BIOINFORMATICS

Vol. 00 no. 00 2005
Pages 1–7

Multivariate methods for identifying differentially expressed genes

Bret M. Hanlon [1]*and Anand N. Vidyashankar [1]

[1] Department of Statistical Science, Cornell University, Ithaca, NY 14853, USA.

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

Motivation: Univariate testing procedures remain the most common way to identify differentially expressed genes (DEGs). Univariate techniques suffer from the multiple comparison problem and reduced power, because they fail to account for gene interaction. Motivated by these issues, we adopt a multivariate procedure. Namely, we utilize the sup-norm test, which was specifically developed for high dimensional, low sample size problems. We propose an algorithm which repeatedly applies the sup-norm test to screen for DEGs.

Results: We evaluate our methodology with both simulated and experimental data. Our simulation studies establish the validity of the sup-norm statistic in terms of Type I error and power. With simulated data sets, the screening algorithm retains the majority of DEGs under a variety of experimental conditions. We also used our methodology to analyze the publicly available ApoAI knockout data set. Our algorithm identified the biologically significant genes, as discussed by other authors (Callow *et al.*, 2000; Smyth, 2004).

Availability: A set of Matlab functions used to implement the proposed methodology is available at *Bioinformatics* online.

Contact: bmh35@cornell.edu

1 INTRODUCTION

Multiple hypothesis testing is an important statistical problem that is an object of intense study in contemporary science. This is primarily due to the collection of high dimensional data from scientific experiments. A prototypical case of this phenomenon is microarray data, although similar issues are present in other genomic data. Analysis of gene expression microarray data poses significant challenges since they are not only characterized by high dimensions but also by small sample sizes (Leung and Cavalieri, 2003). Because the standard notation for the number of arrays (available samples) is n and the number of genes (dimension of the data) is p, this problem is frequently referred to as the *large p, small n* problem.

In the context of microarrays, a variety of procedures have been proposed for identifying differentially expressed genes (DEGs). These include methods based on modified t-statistics, fold change methods, linear models, and Bayesian analysis. Dudoit *et al.* (2002b) provides a survey of these commonly used statistical methods; Dudoit and van der Laan (2008) is also a useful resource for this material.

*To whom correspondence should be addressed.

A major drawback of the methods listed above is that they are all essentially univariate. Lu *et al.* (2005) describe several of the disadvantages of using univariate methods for identifying DEGs; we summarize the key points here. Univariate methods for high dimensional data suffer from the problem of multiple comparisons (Dudoit and van der Laan, 2008). Furthermore, the power of a univariate hypothesis test is reduced because it does not account for correlations between the genes. To account for gene interactions it is natural to adopt multivariate techniques, because test statistics in multivariate procedures are functions of the covariance matrix. Szabo *et al.* (2003), Lu *et al.* (2005), and Kim *et al.* (2005) have all utilized this idea and developed multivariate procedures based on Hotelling's T^2 statistic to identify DEGs in two-sample problems. Tsai and Chen (2009) extended these ideas to the k sample problem ($k \geq 2$) by proposing a modified multivariate analysis of variance solution to the problem. Their work also addresses the important question of identifying associations in gene pathways.

A difficulty in adopting multivariate techniques in gene expression problems is accurate estimation of the sample covariance matrix. In fact, in the large p, small n setting, the sample covariance matrix is often singular. Therefore, new techniques are needed to define appropriate multivariate test statistics. Recently, Kuelbs and Vidyashankar (2009) introduced an alternative multivariate method for testing multiple hypotheses and justified their methods using both large sample theory and simulations. We adopt their ideas and develop a novel multivariate procedure for identifying DEGs in both one and two sample problems. Our method is easy to implement and we show via simulations that it possesses nominal statistical properties.

This article proposes a multivariate screening procedure for identifying DEGs in one and two sample problems. The procedure is based on a multivariate test statistic recently developed by Kuelbs and Vidyashankar (2009). We illustrate the usefulness of our screening algorithm with extensive simulation experiments. Additionally, we use our procedure to identify DEGs in the ApoAI knockout experiment described by Callow *et al.* (2000).

