Preprint version of

Stefanie Scheid and Rainer Spang (2004): **A stochastic downhill search algorithm for estimating the local false discovery rate**, *IEEE Transactions on Computational Biology and Bioinformatics vol. 1, no. 3, p. 98-108*.

A Stochastic Downhill Search Algorithm for Estimating the Local False Discovery Rate

Stefanie Scheid and Rainer Spang

Abstract—Screening for differential gene expression in microarray studies leads to difficult large-scale multiple testing problems. The local false discovery rate is a statistical concept for quantifying uncertainty in multiple testing. In this paper, we introduce a novel estimator for the local false discovery rate that is based on an algorithm which splits all genes into two groups, representing induced and noninduced genes, respectively. Starting from the full set of genes, we successively exclude genes until the gene-wise p-values of the remaining genes look like a typical sample from a uniform distribution. In comparison to other methods, our algorithm performs compatibly in detecting the shape of the local false discovery rate and has a smaller bias with respect to estimating the overall percentage of noninduced genes. Our algorithm is implemented in the Bioconductor compatible R package TWILIGHT version 1.0.1, which is available from http://compdiag.molgen.mpg.de/software or from the Bioconductor project at http://www.bioconductor.org.

Index Terms—Local false discovery rates, stochastic search algorithms, microarray analysis, biology and genetics.

1 Introduction

While multiple testing is a well-established field, statisticians are currently reconsidering its foundations in the light of high dimensional data. Modern high throughput technologies perform exceptionally large numbers of measurements in parallel. Among the most prominent of such technologies are microarrays used in molecular biology and clinical research. Microarrays measure gene expression levels on genome scale. Searching for differentially expressed genes amounts to testing hypotheses about tens of thousands of variables simultaneously.

In general, the problem setting is as follows: Consider a microarray experiment with $r(t)$ genes all showing about t-fold increased expression in a comparison of two predefined groups of samples. By a multiple testing procedure, one can decide whether the level t-fold is significant. In case it is not, one discards all $r(t)$ genes. This is a rushed conclusion. The number $r(t)$ can still be too high to be explained by random fluctuations. The set of genes is in a twilight zone, consisting of both induced and noninduced genes. Being interested in genes in the twilight zone, it is natural to ask how much daylight is left. This is equivalent to asking for the probability that a gene is noninduced conditional on all observations, that is, the entire array of test statistics for differential gene expression. The local false discovery rate introduced by Efron et al. [7] estimates this conditional probability. While methods which aim at controlling error rates detect only the beginning of twilight, local false discovery rates describe the entire course of a sunset.

The paper is organized as follows: In Section 2, we review the concept of the local false discovery rate and put it into context with other approaches to the multiple testing problem. Section 3 reviews existing estimators of the local false discovery rate. In Section 4, we introduce the stochastic downhill search algorithm. We compare the algorithm to previously described estimators in a simulation-based study in Section 5. In Section 6, we apply our estimator to a clinical microarray data set. The discussion contains a summary of our findings and some concluding remarks.

2 Preliminaries

The local false discovery rate is conceptually different from traditional approaches to multiple testing. It is not a standard error measure of a test procedure, but a probability conditioned on all observed data, like it is commonly used in Bayesian analysis. This section reviews common concepts in multiple testing and is thereby redeveloping the idea of the local false discovery rate step by step.

Our discussion is set in the following framework: For samples dividing into two distinct biological conditions, A and B, expression values of M genes are measured as intensity values on a microarray. For each gene $i = 1, \ldots, M$, we compute a score t_i quantifying its differential gene expression between the two conditions. We assume that a high absolute score corresponds to differential expression. For a cutoff value t, let $r(t)$ be the number of genes with a score of t or higher:

$$r(t) = \#\{i | t_i \geq t\}. \quad (1)$$

Statistical analysis needs to be based on a model for expression data. Here, we follow the random effects model of Storey [21]. In general, there will be both induced and noninduced genes and the rate of induction will be different from gene to gene. Gene i can either be noninduced, in this case, the observed score t_i should be close to 0, or, if it is induced by some value $\Delta_i \neq 0$, then the observed score should be close to Δ_i. Of course, we do not know which genes are induced, nor do we know the Δ_i.

• *The authors are with the Max Planck Institute for Molecular Genetics, Computational Diagnostics, Ihnestrasse 63-73, D-14195 Berlin, Germany. E-mail: {stefanie.scheid, rainer.spang}@molgen.mpg.de.*

Manuscript received 26 June 2004; revised 28 Oct. 2004; accepted 16 Nov. 2004.
For information on obtaining reprints of this article, please send e-mail to: tcbb@computer.org, and reference IEEECS Log Number TCBB-0073-0604.

We treat the score for differential gene expression as a random variable T and model its distribution by the following random experiment. In the first step, we randomly decide whether the gene is induced or not by sampling from a Bernoulli variable H with success probability π_0. In a second step, we draw a score T from some distribution $\mathcal{P}_X$ with expectation 0 if $H = 0$ and from a different distribution $\mathcal{P}_Y$ with expectation different from 0 if $H = 1$. The score T can then be decomposed into

$$T = X\, I_{\{H=0\}} + Y\, I_{\{H=1\}}, \tag{2}$$

where $I_{\{.\}}$ is an indicator variable. We apply the model to all genes $i = 1, \ldots, M$, and, hence, are given a set of random variables $T_1, \ldots, T_M$ with $T_i = X_i\, I_{\{H_i=0\}} + Y_i\, I_{\{H_i=1\}}$. The distribution of each T_i depends on the joint distribution of the $(H_i, X_i, Y_i)_{\{i=1,\ldots,M\}}$ under some data generating model P. We assume that:

1. The a priori probability $P[H_i = 0] = \pi_0$ that gene i is not induced is the same for all genes. This allows us to interpret π_0 as the expected proportion of noninduced genes on the microarray.
2. $E[X_i] = 0$ and $E[Y_i] = \Delta_i \neq 0$ for all $i = 1, \ldots, M$.

The joint distribution of variables T_i determines the distribution of the random variables $R(t)$ and $V(t)$ which are defined as

$$R(t) = \sum_{i=1}^{M} I_{\{T_i \geq t\}} \quad \text{and} \quad V(t) = \sum_{i=1}^{M} I_{\{T_i \geq t, H_i = 0\}}. \tag{3}$$

2.1 Relaxing the Error Measure and Gaining Power

For a fixed level t, we can ask:

A.1. Can a score at level t or higher be a chance artifact?
A.2. What fraction of the $r(t)$ genes with score t or higher is expected to be genuinely induced?

Note that the first question emphasizes the level of differential expression t, while the second question emphasizes the number of genes $r(t)$ exceeding this level. Even if the first question needs to be answered with *yes* because the score level t can be reached by random fluctuations with high probability, the number $r(t)$ might be much higher than expected by random fluctuations alone. In this case, it is reasonable to conclude that a certain fraction of genes is genuinely induced.

Question A.1 fits into the setting of classical multiple hypothesis testing. It amounts to calculating the joint type I error rate under the data generating model P when rejecting all genes with $T_i \geq t$, which equals $P[V(t) > 0]$. This error rate is also called the family-wise error rate FWER and amounts to adjusting p-values from single gene tests for multiplicity. Several adjustment procedures have been suggested and evaluated on expression data. For a review, see Dudoit et al. [6]. In the context of several thousand genes, control of the family-wise error rate typically comes at the price of an impractically low power of the test and more relaxed criteria are needed [3]. Shifting the emphasis of statistical analysis from the level of test statistics to the number of genes exceeding the level, as is done in question A.2, is equivalent to switching from p-values to false discovery rates.

