Published in final edited form as:

Biom J. 2010 April ; 52(2): 222–232. doi:10.1002/bimj.200900177.

A Robust Method for Large Scale Multiple Hypotheses Testing

Seungbong Han¹, Adin-Cristian Andrei^{*,2}, and Kam-Wah Tsui¹

¹Department of Statistics, University of Wisconsin-Madison, Medical Science Center 1300 University Avenue, Madison, Wisconsin, 53706, U.S.A.

²Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison, K6/428 CSC 600 Highland Avenue, Madison, Wisconsin, 53792-4675, U.S.A.

Summary

When drawing large scale simultaneous inference, such as in genomics and imaging problems, multiplicity adjustments should be made, since, otherwise, one would be faced with an inflated type I error. Numerous methods are available to estimate the proportion of true null hypotheses π_0 , among a large number of hypotheses tested. Many methods implicitly assume that the π_0 is large, that is, close to 1. However, in practice, mid-range π_0 values are frequently encountered and many of the widely-used methods tend to produce highly variable or biased estimates of π_0 . As a remedy in such situations, we propose a hierarchical Bayesian model (HBM) that produces an estimator of π_0 that exhibits considerably less bias and is more stable. Simulation studies seem indicative of good method performance even when low to moderate correlation exists among test statistics. Method performance is assessed in simulated settings and its practical usefulness is illustrated in an application to type II diabetes study.

Keywords

Bayesian Framework; Microarray Data; Mixture Model; Multiple Testing; Robustness

1 Introduction

Rapid technological advancements in genomics and image processing have created new statistical challenges in large scale simultaneous inference, and oftentimes, traditional approaches fall short in addressing such problems. Since Benjamini and Hochberg (1995) have introduced the false discovery rate (FDR) as a new error measure in multiple hypotheses testing problems, many researchers have developed new methods. Estimating the proportion of true null hypotheses, π_0 , has been of central interest because most methods rely on the correct estimation of π_0 . In general, a large proportion of true null hypotheses has been widely assumed in existing literature. For further details, see Efron *et al.* (2001), Pounds and Morris (2003), or Pounds and Cheng (2006).

However, when the proportion π_0 of true null hypotheses is moderate, such as between 0.5 and 0.8, many existing methods do not appear to perform well, possibly due to highly variable or biased estimates of π_0 . As a recent example, Idaghdour *et al.* (2008) have found differential expression percentages varying between 30% and 50%, when examining the contribution of environmental factors to variation in gene expression levels for leukocyte samples. To further emphasize their point, these authors have re-analyzed several existing

*Adin-Cristian Andrei: andrei@biostat.wisc.edu, Phone: +1 608 263 6797.

Conflict of Interest Statement The authors have declared no conflict of interest.

datasets and have uncovered percentages of differentially expressed transcripts ranging from 30% to 94%. Similar findings in a type II diabetes study in mice, conducted by Keller *et al.* (2008) at the University of Wisconsin-Madison, constitute our motivation to further research this matter. In this study there were two strains of mice: obesity-induced diabetes-resistant (B6) and diabetes-susceptible (BTBR). Gene expression levels have been obtained in six tissue types: adipose, gastrocnemius, liver, hypothalamus, islet and soleus. Interest lies in the identification of the differentially expressed (DE) genes that might be associated with progression to type II diabetes in liver tissue type. Separate comparisons of the two strains are carried out in the four groups created by age/obesity status combinations: four-week-old/lean, ten-week-old/lean, four-week-old/obese and ten-week-old/obese.

Table 1 provides separate estimates of π_0 in these four groups, using the proposed method (HBM), as well as other existing methods, all to be detailed in Section 2: Storey and Tibshirani (2003) (STOREY), Pounds and Morris (2003) (BUM), Pounds and Cheng (2004) (SPLOSH), Pounds and Cheng (2006) (PC06) and Meinshausen and Bühlmann (2005) (MEIN). These methods all suggest that the true value of π_0 is moderate in each group, that is, between 0.5 and 0.8. Declaring a gene to be DE is a decision depending upon an estimate of π_0 and the wide range of π_0 estimates in each group is a cause for concern because it is unclear which one should be relied upon. Table 2, which presents the number of DE genes in the four groups described, shows drastically varying findings. While BUM identifies at least 1300 more DE genes than SPLOSH, Storey's method produces at least 600 more DE genes than PC06. The fact that true π_0 is most likely moderate, is probably one explanation. Therefore, developing an estimator for π_0 that is robust in a variety of cases, including situations when the true π_0 is moderate, is imperative.