2 METHODS

In this section we describe the multivariate procedure, which is based on the absolute maximum of the observed mean vector. We first present the one-sample formulation of the problem, and then extend it to the two-sample problem. Finally, we describe our screening algorithm which repeatedly performs the multivariate test with the goal of identifying DEGs.

2.1 One-Sample Formulation

To explain the test developed in Kuelbs and Vidyashankar (2009), we first recall some definitions from multivariate analysis. Let $\mathbf{x}$ be a vector in $\mathbb{R}^p$. For $1 \leq \rho \leq \infty$, the ℓ^ρ norms are defined by

$$\|\mathbf{x}\|_\rho = \begin{cases} \left(\sum_{j\geq 1} |x_j|^\rho\right)^{\frac{1}{\rho}} & \text{if } 1 \leq \rho < \infty, \\ \max_{1\leq j\leq p} |x_j| & \text{if } \rho = \infty \end{cases}$$

(see for example Friedman (1982)). We will refer to $||x||_\infty$ as the sup-norm.

We introduce notation to describe our test statistic. We assume that there are n arrays and p genes. Let $X_{i,j}$ represent the expression level of gene j from array i. Then, $\mathbf{X}_i = (X_{i,1}, ..., X_{i,p})^t$ represents the expression data for array i. We assume that $\mathbf{X}_1, ..., \mathbf{X}_n$ are independent and identically distributed (i.i.d.) random vectors with mean μ. Furthermore, let $\bar{\mathbf{X}}$ denote the p dimensional vector of averaged expression levels; that is, $\bar{\mathbf{X}} = \left(\bar{X}_1, ..., \bar{X}_p\right)^t$, where $\bar{X}_j = \frac{1}{n}\sum_{i=1}^n X_{i,j}$, $j = 1, ..., p$.

Let $\mu_0 \in \mathbb{R}^p$ be a p-dimensional mean vector. To test the null hypothesis, $H_0 : \mu = \mu_0$, consider the test statistic of the form, $T_\rho \equiv \|\sqrt{n}\left(\bar{\mathbf{X}} - \mu_0\right)\|_\rho$. Under regularity conditions, Kuelbs and Vidyashankar (2009) prove the asymptotic normality (in large p, small n framework) of $\sqrt{n}\left(\bar{\mathbf{X}} - \mu_0\right)$. Now, let $\mathbf{\Sigma}$ denote the covariance matrix of the data, that is $cov\left(\mathbf{X}_1\right) = \mathbf{\Sigma}$. Then, informally, the asymptotic normality result gives $\sqrt{n}\left(\bar{\mathbf{X}} - \mu_0\right) \approx N_p\left(\mathbf{0}, \mathbf{\Sigma}\right)$. Using the continuous mapping theorem gives the asymptotic normality of T_ρ,

$$T_\rho \approx \|N_p\left(\mathbf{0}, \mathbf{\Sigma}\right)\|_\rho. \tag{1}$$

Of course, if the $\mathbf{X}_i$ are random samples from a multivariate normal distribution, then these statements are no longer approximate; the distributions are exactly equal to the specified norm of the corresponding normal distribution. We focus on T_∞, which we refer to as the sup-norm (SN) statistic. One of the strengths of this procedure is that the results continue to hold (in an asymptotic sense) even if the underlying distribution is non-normal.

The statistic is based on the following intuitive idea. For each gene, compute the average expression level across the replicates; assuming there are p genes the resulting mean vector will be an element of $\mathbb{R}^p$. When concerned with finding differentially expressed genes it is natural to compute the maximum of suitable "averages" of gene expressions. This argument suggests using the sup-norm. In fact, simulation results presented in Kuelbs and Vidyashankar (2009) suggest the superiority of the sup-norm over other ℓ^ρ norms.

We comment on the relevance of the one-sample problem for microarray data. As explained by Smyth (2004), microarray experiments where wild-type and mutant labeled cDNA samples are competitively hybridized to a single array result in a one-sample location problem. In the context of identifying DEGs, we are then interested in testing the null hypothesis, $H_0 : \mu = \mathbf{0}$. This hypothesis corresponds to testing for the equality of expression levels between the mutant and wild type samples.