Benjamini and Hochberg [3] define the false discovery rate as

$$\mathrm{FDR}(t) = E_P\left[\frac{V(t)}{R(t)}\, I_{\{R(t)>0\}}\right] \tag{4}$$

with $V(t)$ and $R(t)$ as given in (3). In the case that there are no induced genes at all, we have $\mathrm{FWER}(t) = \mathrm{FDR}(t)$. However, if there are both induced and noninduced genes, the FDR is a relaxed test criterion. A score level t with $\mathrm{FDR}(t) \leq \alpha$ can very well be exceeded due to random artifacts. However, among the $R(t)$ excess, we expect only a fraction of α false positives. Using false discovery rates, we allow for some false positive genes but keep control of their proportion.

2.2 Replacing Generating Data Models by Large Sets of Observed Data

For the further development of the false discovery rate, we discuss two variations of question A.2 above. Given a fixed score threshold t, we ask:

B.1. When generating data from the model P and collecting all genes with scores above t, what is the average fraction of false positives among them?
B.2. Given the complete data from a microarray experiment, what is the expected fraction of false positives among all genes with scores equal or higher than t, given that there are $r(t)$ of them?

The first question emphasizes the generating model, while the second emphasizes the large set of observed scores. Question B.1 is based on the idea of using the false discovery rate as a frequentist error measure, like in Benjamini and Yekutieli [4] and Genovese and Wasserman [10]. For a significance level α, Benjamini and Hochberg [3] and Reiner et al. [18] describe threshold rules $t(\alpha)$. When resampling from the generating distribution P many times and collecting all genes with $T_i \geq t(\alpha)$, it is guaranteed that there is, on average, no more than a fraction of α false positives among them, no matter what the underlying model P is. This analysis is only driven by the joint distribution of $R(t)$ and $V(t)$ under the data generating model P and is independent from the actually observed number $r(t)$ in a given microarray experiment. If the observed number $r(t)$ differs from its expectation $E_P[R(t)]$, interpreting the false discovery rate α as the proportion of false positives is misleading. Shifting the emphasis from the generating model in question B.1 to the actually observed data in question B.2 is equivalent to switching from frequentist analysis to conditional approaches including Bayesian analysis.

The first attempt to interpret the false discovery rate in a Bayesian like setting can be found in Storey [21] in a nonconditional approach. The author modifies the definition of the false discovery rate by Benjamini and Hochberg [3] and introduces the positive false discovery rate as

$$\mathrm{pFDR}(t) = E_P\left[\frac{V(t)}{R(t)} \mid R(t) > 0\right]. \tag{5}$$

He also introduces the term q-value. Similar to p-values or adjusted p-values, a q-value can be assigned to each gene and denotes the smallest positive false discovery rate that can be reached if we include this gene into any list of significant genes.

The author shows that, for independent or weakly dependent genes, the pFDR can be rewritten as

$$\mathrm{pFDR}(t) = P[H = 0 \,|\, T \geq t] \tag{6}$$

$$= P[H = 0]\, \frac{P[T \geq t \,|\, H = 0]}{P[T \geq t]}. \tag{7}$$

Although (6) has a mathematically simple form, we find it hard to interpret statistically. Note that $\mathrm{pFDR}(t)$, like the original $\mathrm{FDR}(t)$, only depends on the data generating model P and not on the observed scores $(t_i)_{\{i=1,\ldots,M\}}$. The conditional probability on the right-hand side of (6) is not a real posterior probability, where conditioning on all relevant observations is the goal. The indicator $T \geq t$ can hardly be considered all relevant information available. The term $\mathrm{pFDR}(t)$ depends on the counts $R(t)$ and $V(t)$. Since $R(t) = r(t)$ can be observed, a more intuitive measure is the conditional false discovery rate introduced by Benjamini and Hochberg [3] as

$$\mathrm{cFDR} = E_P\left[\frac{V(t)}{R(t)} \,|\, R(t) = r(t)\right] \tag{8}$$

$$= \frac{E_P[V(t) \,|\, R(t) = r(t)]}{r(t)}. \tag{9}$$

For a comparison between these false discovery rate variants, see Tsai et al. [23].

2.3 From the Significance of Lists of Genes Back to the Significance of Single Genes

We continue with two variations of question B.2. Given the complete data from a microarray experiment and a fixed gene i_0 with a score of t:

C.1. What is the expected fraction of false positives among all genes with scores equal or higher than t?

C.2. What is the probability that gene i_0 is among these false positives?

In the previous section, we have suggested using the cFDR for answering question C.1. Given a list of candidate genes, biologists typically pick a few of them that appear interesting to them. In this case, it is question C.2 and not question C.1 we are interested in. The importance of the difference was first pointed out by Finner and Roters [9]. It becomes apparent in the following scenario: Assume we have 100 genes with scores equal or higher than t, 99 of them with $t_i \gg t$ and one gene i_0 with a score only slightly above t and $\mathrm{cFDR} \approx 0.01$. Since cFDR directly depends on the number $r(t)$ of scores equal or higher than t, it is a property of the entire list of 100 genes including gene i_0. We expect only one false positive among the 100 genes. It is misleading to conclude that each gene in the list of 100 has a probability of 0.01 for being false positive. This probability should not be considered constant among the 100 genes. Certainly, gene i_0 is the most likely candidate for being *the* false positive expected according to the cFDR. The cFDR is a property of a list of genes with little implications on the uncertainties associated to single genes inside this list.

This last obstacle is overcome by the concept of the local false discovery rate introduced by Efron et al. [7]. The local false discovery rate aims at estimating the probability that gene i is false positive given its observed score $t_i = t$, conditional on the vector of all observed scores. The idea is usually formalized by the mixture model

$$f(t) = \pi_0\, f_0(t) + \pi_1\, f_1(t), \tag{10}$$

where $f(t)$ is the density of scores for all genes on the chip. The mixture density is decomposed into f_0, the score density of genes with $H_i = 0$, and f_1, the density under differential expression ($H_i = 1$). The factor π_0 denotes the unknown global proportion of noninduced genes and corresponds to $P[H = 0]$ in (7). Factor π_1 is simply $1 - \pi_0$. With the notation of (10), the local false discovery rate can be defined as

$$\mathrm{fdr}(t) = \pi_0\, \frac{f_0(t)}{f(t)}. \tag{11}$$

The local false discovery rate can also be interpreted as the posterior probability of nondifferential gene expression [7]. Estimating $\mathrm{fdr}(t)$ amounts to estimating all terms on the right-hand side of (11). The density $f(t)$ directly corresponds to the complete vector of scores $(t_i)_{\{i=1,\ldots,M\}}$ and can be estimated for example by smoothing techniques. By doing so, the local false discovery rate is not determined by the data generating model P but depends directly on the observed data vector. In this sense, it is a conditional false discovery rate like the cFDR. However, the density $f_0(t)$ needs to be determined by some data generating model, typically by permutations. Finally, the prior π_0, describing the total proportion of noninduced genes on the chip, can either be determined by an expert, as is done in standard Bayesian analysis, or it can be estimated from the data itself, as is done in empirical Bayesian analysis. Like Efron et al. [7], we will follow the empirical Bayes approach.

