produce uniform densities. No. 3 represents the original class labels $\boldsymbol{c}_0$. The scatterplot on the right-hand side shows a second MDS mapping of the permutations, now based directly on the Hamming distances of permuted class labels. The permutations do not cluster but scatter randomly around the origin. **B: Filtering of permutations leaves uniform p-value distributions.** The filtering algorithm returns 1000 permutations that produce uniform p-value distributions, which cluster around the origin in the MDS plot on the left-hand side. Again, no. 3 represents the original labeling $\boldsymbol{c}_0$ while the other three permutations were chosen from the extremes of the filtered set to show that these are still admissible. The MDS plot based on Hamming distances between permutations is similar to the one in A. Filtered permutations still spread evenly in the permutation space. Note that both pairs of MDS plots were derived from joint sets of filtered and unfiltered permutations.

Fig. 1 one can see that this is not the case. The top left plot shows a multi-dimensional scaling (MDS) representation of the p-value distributions obtained by fixing single permutations. We derived the mapping into two dimensions from the Euclidean distances between the empirical cumulative distribution functions (CDFs) of the associated sets of p-values. Close points represent permutations $\boldsymbol{c}_b$, which produce similarly distributed p-values $U_{\mathcal{C}}(\boldsymbol{c}_b)$.

We annotated four exemplary permutations by numbers including the original labels, whose p-value distributions are shown in the top middle plot. Only permutations close to the MDS origin produce uniform p-value distributions. The majority of permutations, however, deviates substantially from uniformity, and often produces distributions, which deviate stronger from uniformity than that of the original class labels.

These results show that random permutations do not produce valid null distributions. Many permutations produce more differential gene expression than the original labels. The scores are not random and the randomization process has not destroyed all biological signal in the data.

4 Permutation Filtering

We now present the permutation filtering procedure. The key idea is to apply the function $U_{\mathcal{C}}$ not only to the original class labels, but also to the permuted ones, as was already done in the previous section. We argue that a valid permutation-based null distribution has to be derived from a set $\mathcal{C}$ of permutations, satisfying the requirement that $U_{\mathcal{C}}(\boldsymbol{c})$ is uniformly distributed for all $\boldsymbol{c} \in \mathcal{C}$.

Assume we have identified a set of permutations $\mathcal{C}_0$, which consists only of permutations that represent valid null hypotheses across all genes. We expect that $U_{\mathcal{C}_0}(\boldsymbol{c})$ is uniform for all $\boldsymbol{c} \in \mathcal{C}_0$. If however, we observe strong deviations from uniformity, either $\boldsymbol{c}_0$ or large parts of the remaining permutations in $\mathcal{C}_0$ correlate with some non-random structure in the data.

We propose the following filtering procedure to derive a set of permutations $\mathcal{C}_0$, which consistently produces uniform p-value distributions when calculating p-values for a fixed permutation using the remaining permutations in $\mathcal{C}_0$:

1. Let $\mathcal{C} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_B\}$ be a set of unique random permutations of the original class labels $\boldsymbol{c}_0$. Apply function $U_{\mathcal{C}}$ to all $\boldsymbol{c}_b \in \mathcal{C}$, which yields the p-value vectors $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_B$. Choose a stepsize k and set $v = 1$.
2. Let F_b be the empirical CDF of the p-values in $\boldsymbol{p}_b$. Test each permutation for uniformity of its p-value CDF by computing the Kolmogoroff-Smirnoff statistic
$$\mathrm{KS}_b = \max_{i=1,\ldots,m} |F_b(p_{ib}) - p_{ib}|.$$
Keep the $v \cdot k$ permutations with the smallest KS statistic in the set $\mathcal{C}_0$. Increase $v = v + 1$.
3. Generate a new set of unique random permutations $\mathcal{C}$, join it with $\mathcal{C}_0$ and apply $U_{\mathcal{C}_0 \cup \mathcal{C}}$ to all $\boldsymbol{c}_b \in \mathcal{C}_0 \cup \mathcal{C}$.
4. Iterate steps 2 and 3 until $|\mathcal{C}_0|$ reaches a predefined number of permutations.
5. Compute the final vector of empirical p-values $\boldsymbol{p}_0 = U_{\mathcal{C}_0 \cup \boldsymbol{c}_0}(\boldsymbol{c}_0)$ for the original class labels.