Clearly, for (1) to be useful in testing for DEGs we need to estimate $\mathbf{\Sigma}$ accurately. Several authors have discussed the difficulty in estimating $\mathbf{\Sigma}$ in large p, small n settings (Tsai and Chen, 2009). We use the shrinkage based estimator of Ledoit and Wolf (2004) as developed by Strimmer and his co-authors (Schafer and Strimmer, 2005; Opgen-Rhein and Strimmer, 2007). The choice of our estimator of covariance matrix is dictated by good finite sample properties as described in Ledoit and Wolf (2004). In particular, the estimator is guaranteed to be positive definite. The algorithm is implemented in both R language (corpcor) and Matlab language (covshrink), which are freely available at `http://strimmerlab.org/software.html`.

Using (1) and the shrinkage estimator for the covariance matrix, we now provide an algorithm for testing the null hypothesis that the mean of $\mathbf{X}_1$ is zero, that is that the genes are not differentially expressed. This is a Monte-Carlo algorithm used to approximate the distribution of $\|N_p\left(\mathbf{0}, \mathbf{\Sigma}\right)\|_\rho$. The user would first decide on a value of ρ and level of significance α to use. Again, for testing for DEGs, $\mu_0 = \mathbf{0}$. When $\rho = \infty$, we refer to this procedure as the sup-norm (SN) test.

1. Compute the observed test statistic, T_ρ.
2. Estimate the covariance matrix S, using shrinkage.
3. Generate B random vectors $\mathbf{Y}_1$,...,$\mathbf{Y}_n \sim N_p(\mathbf{0}, S)$; compute the norm of these vectors, $T_i^\star \equiv \|\mathbf{Y}_i\|_\rho$; finally compute the $(\alpha/2)$ sample quantile $\hat{q}_{\alpha/2}$ and the $(1-\alpha/2)$ sample quantile $\hat{q}_{1-\alpha/2}$ from $T_1^\star, ..., T_B^\star$.
4. Reject if $T_\rho < \hat{q}_{\alpha/2}$ or if $T_\rho > \hat{q}_{1-\alpha/2}$.

Missing data, where the expression values for some genes in some replicates are unavailable, is a common feature of microarray experiments. One of the advantages of our methodology is that it is easily extended to handle this situation. In this case, there are $n_j \leq n$ observations for each gene. And we define the averaged expression level for gene j as $\bar{X}_j = \frac{1}{n_j}\sum_{i=1}^{n_j} X_{i,j}$, $j = 1, ..., p$. Before, each component was scaled by $\sqrt{n}$, and now each component is scaled by $\sqrt{n_j}$. That is, we define the vector of gene-wise averaged, centered and scaled expression levels $\mathbf{X}_s = \left(X_{s,1}, ..., X_{s,p}\right)^t$, where $X_{s,j} = \sqrt{n_j}\left(\bar{X}_j - \mu_{0,j}\right)$, $j = 1, ..., p$. Now, to test the null hypothesis, $H_0 : \mu = \mu_0$, we now use the statistic $T_\rho \equiv \|\mathbf{X}_s\|_\rho$. We then approximate the null distribution as described above. As discussed in Kuelbs and Vidyashankar (2009), the procedure continues to work as long as the missing structure is modeled as missing completely at random (Little and Rubin, 2002).

2.2 Two-Sample Formulation

The two-sample problem is a straight forward generalization of the one-sample problem given above. In this case we have two independent samples $\{\mathbf{X}_{i1} : 1 \leq i \leq n_1\}$ and $\{\mathbf{X}_{i2} : 1 \leq i \leq n_2\}$; for fixed k, $\{\mathbf{X}_{i1} : 1 \leq i \leq n_k\}$ are i.i.d. random vectors with mean μ_k and covariance matrix $\mathbf{\Sigma}_k$. Here X_{ijk} represents the expression level of gene j from array i in sample k and $\bar{\mathbf{X}}_k = \left(\bar{X}_{1k}, ..., \bar{X}_{pk}\right)^t$, where $\bar{X}_{jk} = \frac{1}{n_k}\sum_{i=1}^{n_k} X_{ijk}$, $j = 1, ..., p$. To test the null hypothesis of equal sample means, $H_0 : \mu_1 = \mu_2$, which corresponds to testing for the equality of expression levels under two conditions, we consider statistics of the form, $T_{\rho,2} \equiv \|\bar{\mathbf{X}}_1 - \bar{\mathbf{X}}_2\|_\rho$. Again, using results from Kuelbs and Vidyashankar (2009), we have that $T_{\rho,2} \approx \|N_p\left(\mathbf{0}, \frac{1}{n_1}\mathbf{\Sigma}_1 + \frac{1}{n_2}\mathbf{\Sigma}_2\right)\|_\rho$.