3 Estimators of the Local False Discovery Rate

There are several papers on estimating the local false discovery rate. The models in [2], [5], [11], [14], [16], and [17] share the assumption that there exists a transformation W of score T, such that $U = W(T)$ is uniformly distributed in $[0, 1]$ across all genes with $H_i = 0$. In fact, this assumption is not restrictive and the models of Efron et al. [7] and Scheid and Spang [19] can be easily adopted to it. In Section 6, we will review how the transformation can be constructed using permutations of class labels. The transformation W amounts to mapping scores to associated p-values in a single gene test scenario. Since we do not interpret these numbers as p-values, we also omit calling them so. In terms of observed u-values $(u_i)_{\{i=1,\ldots,M\}}$ and a cutoff value u, we write the mixture model as

$$f(u) = \pi_0\, 1 + \pi_1\, f_1(u), \tag{12}$$

where 1 denotes the uniform density on $[0, 1]$. The local false discovery rate can be written as

$$\mathrm{fdr}(u) = \frac{\pi_0}{f(u)}. \tag{13}$$

Equation (12) defines a mixture model with a fixed component, the uniform distribution, and an unknown component $f_1(u)$, the distribution of u-values of induced genes. For simplicity, we will discuss all approaches in the context of a mixture model with a uniform component, also, if the original discussion is on nontransformed scores. Additional assumptions on $f_1(u)$ are needed for identifying the mixture parameter π_0. The literature describes two types of approaches: The first uses fully parametrized models for $f_1(u)$, which ensure by the choice of model that π_0 can be identified. The second type of models are nonparametric with respect to $f_1(u)$ and employ additional assumptions on $f_1(u)$ and π_0 to derive a unique mixture model.

Pounds and Morris [17] establish a fully parametric mixture model by choosing a single parameter beta distribution for the induced genes and estimate the unknown parameters including π_0 by maximum likelihood. Allison et al. [2] describe a more general version of the uniform-beta mixture model by allowing finite mixtures of two-parameter beta distributions for the induced genes. Model selection, with respect to the number of beta components, is done using a bootstrap approach. Liao et al. [14] describe a local version of the uniform-beta mixture model. The authors bin the u-values and fit separate models similar to that of Pounds and Morris [17] for each bin. For model fitting, they use a full Bayesian model with conjugate prior distributions to derive the joint posterior distribution of all model parameters, including π_0.

Nonparametric models for $f_1(u)$ need additional assumptions. Efron et al. [7] assume that

$$\pi_0 \leq \min_t \left\{ \frac{f(t)}{f_0(t)} \right\}. \tag{14}$$

Note that Efron et al. [7] do not use the transformation W and the associated assumption of a uniform component in the mixture model. Using the transformation W, their assumption simplifies to

$$\pi_0 \leq \min_u \{f(u)\}. \tag{15}$$

The authors suggest estimating $f(u)$ by smoothed logistic regression and then using the upper bound $\min_u\{\hat{f}(u)\}$ as an estimator for π_0. Pounds and Cheng [16] also employ (15) in the context of a mixture model with a uniform component for the noninduced genes. However, they use a spacing LOESS histogram estimator for estimating $f(u)$.

Factor π_0 is the Bayesian prior probability $P[H = 0]$. Since it is estimated using the data, the procedures of Efron et al. [7] and Pounds and Cheng [16] are empirical Bayes methods. Assumption (15) is equivalent to assuming that $f_1(u)$ has no uniform component. In case it has, the method of Efron et al. [7] overestimates π_0. Do et al. [5] criticize the biased estimation of π_0 and develop a full nonparametric Bayesian mixture model using Dirichlet processes. Instead of a data driven plug-in estimate of π_0, they impose a uniform prior distribution on it.

In the context of the global false discovery rate, Genovese and Wasserman [11] assume, in addition to the upper bound (15), that $f_1(u)$ is monotonously decreasing, implying:

$$\min_u \{f(u)\} = f_1(1). \tag{16}$$

Equation (16) implies that π_0 can be determined by estimating $f_1(1)$. This is also the strategy suggested by Storey and Tibshirani [22], who describe a smoothed extrapolation-based estimator for $f_1(1)$. The original paper of Tusher et al. [24] contains a simplified version of the extrapolation-based estimator. A review on various other estimators of π_0 can be found in Ferkingstad et al. [8].

In Scheid and Spang [19], we transfer the approach of Tusher et al. [24] from global to local false discovery rates by binning t-values, in this case Wilcoxon ranksum scores, and applying the false discovery rate concept with each bin as a predefined rejection area. The proportion π_0 is calculated globally as in Tusher et al. [24].

4 Estimation of the Local False Discovery Rate via Stochastic Downhill Search

Here, we introduce a novel estimator for the local false discovery rate that is based on an algorithm which splits all genes into two groups, representing induced and noninduced genes, respectively. Starting from the full set of genes, we successively exclude genes until the u-values of the remaining genes look like a typical sample from a uniform distribution. Of course, we cannot conclude that the individual genes in the first set are really *the* induced genes and those in the second set are *the* noninduced genes. However, the size of the two subsets of genes gives rise to an estimator for π_0 and the local false discovery rate can be estimated by the numbers of observed scores in the two sets. The obvious identification problem for π_0 is addressed by searching the largest set of genes such that the distribution of u-values can still pass as a sample from a uniform. We call the algorithm SEP for *successive exclusion procedure*.

The procedure works as follows: We divide all genes into two sets. Let J denote the set of indices representing noninduced genes. Let F_J be the empirical cumulative distribution function of the set of u-values $(u_i)_{\{i \in J\}}$. Our goal is to find the largest set J such that F_J is sufficiently close to a uniform distribution. For a given set J, we measure the goodness-of-fit using the Kolmogoroff-Smirnoff score $S(J) = \max_{i \in J} |F_J(u_i) - u_i|$. In addition, we need a size dependent component that guarantees a high Kolmogoroff-Smirnoff score without removing more values than necessary from the uniform part. This results in a regularized fitting approach using an objective function composed of fit component $S(J)$ and a size component R_λ:

$$g(J, \lambda) = S(J) + R_\lambda(|J|) \tag{17}$$

$$\text{with} \quad R_\lambda(|J|) = \lambda \frac{M - |J|}{M} \log(M - |J|), \tag{18}$$

where $R_\lambda(|J|)$ is strictly monotone in the size of the set J.

For $\lambda = 0$, we have $R_\lambda(|J|) = 0$ and, hence, the objective function only depends on the fit of the empirical distribution of the set $(u_i)_{\{i \in J\}}$ to the uniform distribution. Assume

TABLE 1
Successive Exclusion Procedure in Detail

Stochastic downhill search for fixed λ

Let J be the index set of u-values representing the uniform part in the total set of u-values $(u_i)_{\{i=1,\dots,M\}}$.

1. Set $J_{all} = \{1, \dots, M\}$. Start with the full set $J = J_{all}$.
2. Let F_J be the empirical cumulative distribution function of $(u_i)_{\{i \in J\}}$. Calculate the objective function
$$g(J, \lambda) = \max_{i \in J} |F_J(u_i) - u_i| + \lambda \frac{M - |J|}{M} \log(M - |J|).$$
3. Randomly select an index $i \in \{1, \dots, M\}$. If $i \in J$, set $J' = J \backslash \{i\}$. Else, set $J' = J \cup \{i\}$. Compute $g(J', \lambda)$. If $g(J', \lambda) < g(J, \lambda)$, set $J = J'$. Else, keep J unchanged.
4. Iterate steps 2 and 3 until the objective function was not reduced in $2 \cdot M$ iterations.
5. Output final configuration J.

Estimation of the local false discovery rate

1. Estimate the proportion of non-induced genes as $\widehat{\pi_0} = |J| \cdot M^{-1}$.
2. Divide the interval $[0, 1]$ into 100 equidistant bins. Compute histogram estimators $(h(l))_{\{l=1,\dots,100\}}$ for the density of $(u_i)_{\{i \in J_{all}\}}$, and $(h_0(l))_{\{l=1,\dots,100\}}$ for the density of $(u_i)_{\{i \in J\}}$. For all l, set $q(l) = \widehat{\pi_0}\, h_0(l)/h(l)$.
3. Apply smoothing spline with 7 degrees of freedom and decreasing weights to $q(l)_{\{l=1,\dots,100\}}$. Compute the smoothed spline output in each $(u_i)_{\{i \in J_{all}\}}$. Truncate at 0 and 1 respectively if values exceed the interval $[0, 1]$.