In the multiple testing problems, it is usually assumed that the p -value p_i of test i has a mixture density $f(p_i) = \pi_0 f_0(p_i) + \pi_1 f_1(p_i)$, where $i = 1, \dots, N$. Here, $f_0(p_i)$ is the density under the null hypothesis, $f_1(p_i)$ is the density under the alternative, and $\pi_1 = (1 - \pi_0)$ is the proportion of true alternative hypotheses out of a grand total of N hypotheses. Earlier work by Schweder and Spjøtvoll (1982) presents an informal graphical procedure to estimate π_0 . It relies on fitting a regression line with slope depending on a constant chosen after visual examination of a p -value plot, so inference is subjective. In a sense, this method can be seen as a precursor of the widely used Storey (2002). When $0 < \lambda < 1$, a naive estimator of π_0 is

$\hat{\pi}_0(\lambda) = \frac{\sum_{i=1}^N I_{\{p_i \geq \lambda\}}}{N(1 - \lambda)}$, where $I_{\{\cdot\}}$ is the indicator function. Storey (2002) suggest estimating π_0 by minimizing the mean squared error $MSE(\hat{\pi}_0(\lambda))$ for fixed λ , using bootstrapping. However, this tends to overestimate π_0 unless λ is very close to one. Therefore, Storey and Tibshirani (2003) uses a smoothing spline method to obtain an estimator of π_0 . When λ is very close to one, Storey and Tibshirani (2003)'s estimator is very close to the true π_0 . However, as Qiu *et al.* (2006) point out, their estimator could be highly variable, hence unstable. In the BUM method of Pounds and Morris (2003), the estimator $\hat{\pi}_0$ is likely to underestimate π_0 , especially when π_0 is near zero. Pounds and Cheng (2006) use $\min(1, 2 * \bar{p})$ to estimate π_0 , where $\bar{p}$ is the sample mean of the N p -values. On the other hand, the SPLOSH method uses $\min_{1 \leq i \leq N} \hat{f}(p_i)$ to estimate π_0 , where $\hat{f}(p_i)$ is an estimate of the mixture density f at p_i . Both PC06 and SPLOSH appear to overestimate π_0 , especially when π_0 is not close to 1.

Working on a related problem, Meinshausen and Bühlmann (2005) have developed methods for estimating the lower bound of the number of false null hypotheses under a general dependence structure for the p -values. Their procedure tends to produce conservative estimates of π_0 . However, our research objective is concerned with correctly classifying each individual hypothesis in large-scale testing, not with estimating the total number of

false null hypotheses. In order to address this important issue of obtaining a reliable estimate of π_0 , we develop a new methodology that builds on the parametric Bayesian method idea of Ji *et al.* (2008). The advantage resides in reducing the computational burden associated with conditional posterior distributions by modeling the conditional posterior distributions on hyperparameters proportional to posterior densities of corresponding hypotheses.

We develop a joint probability model across the set of all p -values, by assuming that they are uniformly distributed under the null hypotheses, while following a general beta distribution under the alternative hypotheses. The mixture of uniform and beta distributions has been used by other authors such as Allison *et al.* (2002) and Pounds and Morris (2003) to model the observed p -values. For further justification, see Parker and Rothenberg (1988).

In Section 2, we propose a hierarchical Bayesian model (HBM) that produces an estimator of π_0 that exhibits less bias and is more stable. Ours is an approach that naturally accommodates dependence. In the first stage of the hierarchical structure we have assumed the p -values are conditionally independent. In the higher stages, our distributional structure allows us to relate the individual hypotheses together. The resulting distribution of the p -values thus no longer requires marginal independence. This is one of the primary advantages of using a hierarchical Bayes approach that provides a systematic way of building dependencies.

The resulting multiple testing procedure is compared to Storey and Tibshirani (2003)'s method, BUM, SPLOSH, PC06 and MEIN. In Section 3, we present extensive simulation studies showing that HBM produces reliable estimates of π_0 . In Section 4, we present a data example and further discuss the matter in Section 5.

2 A Flexible Bayesian Framework

We propose a general setup of the mixture model. Let $\mathcal{P}$ be a family of distributions, $\mathcal{P}_0 \subset \mathcal{P}$, and $\mathcal{P}_1 = \{P \in \mathcal{P} : P \notin \mathcal{P}_0\}$. We formulate our testing hypothesis problem as follows:

$$\{H_i=0\} : P \in \mathcal{P}_0 \text{ versus } \{H_i=1\} : P \in \mathcal{P}_1 \quad (1)$$

where $\{H_i = 0\}$ denotes the null hypothesis and $\{H_i = 1\}$ denotes the alternative hypothesis for test i , where $i = 1, \dots, N$. We then obtain the p -values in order to test the statistical significance of each null hypothesis. Let $p_i \equiv p_i(y_i) = P(|Y_i| > |y_i| | H_i = 0)$ for the continuously distributed test statistic Y_i and the observed value y_i . Recall that the distribution of the p -value p_i has a mixture density

$$f(p_i) = (1 - \pi_1)f_0(p_i) + \pi_1 f_1(p_i), \quad (2)$$

where π_1 is the proportion of true alternative hypotheses among the N hypotheses of interest. As a reminder, $f_0(p_i)$ is the density of the p -value under the null hypothesis and $f_1(p_i)$ is the density of p -value under the alternative. We assume that $f_0(p_i)$ has the uniform density $U(0, 1)$ and that $f_1(p_i)$ follows a $Beta(\gamma\theta, \gamma(1 - \theta))$ distribution with mean θ , where $\gamma > 0$ and $0 < \theta < 1$. While Allison *et al.* (2002) modeled the $f_1(p_i)$ with a mixture of several beta distributions, Pounds and Morris (2003) and Liao *et al.* (2004) used a restricted beta distribution. We assume that θ and γ have distributions:

$$\theta \sim Beta(\eta\xi, \eta(1 - \xi)), \text{ with mean } \xi \quad (3)$$

$$\log \gamma \sim N(\log \mu, \sigma^2), \quad (4)$$

where $\eta > 0$ and $0 < \xi < 1$. We assume a beta prior distribution for π_1 ,