To the test the null hypothesis we carry out the same basic steps as described in the subsection above. First we use the shrinkage algorithm described in Opgen-Rhein and Strimmer (2007) to estimate the covariance matrices and then approximate the null distribution of the test statistic using the Monte-Carlo algorithm given above.

2.3 Screening Algorithm

The strategy of our screening algorithm is to repeatedly apply the SN test, 'throwing out' genes that result in tests of accepting the null hypothesis and keeping genes that result in rejecting the null hypothesis. In the algorithm we use the notation I_i to denote the indicator for test i, indicating whether the test detected a DEG among all of the genes in group i. Specifically, $I_i = 1$ means the SN test rejected the null hypothesis for group i ($I_i = 0$ otherwise). There are certain parameters one needs to set to run the algorithm: the (expected) initial dimension size (d_0), the reduction factor (r), and the final cutoff (p_f).

We now describe an example which explains the role of these parameters. Assume the data consists of $p = 2000$ genes; set $d_0 = 100$, $r = 2$, and $p_f = 30$. The value $d_0 = 100$ means that in the first round of tests we will divide the genes into $2000/d_0 = 20$ groups with an expected group size of $d_0 = 100$; the value of $r = 2$ means in each subsequent stage the expected group size will be reduced by a factor of 2; finally, the value $p_f = 30$ means that the algorithm will run until the total number of remaining genes is less than or equal to 30. In the first stage, randomly subset the genes into 20

groups (with an average of 100 genes a group) and perform 20 SN tests on these groups. Keep all of the genes in groups with $I_i = 1$, and throw out the others. To start stage 2 take these remaining genes and divide them into groups with expected size $d_0/r = 50$; now repeat the process.

In practice one would set the parameters of the algorithm based on the characteristics of the observed data; for instance, the number of samples, the number of genes, and the variance of the data.

The structure of our algorithm is outlined below. The algorithm outputs a reduced set of genes that ideally will contain all of the differentially expressed genes. Notice in the update step, there is a check for the case $p_u = p_a$. This case comes about if in the current stage the SN test for each group rejects the null hypothesis, and no genes can be removed. If the true number of DEGs is greater than the chosen cutoff, i.e. $p_d > p_f$, then it is desirable for the algorithm to halt before the number of genes is reduced to below p_f. But it is also possible that this case arises because of the particular assignment of genes. For example, suppose that there are ten groups and 10 DEGs. If the assignment is such that one DEG is placed in each of the 10 groups, then all 10 tests may (correctly) reject the null hypothesis. To account for this situation, when $p_u = p_a$, we do a second allocation of the genes and test if any genes can be removed after this second allocation. In principle, a user could re-allocate any number of times before deciding the set of genes cannot be reduced further.

Screening Algorithm

Input. Set $p_a = p$, $d_a = d_0$, $count = 0$. Continue to Step 1.

Step 1. (Random Allocation) Randomly allocate the p_a genes to $K \equiv \lceil p_a/d_a \rceil$ groups. Continue to Step 2.

Step 2. (Test). Perform the SN procedure on each of the K groups. Continue to Step 3.

Step 3. (Update) Remove all genes in groups with $I_i = 0$. Let p_u denote the updated number of genes (after removal).
if $p_u = 0$

Stop. Output $\emptyset$ (declare that none of the genes are differentially expressed).

elseif $p_u \leq p_f$.

Stop. Output the set G_f, which consists of the labels for the remaining p_u genes.

elseif $p_u = p_a$.

if $count = 0$.

Update $count = 1$. Return to Step 1.

elseif $count = 1$.

Stop. Output the set G_f, which consists of the labels for the remaining p_u genes.

else

Update $d_u = d_a/r$. Set $p_a = p_u$, $d_a = d_u$, $count = 0$. Return to Step 1.