Calibration of λ

1. Set $\lambda_0 = 0$, $\lambda_1 = 0.005$, $\lambda_2 = 0.01, \dots, \lambda_{40} = 0.2$, $\lambda_{41} = 0.21, \dots, \lambda_{70} = 0.5$. For λ_k, $k = 0, \dots, 70$, draw 100 bootstrap vectors $(u_i^B)_{\{i=1,\dots,\min(500,M)\}}$ from the given set of u-values, run SEP with $\lambda = \lambda_k$ on each vector and keep the final configuration J^B. For each vector, output the fit component $S(J^B) = \max_{i \in J^B} |F_J(u_i^B) - u_i^B|$.
2. For $k = 1, \dots, 70$: Compute p-value p_k from a twosample Wilcoxon ranksum test between fit values computed with $\lambda = \lambda_k$ and fit values computed with $\lambda = \lambda_0 = 0$.
3. Apply Bonferroni correction. The adjusted p-values are $(\tilde{p}_k)_{\{k=1,\dots,70\}} = 70 \cdot (p_k)_{\{k=1,\dots,70\}}$. Find first k' such that $\tilde{p}_{k'} \leq 0.05$ and set the regularization parameter to $\lambda = \lambda_{k'-1}$.

For a fixed λ, we need to minimize $g(J, \lambda)$ over all subsets J of genes on the chip which is not feasible by exhaustive search. For heuristic optimization, we use a stochastic hill-descending routine.

that no gene on the chip is induced. In this case, $|J| = M$ and the empirical distribution F_J is a typical sample from a uniform distribution, but the fit is $g(J, 0) \neq 0$ due to the sample variance of F_J. One can still find genes to exclude such that F_J gets closer to identity, hence constructing distributions which are even more "uniform" in terms of goodness-of-fit than a typical sample from a uniform. This overfitting effect leads to a systematic underestimation of π_0. Note that in the overfitting phase we will only marginally improve the fit to the uniform, while $|J|$ and, hence, $\widehat{\pi_0}$ can still change significantly. When choosing $\lambda > 0$, improving the fit by exclusion comes at a price in the size component $R_\lambda(|J|)$. Hence, the estimation of π_0 can be tuned by λ.

Our strategy is to adaptively choose λ such that only significant improvements of the fit component are accepted and, hence, overfitting is avoided. For small λ, the fit values do not differ significantly, which indicates overfitting. With larger λ, we observe significantly worse fit values indicating underfitting. Our goal is to choose λ at the transition of over and underfitting. The calibration of λ is given in detail in Table 1.

Given a candidate set J, we randomly choose a single gene i from the chip. If $i \in J$, let $J' = J \backslash \{i\}$, otherwise, let $J' = J \cup \{i\}$. If $g(J', \lambda) < g(J, \lambda)$, let $J = J'$, otherwise, J remains unchanged. Starting with the full set of genes, this procedure is iterated until the number of unsuccessful trials for a new configuration exceeds twice the total number of genes M. Given the final configuration J, we set

$$\widehat{\pi_0} = \frac{|J|}{M}. \tag{19}$$

We divide the interval $[0, 1]$ into 100 equidistant bins and derive a corresponding histogram estimator $(h(l))_{\{l=1,\dots,100\}}$ for the density of the complete set of u-values and a histogram estimator $(h_0(l))_{\{l=1,\dots,100\}}$ for the subset of genes indexed by J. For all l, we set

$$q(l) = \widehat{\pi_0}\, \frac{h_0(l)}{h(l)} \tag{20}$$

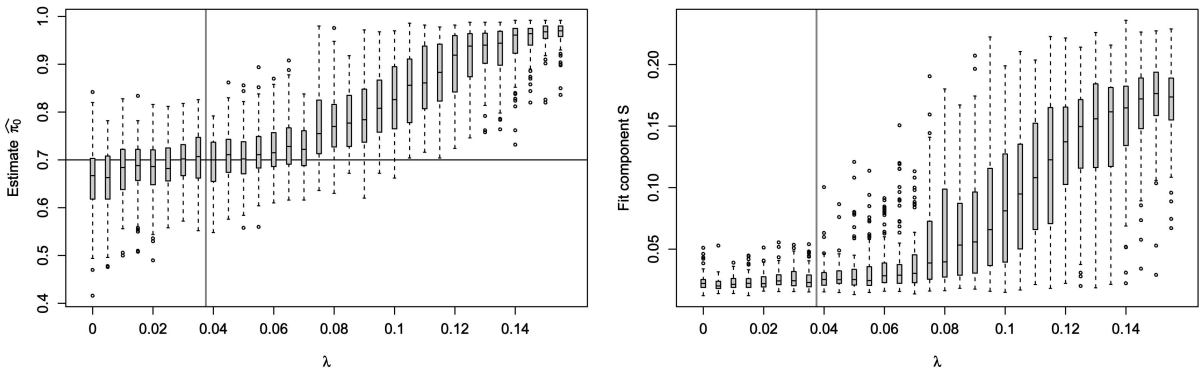

Fig. 1. Effect of penalty term on estimator $\widehat{\pi_0}$ and fit component S. Each boxplot contains values from 100 bootstrap samples of size 500. The vertical line indicates the border between over and underfitting found by SEP. Left: Estimator $\widehat{\pi_0}$. The horizontal line indicates the target π_0. Right: Fit component S.

and estimate the local false discovery rate by interpolating the vector $(q(l))_{\{l=1,\ldots,100\}}$ using a smoothing spline with 7 degrees of freedom and decreasing weights $1/c(l)$, where $c(l)$ denotes the center of bin l. In our experience, this choice of smoothing parameters satisfyingly corrects for increasing variance of the histogram estimates. Table 1 summarizes the overall algorithm.

The stochastic downhill algorithm produces local minima of the objective function. Moreover, it will lead to different local minima when repeated several times. However, the resulting estimates of the local false discovery rate are very stable among reruns. In order to assess the variability of these estimates, we run the algorithm on 1,000 bootstrap samples of the original set of u-values to produce bootstrap averages as point estimators and bootstrap confidence intervals as measures of uncertainty.

5 Simulation Study

For evaluating the SEP algorithm and comparing it to existing approaches from the literature, we use a controlled simulation setting where the true π_0, f_0, and f_1 are known. The simulation is set up such that the resulting u-value distribution is similar to those that we typically observe in applications to microarray data. For simplicity, we simulate u-values rather than gene expression values.

We set $\pi_0 = 0.7$ and compose the distribution of induced genes from two beta distributions reflecting moderate differential expression plus a small amount of very low values from a normal distribution reflecting high differential expression. Overall, we have the mixture density

$$\begin{aligned} f(u) = {} & 0.7\, U[0,1] + 0.15\, B(0.5, 10) \\ & + 0.1\, B(2,5) + 0.05\, |N(0, 0.01)|, \end{aligned} \tag{21}$$

where $U[0,1]$ is a uniform distribution with support $[0,1]$, $B(a,b)$ a beta distribution with shape parameters a and b, and $|N(\mu,\sigma)|$ an absolute normal distribution. The parameters are chosen such that 25 percent of the u-values reflect moderate to high induction modeled by the two beta distributions. The absolute normal distributed values correspond to highly over or under-expressed genes. From (21), we randomly draw 10,000 values. Throughout this section, these are kept fixed to simulate a set of "observed" u-values and all further analysis refers to them.