We chose an iterative design to reduce computational time and save memory. Only a subset of unique random permutations is drawn and tested for uniformity in each step. We keep the admissible permutations in $\mathcal{C}_0$. We do not have to recompute the corresponding scores for the kept permutations. Only when we join $\mathcal{C}_0$ with a new set of permutations, we need to recompute the p-values since we then use an altered set of permutations.

The proposed algorithm is flexible and adaptable to various types of screening studies. We provide an implemention of the procedure in the statistical software language R. We included the algorithm in the Bioconductor package *twilight* for estimating global and local false discovery rates [7, 14, 15, 16, 17].

5 Results

We apply permutation filtering to the breast-cancer data set described in Section 3. Again we use the z-scores for testing. As default parameters in the filtering process we set the stepsize to $k = 50$, the number of permutations per iteration to 1000 and the stopping criterion to $|\mathcal{C}_0| \geq 1000$.

5.1 Permutation Filtering Produces Valid Null Distributions

The effect of permutation filtering is shown in panel B of Fig. 1. Both pairs of plots were derived from joint sets of filtered and unfiltered permutations. Hence the axes of the MDS plots equal those in panel A. As expected, the filtered permutations lie closer to the origin, and even permutations from the margins of the cloud produce acceptable uniform p-value distributions (middle plot).

We removed identical permutations within the iterative filtering. One might suspect that filtering introduces a selection bias in that the filtered permutations cluster strongly and do not spread over the entire permutation space. To show that this is not the case, we display a two-dimensional MDS mapping of the permutations that we derived from the Hamming distances between the binary

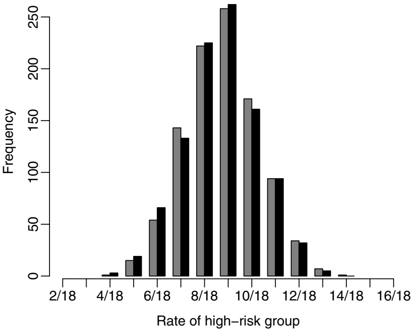


Fig. 2. Permutation filtering does not introduce biased permutations. Distribution of the percentage of the 18 high-risk patients re-assigned to the high-risk group based on random (grey bars) and filtered permutations (black bars). The two distributions do not differ substantially indicating that the filtering does not introduce biased permutations.

vectors of permuted class labels before (panel A) and after (panel B) filtering. Filtered permutations do not form clusters but spread evenly over the permutation space in the MDS representation. There is no visible difference to the corresponding plot for random permutations. We further examine the distribution of the number of samples being randomly re-assigned to their original group. To this end, we count the occurrences of the 18 high-risk patients in the high-risk group for both random and filtered permutations. The result is shown in Fig. 2. We do not observe substantial difference between the two distributions and hence conclude that filtering does not lead to a biased selection of permutations.

5.2 Permutation Filtering Leads to More Significant Genes

A widely used approach to account for multiplicity in microarray studies is to estimate the false discovery rate (FDR) of a list of genes with scores above some prespecified cutoff [18, 19]. The FDR is the expected rate of false positives in this list of genes. Filtering has the effect that one identifies more genes on the same FDR level than without filtering. Hence it increases the sensitivity of the screening for differentially expressed genes.

To show this, we compute p-values of the original labeling $\boldsymbol{c}_0$ based on the random as well as on the filtered set of permutations. For both sets, we estimate false discovery rates as defined in [8]. In Fig. 3, we display FDRs versus the corresponding number of significant genes. As an example, we marked the FDR cutoff of 0.2 with the dashed line. With filtering, this leads to a list of 103 significant genes, which more than doubles the size of a list without filtering (45 genes).

The increase of significant genes is due to the removal of permutations with p-value distributions similar to that of the original labeling, that is with more small p-values than expected. These distributions correspond to score distributions with heavy tails. The removal of these distributions increases the empirical p-values of genes with high scores.