3 SIMULATION STUDY

We begin by outlining the specifications used in the simulation experiments. We then present results for the one-sample problem, followed by results for the two-sample problem.

3.1 Specifications for the Simulation Studies

All of the Monte-Carlo experiments presented below are based on 5000 simulated data sets. In all cases, the size of the test is fixed at $\alpha = .05$ and $B = 2000$ samples are used to approximate the null distribution. We generate data from multivariate normal distributions. For simulations concerning the one-sample problem , data is simulated from $N_p(\mu, \mathbf{\Sigma})$; for the two-sample problem, sample k is simulated from $N_p(\mu_k, \mathbf{\Sigma}_k)$, for $k = 1, 2$. We now describe our choices for the mean vector and covariance matrix.

In the one-sample problem the mean vector is chosen as follows. Let p_d be the number of DEGs in the data set, which we will assume all have a common mean $\mu_d \neq 0$. The mean vector contains p_d non-zero elements and $p - p_d$ zeros, Hence, the mean vector is $\mu = (\mu_d, ..., \mu_d, 0, ..., 0)^t$. Note that $p_d = 0$ corresponds to the case of no DEGs; with the corresponding mean vector $\mu = \mathbf{0}$. For the two-sample problem, only sample one contains DEGs, while sample two contains all null genes. Thus, $\mu_1 = (\mu_d, ..., \mu_d, 0, ..., 0)^t$, while $\mu_2 = \mathbf{0}$.

We use experimental data to set the covariance matrices for our simulations. Specifically, we use the leukemia dataset described by Golub *et al.* (1999), which studies the gene expression in two types of leukemia, acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML). We use the same pre-processing step as described in Dudoit *et al.* (2002a) Section 3.1; leaving 3571 genes from 72 patients, 38 ALL and 25 AML. On the remaining 3571 genes, we apply the standardization technique described in Section 3.3 of the same paper. We then separately estimate a covariance matrix from the ALL group and the AML group, denoted $\mathbf{\Sigma}_L$ and $\mathbf{\Sigma}_M$, respectively. Specifically, we randomly permuted the 3571 genes and then estimated the covariance matrix using the shrinkage algorithm (for correlations) of Schafer and Strimmer (2005). This method produces two 3571×3571 covariance matrices which are fixed throughout the paper. For a simulation study based on $p \leq 3571$ genes, we first fix the covariance matrix of appropriate dimensions by considering the $p \times p$ upper sub-matrix of $\mathbf{\Sigma}_J$ denoted $\mathbf{\Sigma}_{J,p}$, $J = L, M$. That is, $(\mathbf{\Sigma}_{J,p})_{l,m} = (\mathbf{\Sigma}_J)_{l,m}$, for $1 \leq l, m \leq p$. Finally, for each simulated data set, we simulate n random vectors from the $p-$dimensional normal distribution $N_p(\mu, \mathbf{\Sigma}_{I,p})$, where μ is a specified $p \times 1$ vector which represents the mean expression level of the genes.

Throughout, $\mathbf{\Sigma}$ is set to be a constant multiple of $\mathbf{\Sigma}_{L,p}$, $\mathbf{\Sigma} = v\mathbf{\Sigma}_{G,p}$. By taking different values of v , we will examine the role of variances on the screening algorithm. In particular, $v > 1$ gives a simple way for examining the effect of increased variance in the data. Similarly, for the two-sample simulations, $\mathbf{\Sigma}_1 = v_1\mathbf{\Sigma}_{L,p}$ and $\mathbf{\Sigma}_2 = v_2\mathbf{\Sigma}_{M,p}$.

3.2 One-Sample Simulation Results

We consider simulation experiments related to the one-sample problem. First we present results concerning the size and power of the SN test. We then present results for the screening algorithm.

First we evaluate the size of the SN test under different conditions. For size experiments, all of the genes are null, and thus we set $\mu = \mathbf{0}$. We consider data based on $n = 20$ replicates and examine the size as the number of genes increases from $p = 20$ to 100. The result of the simulations are displayed in Figure 1. We observe that in all cases the Type I error rates are close the nominal values of $\alpha = .05$.