First, we monitor the performance of our algorithm for various choices of the regularization parameter λ. From the mixture model (21), we target $\pi_0 = 0.7$. The left plot in Fig. 1 shows boxplots of the distributions of estimated π_0-values across bootstrap samples of size 500 for various choices of λ. The small size of 500 was chosen to keep the computation efficient. The horizontal line indicates the target value of 0.7. One can clearly observe that, for small choices of λ, proportion π_0 is underestimated due to overfitting, while large numbers of λ lead to significantly overestimated values of π_0. On the right plot of Fig. 1 are boxplots of the corresponding fit values. For small λ, one can observe an almost constant plane of small fit values indicating that, for all these choices of λ, density $\widehat{f_0}$ is approximately uniform. Note that by comparing the two plots, we can observe that the constant area in the right plot corresponds to the overfitting area in the left plot and that the right end of the constant area matches with the transition between over and underfitting. Our algorithm picked $\lambda = 0.035$, which yields $\widehat{\pi_0} = 0.696$ averaged over 1,000 bootstrap samples of size 10,000. The border between over and underfitting at $\lambda = 0.035$ is indicated by the vertical line in both plots.

We apply SEP on 1,000 bootstrap samples of the set of simulated u-values. The left plot of Fig. 2 shows the performance of the algorithm. On the x-axis are u-values and on the y-axis is $1 - \mathrm{fdr}$, the proportion of induced genes. The gray line and the background histogram are derived directly from the simulation model. The gray line is calculated from the nonuniform components in (21). The background histogram shows the observed proportion of u-values resulting from nonuniform components in the simulation experiment.

The bootstrap average of local false discovery rate estimated by SEP is shown as a black solid line with 95 percent bootstrap confidence intervals shown as dashed lines. The estimated local false discovery rate approximates the target line very well. The mean squared difference between the two lines is about 0.0012 with a standard deviation of 0.002. The highest observed absolute difference is about 0.114, indicating that the maximal error in estimating the proportion of induced genes at a certain

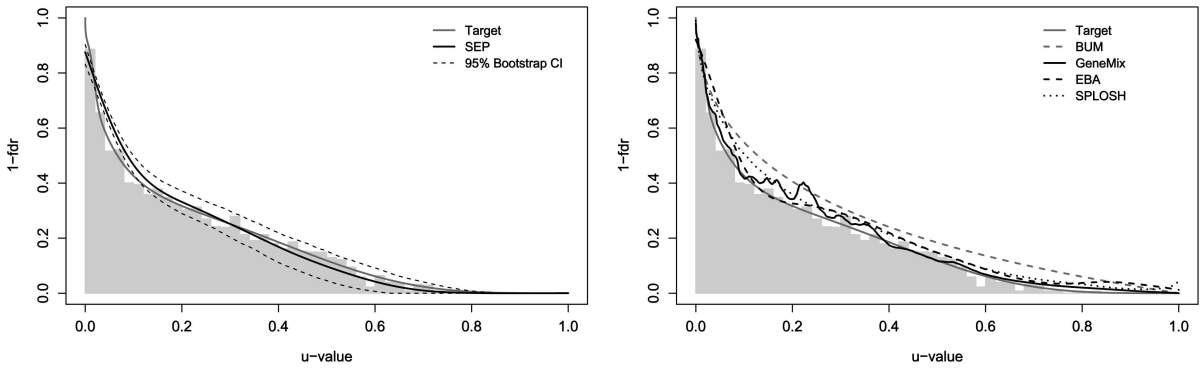

Fig. 2. Comparison of different estimators on simulated u-values. Lines indicate the targeted and estimated local false discovery rate, with the latter averaged over 1,000 bootstrap estimates. The background histogram shows the observed proportion of simulated nonuniform u-values. Left: Successive exclusion procedure. Right: Compared methods.

u-level is not larger than 12 percent. From a practical perspective, this performance is satisfying.

On the simulated set of u-values, we compare SEP to four other methods:

1. The beta-uniform mixture approach BUM of Pounds and Morris [17] using the S-Plus library `post-bum-library.ssc` available from http://www.stjuderesearch.org/statistics/BUM/ and default settings as given in the manual [15].
2. The piecewise beta-uniform mixture approach GeneMix by Liao et al. [14] using the R function `gene_mixture.r` available from http://www.geocities.com/jg_liao/software/ with parameters as given in the simulation part of Liao et al. [14]. We set the number of iterations to 300 and change the source code to output $\widehat{\pi_0}$.
3. The empirical Bayes approach EBA by Efron et al. [7]: The original paper applies logistic regression on observed and permuted test statistics which simulate the null distribution. In order to adapt it to the setting of our simulations, we altered the procedure to work on u-values. Details are given in the Appendix.
4. The spacings LOESS histogram approach SPLOSH by Pounds and Cheng [16] using the S-Plus function `splosh-code.ssc` available from http://www.stjuderesearch.org/statistics/splosh.html with default settings.

For a fair comparison, we use averages from more than 1,000 bootstrap samples for all these methods in addition to cases where this is not suggested in the original paper. Computations used R version 1.8.1 and S-Plus version 4.5. We were not able to include the methods of Allison et al. [2] and Do et al. [5] because no complete software packages are available.

In the right plot of Fig. 2, we compare the performances of the four competing methods. The most important observation is that none of the methods performs badly. BUM performs worse, which is not surprising since the nonnull part of (21) contains at least two beta distributions instead of one. SEP and EBA perform best and are hard to distinguish. Table 2 quantifies the performances further.

For each method, the table shows the estimators' performances in terms of mean squared difference with standard deviation and maximum absolute difference to the targeted local false discovery rate. In terms of mean squared difference, GeneMix performs best, while in terms of maximum absolute difference, SPLOSH has a small advantage. However, the methods' performances do not differ substantially. GeneMix displays a much rougher $1 - \mathrm{fdr}$ curve which is inconsistent with the sample variability shown in the background histogram of Fig. 2. It is not a general shortcoming of the method but, rather, is due to the recommended default values for smoothing parameters.

Fig. 2 also shows that all methods capture the shape of density f_1 well. The main difference of the methods is a vertical shift of estimated $1 - \mathrm{fdr}$ curves indicating differences in estimating π_0. Fig. 3 shows boxplots of bootstrap estimates $\widehat{\pi_0}$. The target value 0.7 is marked by a horizontal line. All methods except SEP severely underestimate π_0. Also, the SEP estimator has smallest variance among all nonparametric models.

In addition, we test the algorithm's performance under the complete null hypothesis by applying SEP on a sample of uniformly distributed values which represent u-values from noninduced genes only. Here, we do not want to exclude any values in order to estimate the percentage π_0 close to the target value 1. This setting evaluates the

TABLE 2
Performance of Different Estimators on Simulated u-Values

Estimator	Mean squared diff. (Stand. dev. of squared diff.)	Max. abs. diff.
BUM	.005166 (.0043)	.122
GeneMix	.000774 (.0016)	.101
EBA	.002204 (.0042)	.139
SPLOSH	.001849 (.0023)	.091
SEP	.001165 (.0020)	.114

Mean and standard deviation of squared differences and maximum absolute difference of bootstrap mean to the local false discovery rate.

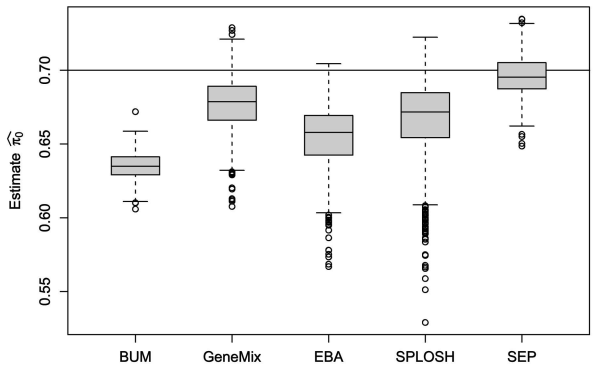

Fig. 3. Comparison of different methods on simulated u-values. Boxplots show 1,000 bootstrap estimators $\widehat{\pi}_0$ for each method. The solid line indicates target π_0.

performance of the algorithm's penalty term or, rather, the procedure that calibrates the penalty parameter λ. We randomly draw 10,000 values from a uniform distribution, apply SEP, and repeat this procedure 100 times. The percentage π_0 is estimated to be 0.99 on average with a standard deviation of 0.0084.