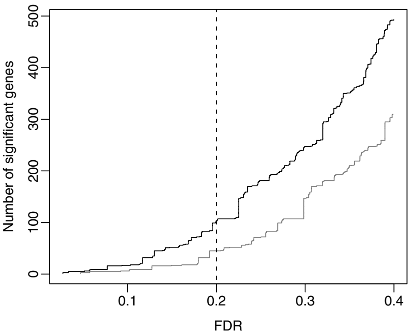


Fig. 3. Permutation filtering leads to more significant genes. FDR cutoffs are plotted versus corresponding numbers of significant genes. The same FDR cutoff leads to more significant findings with filtering (black line) than without (grey line).

5.3 Permutation Filtering Leads to a Higher Accuracy of the Screening

On real data, we can only show an increase of sensitivity since we do not know a priori whether a significant gene is truly induced or not. If the higher sensitivity came for the price of a reduced specificity nothing would be won. To show that this is not the case we use a simulation experiment where the true positives genes are known by design of the simulation.

To this end, we generate random data for 2500 genes and 10 samples per condition. We draw a vector of 2500 random values from a lognormal distribution with location parameter 2 and scale parameter 0.3, and, taking these as mean values, generate 20 random samples from a normal distribution with variance 1. To induce the first 500 genes, we add a value of 2 to the samples of one condition. By adding a value of 4 to five samples of each condition, we introduce hidden non-random structure affecting the following 1000 genes. Note that only the first 500 genes are differentially expressed between populations.

We proceed with the analysis as before and compute p-values based on 1000 filtered and 1000 unfiltered permutations. We rank the genes by p-values and for every rank we estimate the FDR as in [8]. We repeat the data generating procedure 100 times, each time calculating the number of truly induced genes within the list of genes with estimated FDR $\leq 5\%$. Filtering increases the number of correctly identified genes. Without filtering, the list of significant genes includes an average of 457 true positive findings out of 500. Filtering improves the accuracy to 482 correctly identified genes on average. This difference is highly significant in a t-test ($p < 0.0001$).

Hence the filtering increases the sensivity, that is the number of true positives among 500 induced genes, from 0.9134 to 0.9639 on average. The specificity, that is the number of true negatives among 2000 non-induced genes, decreased slightly from 0.9957 to 0.9892. We argue that this loss is negligible regarding the improved sensitivity.

5.4 Permutation Filtering Produces More Precise Estimates of the False Discovery Rate

We use the simulation data from the previous section. The thick black line in Fig. 4 shows the true fraction of induced genes among the top ranking genes. To calculate this line, one has to know a priori which genes are differentially expressed between populations. Hence we can only calculate it in a simulation. The false discovery rate estimates this quantity without knowing the truly differentially expressed genes. Again, one can use both random permutations and filtered permutations to estimate the FDR. The two thin lines in Fig. 4 are the estimated FDR based on filtered (black line) and unfiltered permutations (grey line). While random permutations yield conservative estimates of the false discovery rate, they substantially overestimate it. In contrast, filtered permutation based estimates match the gold standard well. Hence, permutation filtering improves the accuracy of estimated false discovery rates.

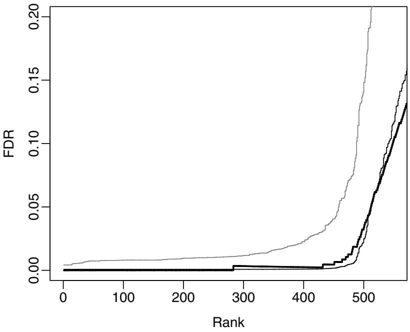


Fig. 4. Permutation filtering leads to more precise FDR estimates. Ranks of high-scoring genes versus the true and estimated FDRs. The FDRs based on filtered permutations (thin black line) estimate the true FDR (thick black line) with high accuracy for the first 500 ranks. FDRs computed without filtering (grey line) lead to conservative but inaccurate estimates.

6 Conclusion

We propose a filtering algorithm that searches for a set of class label permutations where each permutation produces a uniform distribution of p-values. The filtered permutations are then used for calculating empirical p-values and for estimating false discovery rates. The benefits of filtering are valid null distributions, increased numbers of significant genes, a higher accuracy of the screening and more precise estimates of false discovery rates.

We have implemented permutation filtering in the Bioconductor package *twilight* where it is used for calculating both local and global false discovery rates. Permutation filtering is a general concept applicable in many screening studies. It is a novel approach for building valid null distributions. We expect that it will improve the accuracy of high-throughput screenings in various applications in bioinformatics.