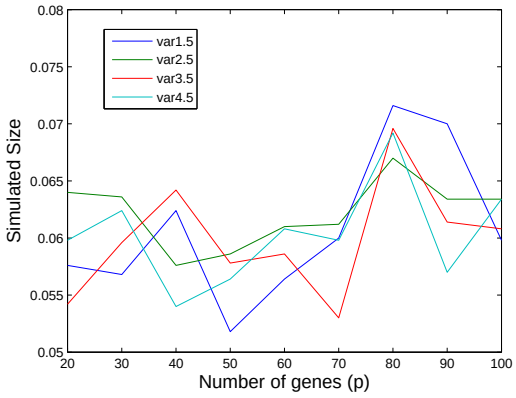
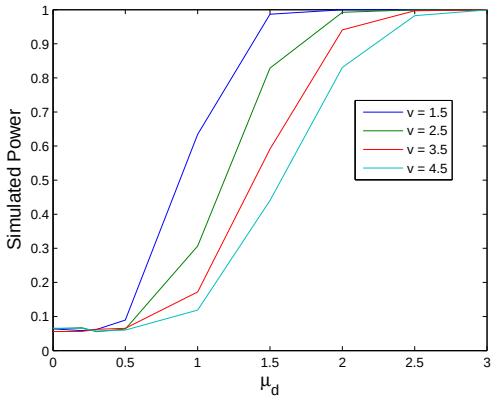
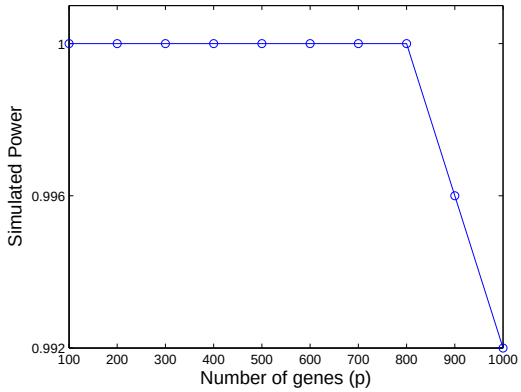


Fig. 1. Simulated size versus the number of genes (p), for different covariance matrices $v\mathbf{\Sigma}_{L,p}$. In all cases, there are $n = 20$ replicates.



(a) Simulated power versus the DE mean (μ_d), for different covariance matrices $v\mathbf{\Sigma}_{L,100}$.



(b) Simulated power versus the number of genes (p)

Fig. 2. Plots of simulated power. In all cases, there are $n = 20$ replicates and one DEG among the total set of p genes. In Figure 2(a), the number of genes is fixed at $p = 100$. In Figure 2(b), the covariance matrix is $\mathbf{\Sigma}_{L,p}$ and the DE mean is $\mu_d = 2$

Next we consider the power of the SN test; for power experiments, the set of genes includes at least one DEG. We consider experiments that examine the power under increasing variance and increasing number of genes. The result of the simulations are displayed in Figure 2. They show that the SN test is very powerful in detecting a single DEG. With $p = 800$ total genes (and only one DEG) the test correctly rejects the null hypothesis in all 5000 experiments; with $p = 1000$ genes the test rejects the null in 4960 of the experiments (see the results in Figure 2(b)).

We now present numerical results obtained by performing the screening algorithm on simulated data sets. We recall that the screening algorithm repeatedly applies the SN test, thereby reducing the original set of genes to a set G_f. For a single data set, we record two performance measures of the screening algorithm: the number of retained DEGs (R) and the total number of genes after the final run, $|G_f|$. For each experiment, we report the minimum, maximum, and mean of R and $|G_f|$ over the 5000 simulations. We will clarify these ideas with a concrete example. Assume that there are 2000 total genes, 10 DEGs, and that we set the cutoff at $p_f = 30$. Furthermore, assume that the algorithm continues to run until $|G_f| < p_f$. Ideally, after the algorithm has run, all 10 DEGs should remain in the final set G_f. We record the size of G_f and the number of DEGs which remain in G_f. We reiterate that the screening algorithm can end in three different ways: *exit one* occurs when the algorithm runs until the cutoff is reached, $1 \leq |G_f| \leq p_f$; *exit two* occurs when the algorithm cannot reduce the number of genes below p_f, $|G_f| > p_f$; and *exit three* occurs when the algorithm declares that all of the genes are null, $G_f = \emptyset$. We only report R and $|G_f|$ for those data sets which result in exit one or exit two. In our simulation experiments all of the data sets contain 10 DEGs; additionally, the parameters of the screening algorithm are fixed at $(r, d_0, p_f) = (2, 100, 30)$.