$$\pi_1 \sim \text{Beta}(\tau\delta, \tau(1-\delta)), \text{ with mean } \delta \quad (5)$$

where $\tau > 0$ and $0 < \delta < 1$. Let the *Bernoulli*(π_1)-distributed random variable J_i be equal to 1 if the i^{th} alternative hypothesis is true, and be equal to 0, otherwise. Define $\mathbf{J}$ to be the vector $(J_1, \dots, J_N)$ and $\mathbf{p} = (p_1, \dots, p_N)$. Hence, assuming independence of the p -values across all hypotheses, their joint posterior density is

$$\begin{aligned} P(\mathbf{J}, \theta, \gamma, \pi_1 | \mathbf{p}) &= \text{Beta}(\eta\xi, \eta(1-\xi)) N(\log \mu, \sigma^2) \frac{1}{\gamma} \text{Beta}(\tau\delta, \tau(1-\delta)) \\ &\times \prod_{i=1}^N \left\{ [(1-\pi_1) + \pi_1 \text{Beta}(\gamma\theta, \gamma(1-\theta))] \pi_1^{J_i} (1-\pi_1)^{1-J_i} \right\}. \end{aligned} \quad (6)$$

The computational burden associated with the implementation of this Bayesian model could be significantly reduced by the suggestions of Ji *et al.* (2008). In that, we conduct the hypothesis testing based on the posterior probabilities $P(J_i | \mathbf{p})$. Under the alternative hypotheses, assume that

$$\begin{aligned} P(\gamma | \mathbf{p}, \theta, \pi_1, \mathbf{J}) &\propto \exp \left\{ -\frac{(\log \gamma - \log \mu)^2}{2\sigma^2} \right\} \frac{1}{\gamma} \\ &\times \prod_{\{i: H_i=1\}}^N \frac{\Gamma(\gamma)}{\Gamma(\gamma\theta)\Gamma(\gamma(1-\theta))} p_i^{\gamma\theta-1} (1-p_i)^{\gamma(1-\theta)-1}, \end{aligned} \quad (7)$$

$$\begin{aligned} P(\theta | \mathbf{p}, \gamma, \pi_1, \mathbf{J}) &\propto \theta^{\eta\xi-1} (1-\theta)^{\eta(1-\xi)-1} \\ &\times \prod_{\{i: H_i=1\}}^N \frac{\Gamma(\gamma)}{\Gamma(\gamma\theta)\Gamma(\gamma(1-\theta))} p_i^{\gamma\theta-1} (1-p_i)^{\gamma(1-\theta)-1}, \end{aligned} \quad (8)$$

$$\begin{aligned} P(\pi_1 | \mathbf{J}) &\propto P(\mathbf{J} | \pi_1) P(\pi_1) \\ &\propto \pi_1^{\sum_{i=1}^N J_i} (1-\pi_1)^{N-\sum_{i=1}^N J_i} \pi_1^{\tau\delta-1} (1-\pi_1)^{\tau(1-\delta)-1} \\ &= \pi_1^{\sum_{i=1}^N J_i + \tau\delta - 1} (1-\pi_1)^{N-\sum_{i=1}^N J_i + \tau(1-\delta) - 1} \\ &\sim \text{Beta}(\sum_{i=1}^N J_i + \tau\delta, N - \sum_{i=1}^N J_i + \tau(1-\delta)) \end{aligned} \quad (9)$$

The conditional posterior distribution for the indicator J_i is equal to

$$P(J_i | p_i, \theta, \gamma, \pi_1) \sim \text{Bernoulli} \left\{ \frac{\pi_1 \text{Beta}(\gamma\theta, \gamma(1-\theta))}{(1-\pi_1) + \pi_1 \text{Beta}(\gamma\theta, \gamma(1-\theta))} \right\}. \quad (10)$$

To draw parameters from these conditional posterior distributions, we use Markov Chain Monte Carlo (MCMC) techniques. A hybrid Gibbs/Metropolis sampler continuously updates γ , θ , π_1 and J_i , $i = 1, \dots, N$ from the above conditional distributions. This algorithm is an adaptation of a random walk that uses an acceptance/rejection rule to converge to the specified target distribution. For instance, we generate a candidate γ^* value from a normal

jumping kernel centered on the previous value γ^{t-1} to update γ . Then we accept the candidate value for the current value γ^t with the probability $\min(1, r)$, where

$$r = \frac{p(\gamma^* | \mathbf{p}, \theta, \pi_1, \mathbf{J})}{p(\gamma^{t-1} | \mathbf{p}, \theta, \pi_1, \mathbf{J})}, \quad (11)$$

given the previous value γ^{t-1} . Following Gelman *et al.* (2004), the ratio of between- and within- chain variances is used to monitor chain convergence. We consider five MCMC chains, with different initial parametrizations. First, we compute the autocorrelation for the respective Markov chain at lags (1, 5, 10), every 50 iterations. If all these autocorrelations are less than 0.5, we compute the ratio using all draws. Otherwise, we thin out the chains or use the second half of draws to compute the chain convergence statistics. We repeat until the chain convergence statistic is less than 1.05. It appears that 200 iterations are sufficient for chain convergence. After the chains have converged, we estimate γ , θ , π_1 via the mean of the last iterates in five chains generated. In addition, π_0 is estimated by $1 - \pi_1$. Guidelines on how to choose model hyperparameters are further discussed in the simulation section.