6 Application to Clinical Data

In this section, we give one application of our algorithm to real clinical microarray data. The data set consists of 327 patients with pediatric acute lymphoblastic leukemia [25]. The samples divide into cytogenetically distinct subgroups. We compare two types of chromosomal rearrangement, that is, 15 cases with fusion protein BCR-ABL and 27 cases with fusion protein E2A-PBX1 against 18 normal cases. The latter patients have acute lymphoblastic leukemia but do not show any of the tested chromosomal aberrations.

The study used Affymetrix oligonucleotide array HG_U95Av2 coding for 12,625 transcripts. Each transcript is represented by a *probe set* containing 16-20 *probe pairs*. The probe pair consists of a *perfect match* and a *mismatch* probe. These probes are oligonucleotides of length 25. The perfect match probe is complementary to the target transcript. The mismatch probe is identical except for a change of the 13th nucleotide. Mismatch probes are assumed to reflect nonspecific binding and are used to model the background expression. The following preprocessing steps are divided into background correction, normalization on probe level, and, finally, summarization of probes within a probe set to output one intensity value per transcript and chip.

The background is calculated similarly as in the Affymetrix software Microarray Suite 5.0 [1]. The only difference is that we do not use Affymetrix's correction to avoid negative values. After background correction, we normalize on probe level using the R package VSN with the variance-stabilizing procedure of Huber et al. [12]. The VSN package is part of the R/Bioconductor project and is available from http://www.bioconductor.org. Perfect match probes within a probe set are summarized by the median polish method introduced by Irizarry et al. [13]. For each probe set, an additive model with probe set, chip, and overall effect is fitted using a robust median polish procedure. Mismatch probes are not taken into account at all. Finally, gene expression intensities correspond to the estimated chip effects plus the overall effects.

The variance-stabilizing method gives us expression values on an additive scale, that is, on inverse hyperbolic sine (arsinh) scale. This scale is additive in the sense that, for highly expressed genes, differences in normalized data correspond to fold changes in original data. In fact, for highly expressed genes, the inverse hyperbolic sine approximates the logarithm plus a constant. For lower expressed genes $\log$ and arsinh differ substantially. We use the score

$$T_i = \bar{\alpha}_i - \bar{\beta}_i, \tag{22}$$

where $\bar{\alpha}_i$ and $\bar{\beta}_i$ are the mean expression values on the arsinh scale.

When scoring differential gene expression, logarithmic data can be highly variable for lowly expressed genes. This leads to the problem that fold change estimates are unstable and sensitive to background estimation. More concretely, if one measures a nonexpressed gene twice, one ends up with two small numbers where the first can easily be ten times higher than the second. Note that data on arsinh scale does not show the instabilities in low expression regions because the procedure reduces the mean-variance dependence.

It is often suggested to take the gene-wise variability into account by using t-test scores for logarithmic data. This requires the estimation of the variance of each individual gene. It has been observed by several authors that lists of differentially expressed genes can easily be corrupted by frequently underestimated variances and several regularization procedures have been suggested [7], [20], [24]. We suggest not to use gene-wise variances at all. While this leads to a loss of information, it gives us more intuitive fold change type scores.

Our method is based on u-values, not on scores. To obtain u-values, we follow Tusher et al. [24] and Efron et al. [7] and use permutations of class labels. We use 10,000 balanced permutations to calculate u-values. For each gene, we compute fold change equivalent scores based on the original class label vector, as well as for each permuted vector. The u-value of a gene is then given as the percentage of absolute permutation scores exceeding the absolute original score. These u-values were then submitted to SEP with 1,000 bootstrap runs. Again, local false discovery rates are estimated as bootstrap averages. In addition, we calculate 95 percent bootstrap confidence intervals.

Fig. 4 and Fig. 5 show the estimated local false discovery rates with 95 percent bootstrap confidence intervals both on linear and on logarithmic scale. We have chosen the linear scale to show the overall shape of the local false discovery rate curve and the logarithmic scale to visualize the local false discovery rate, especially in regions of highly induced genes. The ticks at the bottom are 1 percent quantiles of u-values to show how these values are distributed along the x-axes. Fig. 6 contains boxplots of the corresponding bootstrap estimates $\widehat{\pi}_0$.

Both settings suggest large amounts of differential gene expression with bootstrap medians of 0.85 (BCR-ABL versus normal) and 0.78 (E2A-PBX1 versus normal) for proportion π_0, compare to Fig. 6. Following the estimates in

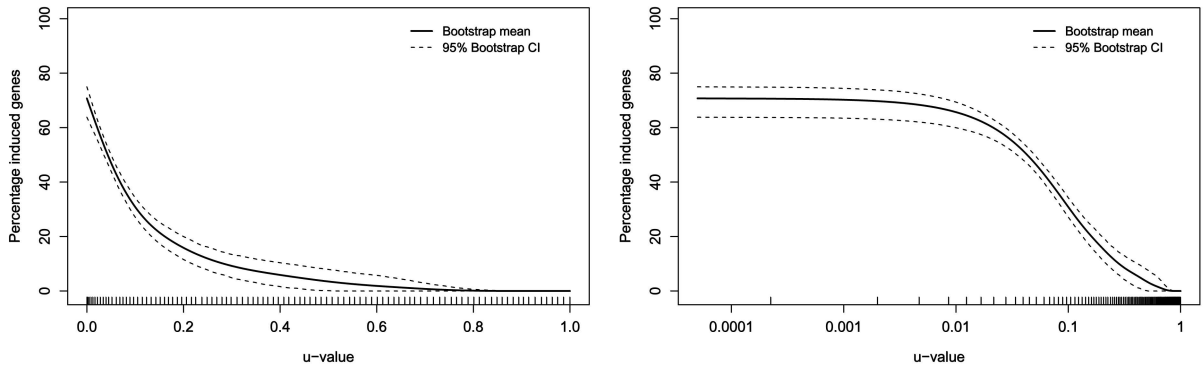

Fig. 4. Leukemia data: BCR-ABL versus normal. SEP bootstrap estimates and 95 percent confidence intervals of local FDR. Bottom ticks mark 1 percent u-value quantiles. Left: Linear scale. Right: Logarithmic scale.

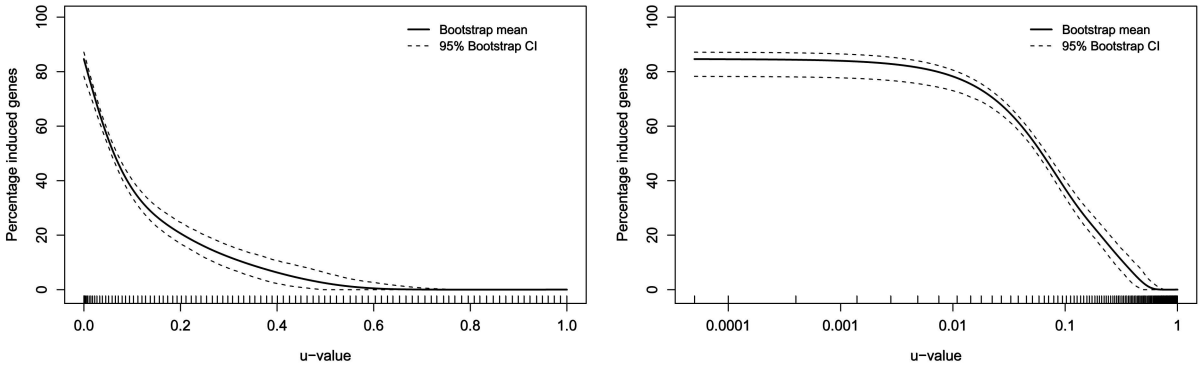

Fig. 5. Leukemia data: E2A-PBX1 versus normal. SEP bootstrap estimates and 95 percent confidence intervals of local FDR. Bottom ticks mark 1 percent u-value quantiles. Left: Linear scale. Right: Logarithmic scale.