First we consider the impact of changing the mean for the DEGs. In this experiment there are $p = 2000$ genes, 1990 of the genes have mean zero while the remaining 10 DEGs have mean μ_d; we consider $\mu_d = .5, 1, 1.5$, and 2. With $\mu_d = .5$, 71.82% of the simulated data sets resulted in exit one, the remaining 28.18% resulted in exit three; for the other values of μ_d all 5000 simulated data sets resulted in exit one. The results of the simulations are displayed in Figure 3.

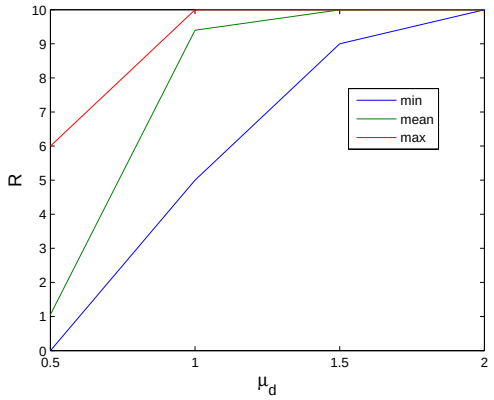
If $\mu_d = .5$, the algorithm does not perform well; on average it only retains one of the DEGs. However, with $\mu_d = 1.5$, the algorithm, on average, is retaining all 10 of the DEGs. With $\mu_d = 2$, in all 5000 simulations, the algorithm retains all 10 of the DEGs.

Next we consider the impact of the total number of genes present. In this case, $\mu_d = 2$ is fixed and we considered $p = 1000, 1500, 2000, 2500$, and 3000 genes. In this experiment, the algorithm ended in exit one and retained all 10 DEGs for every simulated data set. Evidently, with $\mu_d = 2$, the algorithm can handle very high dimensions.

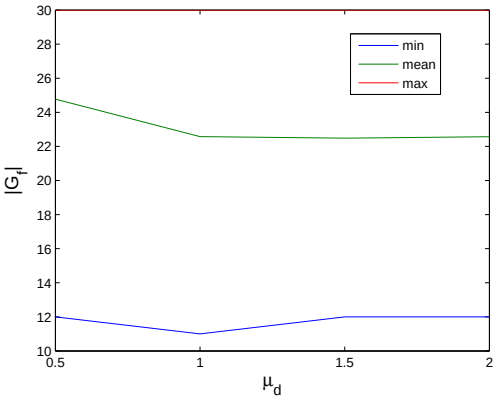
3.3 Two-Sample Simulation Results

In this section we consider simulation experiments related to the two-sample problem. First we present results which study the size and power of the SN test. We then present results for the screening algorithm.

All of the experiments presented in this section use $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_{L,p}$ and $\mathbf{\Sigma}_2 = \mathbf{\Sigma}_{M,p}$. First we evaluate the size of the SN test under

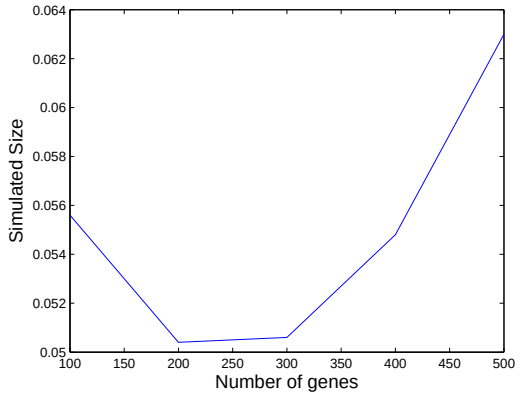


(a) Number of retained DEGs (R) versus the DE mean (μ_d). The minimum, maximum, and average (over the 5000 simulated data sets) are all given.