To test the significance of the i^{th} null hypothesis $\{H_i = 0\}$, we compute Storey's q -value, which depends on the estimated true null proportion π_0 . As such, if $p_{(1)}, \dots, p_{(N)}$ are the ordered p -values, then the estimate of the q -value is

$$\hat{q}_{(i)} = \min_{\{j \geq i\}} t_{(j)}, \quad (12)$$

where $t_{(j)} = \frac{\hat{\pi}_0 p_{(j)}}{\sum_{i=1}^N I_{\{p_i \leq p_{(j)}\}} / N}$ and $i = 1, \dots, N$. The proposed HBM method rejects the null hypotheses whose estimated q -values are less than or equal to a pre-specified FDR level q .

3 Simulation Study

We carry out extensive simulations to illustrate our approach. In the mice data example, the various true null proportions are estimated to be between 0.5 and 0.8, so we devise simulations resembling the example. We consider four different scenarios: the first one is concerned with independent hypotheses, while the other three deal with dependent hypotheses, in a sense to be detailed shortly. Six values of π_0 have been considered: 0.5, 0.6, 0.7, 0.8, 0.95 and 0.99. Under two experimental conditions (treatment and control), expression levels for N genes are generated. In the first three scenarios $N = 10,000$, while in the fourth one $N = 1,000$. Per group sample size is equal to 10 and each scenario is replicated 1,000 times. In each group and for each gene, samples are drawn from two different normal distributions, to be detailed subsequently.

We devise the first scenario as follows. For the differentially expressed (DE) genes, 10 samples are drawn from a Gaussian $N(\mu_t, 2)$ distribution in the treatment group, and from a $N(0, 2)$ distribution in the control group, where μ_t is generated from a uniform $U(2, 3)$. For the non-DE genes, expression levels are drawn from a $N(0, 2)$ in each group. To compare the two groups, a two-sample t-test is used for each of the 10,000 genes. When the ratio of the sample standard deviations is between 0.5 and 2, we use the t-test with a pooled variance; otherwise Welch's t-test is used.

Hyperparameters ξ and η represent the expected value for θ , and an indicator variable reflecting the degree of trust in ξ , respectively. Small η and large σ lead to small weight on the hyperparameters, in terms of precision. Parameter τ plays the role of prior sample size,

with a large τ value leading to an increased weight on the prior. Hyperparameter μ is corresponding to the variation of the beta distribution, under the alternative hypothesis. Small values of μ lead to fast convergence of the parameter estimates for γ and θ . We let δ

be $1 - \min\{1, 4 * (\delta_0)\}$, where $\delta_0 = \frac{\sum_{i=1}^N p_i I_{\{p_i \leq 0.5\}} + \sum_{i=1}^N (1 - p_i) I_{\{p_i > 0.5\}}}{N}$. In addition, we set $\mu = 1$, $\eta = 1$, $\sigma^2 = 100$, and $\tau = 1000 * N$. In all four scenarios, 200 iterations suffice for algorithm convergence. To implement BUM, SPLOSH and PC06, we use programming codes posted on www.stjude.com/depts/biostat and employ the default options. For Storey's method, R package “*q-value*”, with default options, is used. To implement Meinshausen's method for estimating the lower bound of the number of correct rejections, we use *howmany* function for independent hypotheses and *howmany_dependent* function for dependent hypotheses with default options in R package “*howmany*”.

We compute the average π_0 across the 1,000 replications, the squared empirical mean squared errors ($\sqrt{MSE}$) for π_0 and average computation time in seconds. Table 3 presents simulation results for the first scenario.

As well as the proposed method HBM estimates π_0 unbiasedly, it has the smallest MSE when π_0 is between 0.5 and 0.95. When π_0 is 0.99, HBM produces unbiased π_0 estimates although the MSE becomes slightly larger than PC06 and MEIN. Note that BUM and SPLOSH produce biased estimates on π_0 in most cases. Because MEIN does not require permutations in the first scenario, it is computationally efficient. To assess the performance of the proposed method in more complex settings, we have devised three scenarios that involve correlated data among genes. Storey *et al.* (2004) consider pairwise correlations between test statistics. Correlations occur naturally at the data level and are reflected in the test statistics. Let Σ_{ij} be the pairwise correlation between gene i and gene j , with $1 \leq i < j \leq N$. Denote by N_1 and N_0 be the number of true alternative hypotheses and the number of true null hypotheses among N hypotheses, respectively. We generate correlated observations among genes for the treatment group corresponding to true alternative hypotheses. Note that sample size is equal to 10 per group. Vectors $X_1, X_2, \dots, X_{10}$ are drawn from multivariate normal distribution $MVN(\mu_t * 1_{N_1 \times 1}, \Sigma_{N_1 \times N_1})$ where $\mu_t \sim U(2, 3)$ and $1_{N_1 \times 1}$ is a N_1 dimensional vector whose elements are 1. Similarly, we generate control group observations for true alternative hypotheses, $Y_1, Y_2, \dots, Y_{10}$ from $MVN(0 * 1_{N_1 \times 1}, \Sigma_{N_1 \times N_1})$. For the remaining observations for true null hypotheses, 20 vectors $Z_1, Z_2, \dots, Z_{20}$ are drawn from $MVN(0 * 1_{N_0 \times 1}, \Sigma_{N_0 \times N_0})$. Therefore, the data matrix (A) for correlated observations is

$$A \equiv \left(\begin{array}{c|c} X_1 \cdots X_{10} & Y_1 \cdots Y_{10} \\ \hline Z_1 \cdots Z_{10} & Z_{11} \cdots Z_{20} \end{array} \right)$$