Fig. 4 and Fig. 5, the percentage of induced genes decreases quickly in both cases. No differential gene expression is expected for genes with u-values greater than 0.6. The penalty parameters were found to be $\lambda = 0.025$ for BCR-ABL and $\lambda = 0.045$ for E2A-PBX1.

In Table 3, we compare local false discovery rates to the two most widely used concepts in multiple testing: the family-wise error rate, described by adjusted p-values, and the positive false discovery rate, described by q-values. Recall from Section 2 that the adjusted p-value is the family-wise error rate that is reached when rejecting the corresponding gene with a certain multiple testing procedure. The q-value is defined as the smallest positive false discovery rate that is reached when including the corresponding gene into the list of significant genes. We calculate q-values as described in Remark B in Storey and Tibshirani [22] and Bonferroni adjusted p-values $\tilde{p}_i = \min(M \cdot u_i, 1)$ with $M = 12,625$ being the number of genes under examination. Note that u-values correspond to nonadjusted p-values. The first column of Table 3 contains selected percentages of induced genes which corresponds to the posterior probability of differential gene expression, that is, $1 - \text{local false discovery rate}$.

7 Discussion

We have argued that the local false discovery rate, pioneered by Efron et al. [7], is an appropriate measure of statistical significance for large-scale multiple testing encountered in microarray analysis. It is characterized by the following three features:

1. It relates to the number $r(t)$ of genes at a certain level t and not to t itself.
2. It is conditional in the sense that it is constructed from observed data and not from a data generating model.
3. It is local in the sense that it describes the significance of single genes and not of entire lists of genes.

We have proposed a novel algorithm to estimate the local false discovery rate based on a stochastic downhill

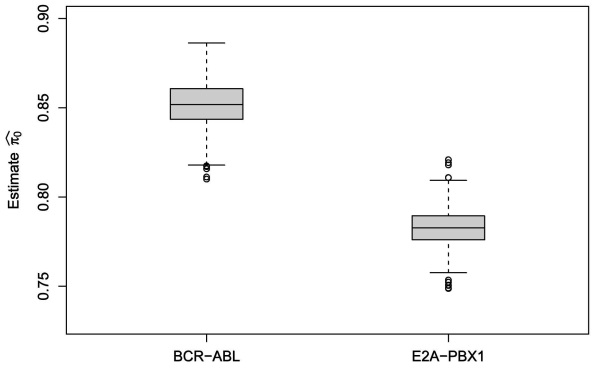

Fig. 6. Leukemia data: Boxplots with bootstrap estimates $\widehat{\pi_0}$ for each experimental setting.

TABLE 3
Leukemia Data: Comparison of u, q, and Adjusted p-Values at Selected Levels of Local False Discovery Rate

Percentage of	**BCR-ABL vs. normal**		
induced genes	u-value	q-value	$\tilde{p}$-value
71	0.0000	0.000	0
50	0.0421	0.337	1
25	0.1252	0.497	1
Percentage of	**E2A-PBX1 vs. normal**		
induced genes	u-value	q-value	$\tilde{p}$-value
85	0.0000	0.0000	0
80	0.0065	0.0665	1
75	0.0144	0.1134	1
50	0.0611	0.2627	1
25	0.1601	0.4142	1

procedure. In comparison to other methods, the algorithm performs compatibly in detecting the shape of the local false discovery rate curve and has a smaller bias with respect to estimating the overall percentage of noninduced genes π_0.

In application to clinical data, we have shown that the local false discovery rate is sensitive to changes in gene expression impossible to detect by classical multiple testing. Note that, in Fig. 4 and Fig. 5, the method is sensitive to surprisingly large u-values. It might appear confusing that the 75 percent level is reached for u-values greater than 0.01 in the E2A-PBX1 comparison. These genes are not significantly induced in the context of adjusted tests and still we estimate their odds for being induced as 3:1, see Table 3. These discrepancies reflect the conceptional differences between classical p-value-based multiple testing and local false discovery rates. Both are measures of uncertainty, but the concepts of uncertainty are different, as we have explained in Section 2. They are complementary and not competing concepts. Consequently, we do not give recommendations to prefer one over the other. However, if one chooses to estimate the local false discovery rate, we believe that our approach is competitive.

There is an important caveat with all estimators of the local false discovery rate discussed in the paper. They all depend on the assumption that u-values from noninduced genes are uniformly distributed. We do not know of any method that guarantees that this assumption holds strictly. In the clinical application, we have constructed u-value distributions using balanced permutations of class labels. While this is a widely used approach, it is not trouble-free. Two points need to be raised: First, strong correlations between genes lead to nonuniformly distributed u-values for noninduced genes. Clustering patterns can emerge. Without knowing the underlying correlation structure of genes, we cannot compensate for this effect. Second, permuted class labels do not ensure that there is no differential gene expression at all on the chip. The data might contain a hidden biological structure, like gender, age, and genetical background of patients. For example, if the patient sample comprises both men and women and a random class label happens to assign most women to the first class and most men to the second, the score will reflect differential gene expression between men and women. If we know which patients are men, we can compensate for this effect. If we are not provided with this information, it is a hidden structure in the data that we cannot compensate for. In this case, the distribution of u-values can be skewed. Nonuniformity of u-value distributions appears to be an unsolved problem in estimating local false discovery rates. We assume that it is also critical for other methods, like the widely used SAM program of Tusher et al. [24]. We believe that statistical science could profit from further research in this field.

APPENDIX

EMPIRICAL BAYES ANALYSIS ON u-Values

The empirical Bayes analysis of Efron et al. [7] is designed for observed test scores t. The original method applies regression to the term

$$\pi(t) = \frac{f(t)}{f(t) + B\, f_0(t)}, \tag{23}$$

where $f(t)$ and $f_0(t)$ are the common densities as given in (11). The authors estimate f_0 from the data matrix with permuted class labels. Factor B is the number of permutations. We adapt the procedure to u-values by setting $B = 1$ and allowing $\widehat{f_0}$ to be the optimal estimate, that is, the uniform density with $\widehat{f_0}(u) = 1$ for all u. As outlined in Section 3, the authors estimate the proportion of noninduced genes π_0 as

$$\widehat{\pi_0} = \min_u \left\{ \hat{f}(u) \right\} = \min_u \left\{ \frac{\hat{\pi}(u)}{1 - \hat{\pi}(u)} \right\}. \tag{24}$$

The estimated local false discovery rate is then given as

$$\widehat{\mathrm{fdr}}(u) = \widehat{\pi_0}\, \frac{1 - \hat{\pi}(u)}{\hat{\pi}(u)}. \tag{25}$$

Along these lines, we apply the same smoothing spline regression as in SEP, compare to Table 1. This is necessary to make EBA directly comparable to SEP such that differences between the two are not due to different smoothing techniques. The original method applies an unweighted natural spline with 5 degrees of freedom [7]. The changed EBA works as follows: Divide the interval $[0, 1]$ into 100 equidistant bins and compute histogram estimators $(h(l))_{\{l=1,\ldots,100\}}$ for the density f of $(u_i)_{\{i=1,\ldots,M\}}$. For all $l = 1, \ldots, 100$, we set

$$\hat{\pi}(l) = \frac{h(l)}{h(l) + 1} \tag{26}$$

and apply the same smoothing spline with 7 degrees of freedom and decreasing weights to $\hat{\pi}(l)_{\{l=1,\ldots,100\}}$, as explained in detail in Table 1. The interpolation of the smoothed spline output in each $(u_i)_{\{i=1,\ldots,M\}}$ gives the estimated ratios $(\hat{\pi}(u_i))_{\{i=1,\ldots,M\}}$, which can then be inserted into (24) and (25).