Acknowledgments

This work was done within the context of the Berlin Center for Genome Based Bioinformatics (BCB), part of the German National Genome Network (NGFN), and supported by BMBF grants 031U209 and 01GR0455. We thank Anja von Heydebreck and all members of the Computational Diagnostics Group for inspiring discussions and helpful comments.

References

1. Dudoit, S., Yang, Y.H., Callow, M.J., Speed, T.P.: Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. Statistica Sinica **12** (2002) 111–139

2. Broberg, P.: A new estimate of the proportion unchanged genes in a microarray experiment. Genome Biology **5** (2004) P10
3. Dalmasso, C., Broët, P., Moreau, T.: A simple procedure for estimating the false discovery rate. Bioinformatics **21** (2005) 660–668
4. Liao, J., Lin, Y., Selvanayagam, Z.E., Shih, W.J.: A mixture model for estimating the local false discovery rate in DNA microarray analysis. Bioinformatics **20** (2004) 2694–2701
5. Nettleton, D., Hwang, J.G.: Estimating the number of false null hypothesis when conducting many tests. Technical Report 9, Department of Statistics & Statistical Laboratory, Iowa State University (2003)
6. Pounds, S., Morris, S.W.: Estimating the occurrence of false positives and false negatives in microarray studies by approximating and partitioning the empirical distribution of p-values. Bioinformatics **19** (2003) 1236–1242
7. Scheid, S., Spang, R.: A stochastic downhill search algorithm for estimating the local false discovery rate. IEEE Transactions on Computational Biology and Bioinformatics **1** (2004) 98–108
8. Storey, J.D., Tibshirani, R.: Statistical significance for genomewide studies. Proceedings of the National Academy of Sciences **100** (2003) 9440–9445
9. Huang, E., Cheng, S., Dressman, H., Pittman, J., Tsou, M., Horng, C., Bild, A., Iversen, E., Liao, M., Chen, C., West, M., Nevins, J., Huang, A.: Gene expression predictors of breast cancer outcomes. Lancet **361** (2003) 1590–1596
10. Affymetrix: Microarray Suite User Guide, Version 5.0. Affymetrix, Santa Clara, CA, USA. (2001)
11. Huber, W., von Heydebreck, A., Sültmann, H., Poustka, A., Vingron, M.: Variance stabilization applied to microarray data calibration and to the quantification of differential expression. Bioinformatics **18** (2002) 96–104
12. Irizarry, R., Bolstad, B., Collin, F., Cope, L., Hobbs, B., Speed, T.: Summaries of Affymetrix GeneChip probe level data. Nucleic Acids Research **31** (2003) e15
13. Efron, B., Tibshirani, R., Storey, J.D., Tusher, V.: Empirical Bayes analysis of a microarray experiment. Journal of the American Statistical Society **96** (2001) 1151–1160
14. R Development Core Team: R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. (2005) ISBN 3-900051-07-0.
15. Gentleman, R., Carey, V., Bates, D., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J., Hornik, K., Hothorn, T., Huber, W., Iacus, S., Irizarry, R., Leisch, F., Li, C., Maechler, M., Rossini, A., Sawitzki, G., Smith, C., Smyth, G., Tierney, L., Yang, J., Zhang, J.: Bioconductor: Open software development for computational biology and bioinformatics. Genome Biology **5** (2004) R80
16. Scheid, S., Spang, R.: twilight; a Bioconductor package for estimating the local false discovery rate. Bioinformatics **21** (2005) 2921–2922
17. Scheid, S., Spang, R.: Estimation of local false discovery rate - User's guide to the Bioconductor package twilight. CompDiag Technical Report 1, Computational Diagnostics Group, Max Planck Institute for Molecular Genetics, Berlin, Germany (2004)
18. Benjamini, Y., Hochberg, Y.: Controlling the false discovery rate: A practical and powerful approach to multiple testing. Journal of the Royal Statistical Society: Series B **57** (1995) 289–300
19. Storey, J.D.: The positive false discovery rate: A Bayesian interpretation and the q-value. Annals of Statistics **31** (2003) 2013–2035