In the type II diabetes mice example, we randomly select 10,000 tests in each group to get a sense of the existing pairwise correlation levels Σ_{ij} . Approximately 75 % of these pairwise correlations are between -0.3 and 0.3 and their average is virtually zero. Inspired by these data example findings, we consider $\Sigma_{ij} = 0.2, 0.4$ and 0.6 for the second scenario, the third scenario and the fourth scenario, respectively. We obtain p -values using the Wilcoxon rank sum test for each gene. Then we apply HBM and the other methods described to the observed p -values. We employ the same default options for STOREY, BUM, SPLOSH and PC06 as in the first scenario. For MEIN, which is designed for situations where correlated tests are encountered, we plug the data matrix A in the function “*howmany_dependent*”, and we use the Wilcoxon rank sum test for testing option. In addition, we apply 400 permutations by default (more details are available in (Meinshausen, 2007)). Table 4 and

Table 5 show the simulation results. HBM performs very well in all π_0 cases, while STOREY and MEIN estimate π_0 conservatively when π_0 is between 0.5 and 0.8. However, when π_0 's are 0.99, STOREY and MEIN produce liberal or slightly Conservative estimates ($\pi_0 = 0.92$ and $\pi_0 = 1.0$, respectively) in Table 5. In addition, HBM is at least 100 times faster than MEIN, which is, by design, more computationally intensive since it involves permutations.

The fourth scenario was similar to the second and third one, except for the tests being even more strongly correlated at level $\Sigma_{ij} = 0.6$, as well as the smaller number of genes ($N = 1000$) considered. Table 6 shows the simulation results. When π_0 is moderate, HBM tends to estimates π_0 unbiasedly. Even for $\pi_0=0.99$, HBM leads to very small negative bias (-0.03). On the other hand, Meinhausen's is somewhat conservative although the bias decreases as π_0 increases. Note BUM and SPLOSH fail to estimate π_0 in the strongly correlated hypotheses. HBM's advantage in terms of computational efficiency is maintained. In this simulation scenario, CPU time tends to be at least 5 times shorter than for 10,000 hypotheses.

4 Type II Diabetes Example

In a mice diabetes study, Keller *et al.* (2008) have studied gene expression levels under three experimental conditions: four-versus ten-week-old, lean versus obese, and B6 versus BTBR strain. The B6 mice do not develop diabetes, regardless of their obesity status. In contrast, the BTBR mice become severely diabetic by ten weeks of age, as long as they are obese. Using unsupervised hierarchical clustering modeling techniques, they identify gene expression modules associated with age and obesity within each strain type. They restrict their interest in comparisons within each strain without actually comparing the two strain types directly under the same age and obesity conditions. However, insulin-resistance or progression to type II diabetes may depend on both obesity and age. We directly compare gene expressions in B6 and BTBR strains, and that might be associated with progression to type II diabetes. To do so, we conduct separate comparisons of the two strain types under each of the following four age/obesity status combinations: four-week-old/lean, ten-week-old/lean, four-week-old/obese and ten-week-old/obese. For each of the two strains (B6 or BTBR), there were five male mice in each of the four groups, hence a total number of 40 mice. In order to carry out gene expression profiling, Rosetta Inpharmatics (Merck & Co.) performs array hybridizations, including RNA preparations. The custom ink-jet microarrays of Agilent Technologies, which are composed of 4, 732 control probes and 39,558 non-control oligonucleotides are used (Keller *et al.*, 2008). Of the six tissue types available, we focus on the liver tissue only. The custom ink-jet microarrays of Agilent Technologies provide three kinds of metrics: Intensity2, Mlratio, and Reratio. While Intensity2 is a non-normalized metric of probe performance, Mlratio is a normalized metric that uses a strain-specific reference pool. Since we are interested in strain comparisons, we use the Reratio metric for the expression values, which utilizes a synthetic reference pool to normalize between the strains. We excluded genes which have at least one fluorescence intensity missing for data screening. To construct a list of DE genes in each of the four groups, we carry out a two-sample t-test for each gene. As in simulations, if the ratio of sample standard deviations is between 0.5 and 2, we use the t-test with pooled variance; otherwise, Welch's t-test is used. We apply our methodology to the p -values obtained in this fashion. For comparison, we also present the results obtained based on the other methods described in the simulation studies. After 2,000 iterations the Markov chains have converged well. The between vs. within-chain variances ratios used to check out the convergence of iterated simulations for π_0 were equal to 1.02, 1.01, 1.004 and 1.003 in the four groups. We estimate π_0 using mean values of the last draws in 5 chains. Table 1 presents the estimates of true null proportions π_0 in each of the four groups. The HBM yields a higher π_0 than Storey's method, SPLOSH, and MEIN, while BUM is producing the smallest π_0 in three of the four groups.