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 031U109C and 03U117. The authors thank Anja von Heydebreck and all members of the Computational Diagnostics group, especially Dennis Kostka and Florian Markowetz, for their helpful comments and discussions.

REFERENCES

[1] Affymetrix Inc., "Microarray Suite User's Guide," version 5.0, http://www.affymetrix.com/support/technical/manuals.affx, 2001.

[2] D.B. Allison, G.L. Gadbury, M. Heo, J.R. Fernández, C.-K. Lee, T.A. Prolla, and R. Weindruch, "A Mixture Model Approach for the Analysis of Microarray Gene Expression Data," *Computational Statistics and Data Analysis,* vol. 39, pp. 1-20, 2002.

[3] Y. Benjamini and Y. Hochberg, "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing," *J. Royal Statistical Soc. B,* vol. 57, no. 1, pp. 289-300, 1995.

[4] Y. Benjamini and D. Yekutieli, "The Control of the False Discovery Rate in Multiple Testing under Dependency," *Annals of Statistics,* vol. 29, no. 4, pp. 1165-1188, 2001.

[5] K.-A. Do, P. Müller, and F. Tang, "A Bayesian Mixture Model for Differential Gene Expression," Dept. of Biostatistics, Univ. of Texas, http://odin.mdacc.tmc.edu/kim/bayesmix/, 2003.

[6] S. Dudoit, J.P. Shaffer, and J.C. Boldrick, "Multiple Hypothesis Testing in Microarray Experiments," *Division of Biostatistics Working Paper Series,* Univ. of California at Berkeley, no. 110, 2002.

[7] B. Efron, R. Tibshirani, J.D. Storey, and V. Tusher, "Empirical Bayes Analysis of a Microarray Experiment," *J. Am. Statistical Assoc.,* vol. 96, no. 456, pp. 1151-1160, 2001.

[8] E. Ferkingstad, M. Langaas, and B. Lindqvist, "Estimating the Proportion of True Null Hypotheses, with Application to DNA Microarray Data," *Preprint Series in Statistics,* no. 4, Dept. of Mathematical Sciences, Norwegian Univ. of Science and Technology, http://www.math.ntnu.no/preprint/statistics/2003/, 2003.

[9] H. Finner and M. Roters, "On the False Discovery Rate and Expected Type I Errors," *Biometrical J.,* vol. 43, no. 8, pp. 985-1005, 2001.

[10] C. Genovese and L. Wasserman, "Bayesian and Frequentist Multiple Testing," *Bayesian Statistics 7—Proc. Seventh Valencia Int'l Meeting,* J.M. Bernardo, A.P. Dawid, J.O. Berger, M. West, D. Heckerman, M.J. Bayarri, and A.F.M. Smith, eds. Oxford Univ. Press, 2003.

[11] C. Genovese and L. Wasserman, "A Stochastic Process Approach to False Discovery Control," *Annals of Statistics,* vol. 32, no. 3, pp. 1035-1061, 2004.

[12] W. Huber, A. von Heydebreck, H. Sültmann, A. Poustka, and M. Vingron, "Variance Stabilization Applied to Microarray Data Calibration and to the Quantification of Differential Expression," *Bioinformatics,* vol. 18, suppl. 1, pp. S96-S104, 2002.

[13] R.A. Irizarry, B.M. Bolstad, F. Collin, L.M. Cope, B. Hobbs, and T.P. Speed, "Summaries of Affymetrix GeneChip Probe Level Data," *Nucleic Acids Research,* vol. 31, no. 4, e15, 2003.

[14] J.G. Liao, Y. Lin, Z.E. Selvanayagam, and W.J. Shih, "A Mixture Model for Estimating the Local False Discovery Rate in DNA Microarray Analysis," *Bioinformatics,* vol. 20, no. 16, pp. 2694-2701, 2004.

[15] S. Pounds, "User's Guide to BUM Library Version 1-1," St. Jude Children's Research Hospital Memphis, http://www.stjuderesearch.org/statistics/BUM/, 2003.

[16] S. Pounds and C. Cheng, "Improving False Discovery Rate Estimation," *Bioinformatics,* vol. 20, no. 11, pp. 1737-1745, 2004.

[17] S. Pounds and S.W. Morris, "Estimating the Occurrence of False Positives and False Negatives in Microarray Studies by Approximating and Partitioning the Empirical Distribution of p-Values," *Bioinformatics,* vol. 19, no. 10, pp. 1236-1242, 2003.

[18] A. Reiner, D. Yekutieli, and Y. Benjamini, "Identifying Differentially Expressed Genes Using False Discovery Rate Controlling Procedures," *Bioinformatics,* vol. 19, no. 3, pp. 368-375, 2003.

[19] S. Scheid and R. Spang, "A False Discovery Rate Approach to Separate the Score Distributions of Induced and Noninduced Genes," *Proc. Third Int'l Workshop Distributed Statistical Computing,* http://www.ci.tuwien.ac.at/Conferences/DSC-2003/Proceedings/, 2003.

[20] G.K. Smyth, Y.-H. Yang, and T.P. Speed, "Statistical Issues in Microarray Data Analysis," *Functional Genomics: Methods and Protocols,* Methods in Molecular Biology, M.J. Brownstein and A.B. Khodursky, eds., vol. 224, pp. 111-136, 2003.

[21] J.D. Storey, "The Positive False Discovery Rate: A Bayesian Interpretation and the q-Value," *Annals of Statistics,* vol. 31, no. 6, pp. 2013-2035, 2003.

[22] J.D. Storey and R. Tibshirani, "Statistical Significance for Genome-wide Studies," *Proc. Nat'l Academy of Sciences,* vol. 100, no. 16, pp. 9440-9445, 2003.

[23] C.-A. Tsai, H.-M. Hsueh, and J.J. Chen, "Estimation of False Discovery Rates in Multiple Testing: Application to Gene Microarray Data," *Biometrics,* vol. 59, no. 4, pp. 1071-1081, 2003.

[24] V. Tusher, R. Tibshirani, and C. Chu, "Significance Analysis of Microarrays Applied to Ionizing Radiation Response," *Proc. Nat'l Academy of Sciences,* vol. 98, no. 9, pp. 5116-5121, 2001.

[25] E.-J. Yeoh, M.E. Ross, S.A. Shurtleff, W.K. Williams, D. Patel, R. Mahfouz, F.G. Behm, S.C. Raimondi, M.V. Relling, A. Patel, C. Cheng, D. Campana, D. Wilkins, X. Zhou, J. Li, H. Liu, C.-H. Pui, W.E. Evans, C. Naeve, L. Wong, and J.R. Downing, "Classification, Subtype Discovery, and Prediction of Outcome in Pediatric Acute Lymphoblastic Leukemia by Gene Expression Profiling," *Cancer Cell,* vol. 1, pp. 133-143, 2002.

Stefanie Scheid received the MS degree in statistics from the University of Dortmund, Germany. In 2001, she became a PhD student at the Max Planck Institute for Molecular Genetics in Berlin. Her research is focused on statistical applications for gene expression data.

Rainer Spang studied mathematics, computer science, and biology at the University of Bonn and the German Cancer Research Center in Heidelberg. He received the PhD degree in 1999. After a postdoctoral position at Duke University, he became a group leader at the Max Planck Institute for Molecular Genetics in Berlin. His research is focused on the computational analysis of molecular disease mechanisms.