On the other hand, the PC06 method produces estimates that are likely to be the most conservative ones, compared to the other methods. The same holds in regard to Table 2, which presents the number of genes declared DE at a 0.05 FDR level. While SPLOSH, and PC06 produce quite conservative results, BUM produces the largest number of DE genes in all groups.

To check model goodness-of-fit, we use quantile-quantile (QQ) plots and also compare the fitted density to the histogram of p -values. On the QQ plots (not shown), we observe a 45 degree straight line in each of the four groups, indicative of good model fit. Histograms presented in Figure 1 indicate that the HBM model fits the observed p -values reasonably well.

5 Discussion

While there is substantial literature on the large scale multiple hypotheses testing, it is mainly concerned with situations when π_0 is very large, approaching 1. The proposed HBM method provides a reasonable solution, in a flexible Bayesian framework, to overestimation/underestimation issues that may occur when π_0 is between 0.5 and 0.8. While Do *et al.* (2005) also propose a Bayesian mixture model, we have significantly reduced the computation time, compared to other Bayesian methods such as Do *et al.* (2005) or Liao *et al.* (2004). We propose a hierarchical Bayesian model (HBM) which is an approach that naturally accommodates dependence. In the first stage of the hierarchical structure we have assumed the p -values are conditionally independent. In the higher stages, our distributional structure allows us to relate the individual hypotheses together. The resulting distribution of the p -values thus no longer requires marginal independence. This is one of the primary advantages of using a hierarchical Bayesian model approach that provides a systematic way of building dependencies. Although pairwise intra-tests correlation appears to be important in microarray studies, it may be less of a problem in areas such as functional magnetic resonance imaging or time-of-flight spectroscopy, where large-scale multiple testing is also required. The method developed, HBM, shows good performance even in scenarios with realistically high pairwise correlated data.

HBM produces an estimator of π_0 that exhibits less bias and is more stable than other compared methods. Due to reliable π_0 estimation, HBM controls the FDR more tightly and is more powerful in testing all hypotheses simultaneously. In order to compute the FDR, we can also use the posterior probability of the null hypothesis being true, as Newton *et al.* (2004) suggested. The HBM approach makes use of the raw p -values only. Hence, it is potentially applicable to a wide variety of situations in which p -values are obtained based on nonparametric tests or on ANOVA models. In conclusion, the robust Bayesian approach proposed appears to be very useful in obtaining more precise estimates of π_0 and in selecting differentially expressed genes.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgments

Adin-Cristian Andrei's research is supported in part by following NIH grants: P30 CA014520-33, UL1 RR025011 and R21 CA132267. Kam-Wah Tsui's work was supported in part by NSF grant DMS-0604931. The authors would like to thank Mark Keller and Alan D. Attie for providing the dataset.

References

- Allison D, Gadbury G, Heo M, Fernandez J, Lee C, A PT, R W. A mixture model approach for the analysis of microarray gene expression data. *Computational Statistics & Data Analysis*. 2002; 39:1–20.
- 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*. 1995; 57:289–300.
- Do KA, Muller P, Tang F. A Bayesian mixture model for differential gene. *Journal of the Royal Statistical Society Series C*. 2005; 54:627–644.
- Efron B, Tibshirani R, Storey JD, Tusher V. Empirical Bayes analysis of a microarray experiment. *Journal of the American Statistical Association*. 2001; 96:1151–1160.
- Gelman, A.; Carlin, JB.; S, SH.; Rubin, DB. Chapman & Hall/CRC. 2nd. Bayesian data analysis; 2004.
- Idaghdour Y, Storey JD, Jadallah SJ, Gibson G. A genome-wide gene expression signature of environmental geography in leukocytes of moroccan amazighs. *PLoS Genetics*. 2008; 4:e1000052. [PubMed: 18404217]
- Ji Y, Lu Y, Mills BG. Bayesian models based on test statistics for multiple hypothesis testing problems. *Bioinformatics*. 2008; 24:943–949. [PubMed: 18245123]
- Keller MP, Choi Y, Wang P, Davis DB, Rabaglia ME, Oler AT, Stapleton DS, argmann C, Schueler KL, S E, Steinberg HA, Neto EC, Kleinhanz R, Turner S, Hellerstein MK, Schadt EE, Yandell BS, Kendzioriski C, Attie AD. A gene expression network model of type 2 diabetes links cell cycle regulation in islets with diabetes susceptibility. *Genome Research*. 2008; 18:706–716. [PubMed: 18347327]
- Liao JG, Lin Y, Selvanayagam ZE, Shih WJ. A mixture model for estimating the local false discovery rate in DNA microarray analysis. *Bioinformatics*. 2004; 20:2694–2701. [PubMed: 15145810]
- Meinshausen, N. Package "howmany" - a lower bound for the number of correct rejections. 2007. <http://cran.r-project.org>
- Meinshausen N, Bühlmann P. Lower bounds for the number of false null hypotheses for multiple testing of associations under general dependence structures. *Biometrika*. 2005; 92:893–907.
- Newton MA, Noueiry A, Sarkar D, Ahlquist P. Detecting differential gene expression with a semiparametric hierarchical mixture method. *Biostatistics*. 2004; 5:155–176. [PubMed: 15054023]
- Parker RA, Rothenberg RB. Identifying important results from multiple statistical tests. *Statistics in Medicine*. 1988; 7:1031–1043. [PubMed: 3206001]
- Pounds S, Cheng C. Improving false discovery rate estimation. *Bioinformatics*. 2004; 20:1737–1745. [PubMed: 14988112]
- Pounds S, Cheng C. Robust estimation of the false discovery rate. *Bioinformatics*. 2006; 22:1979–1987. [PubMed: 16777905]
- Pounds S, Morris SW. Estimating the occurrence of false positives and false negatives in microarray studies by approximating and partitioning the empirical distribution of p -values. *Bioinformatics*. 2003; 19:1236–1242. [PubMed: 12835267]
- Qiu X, Xiao Y, Gordon A, Yakovlev A. Assessing stability of gene selection in microarray data analysis. *BMC Bioinformatics*. 2006; 7:50. [PubMed: 16451725]
- Schweder T, Spjøtvoll E. Plots of p -values to evaluate many tests simultaneously. *Biometrika*. 1982; 69:493–502.
- Storey JD. A direct approach to false discovery rate. *Journal of the Royal Statistical Society Series B*. 2002; 64:479–498.
- Storey JD, Tibshirani R. Statistical significance for genomewide studies. *Proceedings of the National Academy of Science of the United States of America*. 2003; 100:9440–9445.
- Storey JD, Taylor JE, Siegmund D. Strong control, conservative point estimation and simultaneous conservative consistency of false discovery rates: A unified approach. *Journal of the Royal Statistical Society, Series B*. 2004; 66:187–205.

Fig. 1.
Type II diabetes mice example: histogram of p -values and fitted mixture density using the HBM model.

Table 1
Proportion of non-DE expressed genes in the Type II diabetes mice example, using HBM, STOREY, BUM, SPLOSH, PC06 and MEIN

Mice Group	Number of genes	HBM	STOREY	BUM	SPLOSH	PC06	MEIN
4 week-old, Lean	39240 genes	0.744	0.556	0.526	0.570	0.783	0.596
10 week-old, Lean	39492 genes	0.754	0.600	0.583	0.596	0.784	0.650
4 week-old, Obese	38959 genes	0.768	0.515	0.538	0.518	0.808	0.574
10 week-old, Obese	38881 genes	0.688	0.520	0.502	0.546	0.726	0.562

Table 2
Number of DE genes in the Type II diabetes mice example at 5 % FDR level, using HBM, STOREY, BUM, SPLOSH, PC06 and MEIN

Mice Group	Number of genes	HBM	STOREY	BUM	SPLOSH	PC06	MEIN
4 week-old, Lean	39240 genes	2111	2752	3117	1064	2013	2574
10 week-old, Lean	39492 genes	3031	3570	3795	1497	2946	3345
4 week-old, Obese	38959 genes	1365	1952	2138	826	1332	1737
10 week-old, Obese	38881 genes	5908	7006	7168	5091	5777	6641

Table 3

Independent observations and N=10000, simulation scenario 1 results, using HBM, STOREY, BUM, SPLOSH, PC06 and MEIN (computation time (CPU) in seconds).

π_0		HBM	STOREY	BUM	SPLOSH	PC06	MEIN
0.5	MEAN	0.50	0.51	0.36	0.62	0.53	0.52
	$\sqrt{MSE} \times 10^2$	0.51	1.69	13.69	16.03	2.76	2.02
	CPU Time	18.45	0.12	0.07	0.03	2.72	0.15
0.6	MEAN	0.60	0.59	0.47	0.71	0.63	0.62
	$\sqrt{MSE} \times 10^2$	0.51	1.84	12.98	14.32	2.24	1.84
	CPU Time	15.67	0.13	0.07	0.04	2.86	0.15
0.7	MEAN	0.70	0.69	0.58	0.79	0.72	0.72
	$\sqrt{MSE} \times 10^2$	0.53	1.97	11.59	11.08	1.73	1.67
	CPU Time	14.91	0.12	0.06	0.03	2.52	0.15
0.8	MEAN	0.80	0.79	0.70	0.85	0.81	0.82
	$\sqrt{MSE} \times 10^2$	0.54	2.10	9.46	6.79	1.23	1.43
	CPU Time	15.45	0.12	0.06	0.03	2.59	0.15
0.95	MEAN	0.95	0.95	0.91	0.92	0.95	0.96
	$\sqrt{MSE} \times 10^2$	0.58	2.23	4.02	4.31	0.66	0.87
	CPU Time	16.84	0.14	0.08	0.03	3.63	0.17
0.99	MEAN	0.99	0.99	1	0.95	0.98	0.99
	$\sqrt{MSE} \times 10^2$	0.56	0.85	2.90	4.19	0.50	0.15
	CPU Time	13.92	0.12	0.01	0.04	2.94	0.16

Table 4

Pairwise correlated observations at level 0.2 and N=10000, simulation scenario 3 results, using HBM, STOREY, BUM, SPLOSH, PC06 and MEIN (computation time (CPU) in seconds).

π_0		HBM	STOREY	BUM	SPLOSH	PC06	MEIN
0.5	MEAN	0.49	0.63	0.37	0.60	0.47	0.62
	$\sqrt{MSE} \times 10^2$	2.46	16.48	12.91	15.73	6.19	14.66
	CPU Time	31.39	0.11	0.08	0.02	0.03	2970.2
0.6	MEAN	0.59	0.76	0.48	0.63	0.57	0.69
	$\sqrt{MSE} \times 10^2$	2.64	20.09	12.73	11.03	7.74	11.32
	CPU Time	31.44	0.12	0.08	0.02	0.03	2938.2
0.7	MEAN	0.69	0.88	0.59	0.64	0.68	0.77
	$\sqrt{MSE} \times 10^2$	2.77	22.56	11.78	12.26	8.13	8.90
	CPU Time	22.44	0.11	0.06	0.02	0.03	2533.3
0.8	MEAN	0.79	0.95	0.71	0.65	0.77	0.85
	$\sqrt{MSE} \times 10^2$	3.25	18.37	10.09	18.50	8.37	6.01
	CPU Time	24.96	0.10	0.06	0.02	0.03	2630.4
0.95	MEAN	0.94	0.98	0.89	0.70	0.92	0.97
	$\sqrt{MSE} \times 10^2$	4.07	8.64	9.39	26.61	9.90	2.30
	CPU Time	23.26	0.10	0.08	0.02	0.03	2616.3
0.99	MEAN	0.98	0.98	1.00	0.75	0.99	0.99
	$\sqrt{MSE} \times 10^2$	3.88	5.92	13.82	26.22	4.29	0.85
	CPU Time	26.89	0.12	0.03	0.02	0.06	2741

Table 5

Pairwise correlated observations at level 0.4 and N=10000, simulation scenario 4 results, using HBM, STOREY, BUM, SPLOSH, PC06 and MEIN (computation time (CPU) in seconds).

π_0		HBM	STOREY	BUM	SPLOSH	PC06	MEIN
0.5	MEAN	0.49	0.63	0.37	0.51	0.53	0.63
	$\sqrt{MSE} \times 10^2$	4.37	22.95	14.53	16.16	9.79	16.55
	CPU Time	12.50	0.12	0.09	0.02	0.03	3145.5
0.6	MEAN	0.59	0.76	0.47	0.53	0.63	0.70
	$\sqrt{MSE} \times 10^2$	5.01	27.59	15.33	17.34	9.87	13.09
	CPU Time	11.01	0.15	0.11	0.02	0.04	3544.1
0.7	MEAN	0.69	0.84	0.58	0.52	0.73	0.78
	$\sqrt{MSE} \times 10^2$	6.10	26.24	14.17	24.17	10.03	10.01
	CPU Time	8.95	0.11	0.07	0.02	0.03	2648.7
0.8	MEAN	0.79	0.88	0.70	0.51	0.82	0.86
	$\sqrt{MSE} \times 10^2$	5.71	23.04	15.34	33.78	12.45	10.46
	CPU Time	9.12	0.12	0.07	0.02	0.03	2807.2
0.95	MEAN	0.94	0.91	0.87	0.54	0.95	0.97
	$\sqrt{MSE} \times 10^2$	8.17	21.18	17.45	45.89	12.78	5.67
	CPU Time	7.28	0.10	0.08	0.02	0.03	2652.3
0.99	MEAN	0.97	0.92	1.00	0.54	0.99	1.00
	$\sqrt{MSE} \times 10^2$	7.51	19.69	9.69	49.54	7.88	0.92
	CPU Time	7.02	0.11	0.05	0.02	0.05	2949

Table 6

Pairwise correlated observations at level 0.6 and N=1000, simulation scenario 4 results, using HBM, STOREY, BUM, SPLOSH, PC06 and MEIN (computation time (CPU) in seconds).

π_0		HBM	STOREY	BUM	SPLOSH	PC06	MEIN
0.5	MEAN	0.49	0.61	0.95	0.43	0.52	0.58
	$\sqrt{MSE} \times 10^2$	6.21	32.01	72.18	19.96	7.34	12.23
	CPU Time	1.47	0.01	0.01	0.01	0.04	217.67
0.6	MEAN	0.59	0.72	0.96	0.42	0.63	0.67
	$\sqrt{MSE} \times 10^2$	7.46	33.62	72.14	25.45	8.22	10.66
	CPU Time	1.36	0.02	0.01	0.01	0.04	176.55
0.7	MEAN	0.69	0.77	0.95	0.41	0.73	0.76
	$\sqrt{MSE} \times 10^2$	8.35	32.04	72.66	33.73	9.49	8.99
	CPU Time	1.45	0.01	0.01	0.01	0.03	179.7
0.8	MEAN	0.79	0.79	0.98	0.42	0.82	0.84
	$\sqrt{MSE} \times 10^2$	10.09	29.60	72.53	41.71	12.03	6.80
	CPU Time	1.10	0.01	0.01	0.01	0.04	182.49
0.95	MEAN	0.93	0.84	0.99	0.44	0.95	0.97
	$\sqrt{MSE} \times 10^2$	12.76	29.53	64.17	54.61	10.23	2.50
	CPU Time	1.02	0.01	0.01	0.01	0.03	167.99
0.99	MEAN	0.96	0.83	1.00	0.38	0.95	1.00
	$\sqrt{MSE} \times 10^2$	12.80	32.68	6.41	64.19	11.45	1.30
	CPU Time	0.97	0.01	0.01	0.01	0.02	157.91