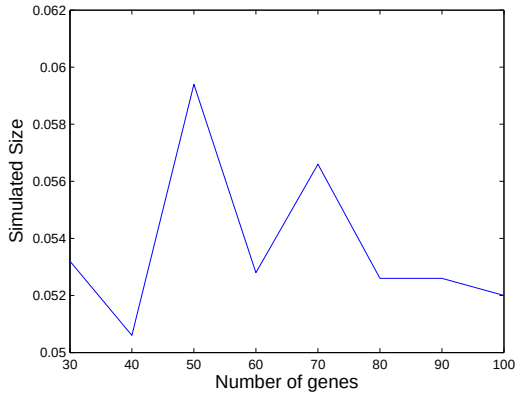


(b) Final number of genes ($|G_f|$) versus the DE mean (μ_d). The minimum, maximum, and average (over the 5000 simulated data sets) are all given.

Fig. 3. Performance of the screening algorithm as the DE mean (μ_d) changes. In all cases there are $n = 20$ replicates, 10 DEGs, $p = 2000$ total genes, and $\mathbf{\Sigma} = \mathbf{\Sigma}_{L,2000}$.



(a) $n_1 = 10, n_2 = 15$



(b) $n_1 = 22, n_2 = 25$

Fig. 4. Simulated size versus the number of genes (p). In both cases, $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_{L,p}$ and $\mathbf{\Sigma}_2 = \mathbf{\Sigma}_{M,p}$.

different conditions. In the first experiment, we consider data based on $n_1 = 10$ and $n_2 = 15$ replicates and examine the size as the number of genes increases from $p = 100$ to 500. In the second experiment, we have $n_1 = 22$ and $n_2 = 25$ replicates and examine the size as the number of genes increases from $p = 30$ to 100. The result of the simulations are displayed in Figure 4. In all cases the Type I error rate is close to the nominal value of .05.

Next we consider the power of the SN test; in these experiments the sample sizes are fixed at $n_1 = 10$ and $n_2 = 15$. We consider experiments that examine the impact of the total number genes, the mean of the DEGs, and the total number of DEGs. The result of the simulations are displayed in Figure 5. It is clear that the SN test is very powerful in detecting even a single DEG.

We now consider the performance of the screening algorithm in the two-sample setting. All of the experiments presented in this section use $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_{L,p}$ and $\mathbf{\Sigma}_2 = \mathbf{\Sigma}_{M,p}$, and $n_1 = 10$, $n_2 = 15$.

First we consider the impact of changing the mean for the DEGs. In this experiment there are $p = 2000$ genes, 1990 of the genes have mean zero while the remaining 10 DEGs have mean μ_d; we consider $\mu_d = .5, 1, 1.5$, and 2. With $\mu_d = .5$, 65.42% of the simulated data sets resulted in exit one, the remaining 34.58% resulted in exit three; for the other values of μ_d all 5000 simulated data sets resulted in exit one. The results of the simulations are displayed in Figure 6. The results of this experiment are almost identical to the one-sample analog. Specifically, with $\mu_d = .5$ the algorithm does not perform well; however, with $\mu_d = 2$, in all 5000 simulations, the algorithm retains all 10 of the DEGs.

Next we consider the impact of the total number of genes present. In this case, $\mu_d = 2$ is fixed and we considered $p = 1000, 1500, 2000, 2500$, and 3000 genes. Just as in the one-sample analog, in this experiment, the algorithm ended in exit one and retained all 10 DEGs in every simulated data set.

4 DATA ANALYSIS

In this section we analyze data from a study of the apolipoprotein AI (ApoAI) gene described in Callow *et al.* (2000). This data has been previously analyzed by Smyth (2004); a tutorial for analyzing the data set is available online as part of the LIMMA user's manual (Smyth *et al.*, 2003). We normalize the data using the LIMMA package as described in Smyth *et al.* (2003).

The ApoAI gene plays a central role in high density lipoprotein (HDL) metabolism; see Williamson *et al.* (1992) and Plump *et al.* (1996) for more detailed discussions of the ApoAI knockout model. The Callow *et al.* (2000) experiment was designed to study the effect of ApoAI deficiency on other genes in the liver. To this end, data was collected on 8 ApoAI knockout mice and 8 control mice. For each of these 16 mice, mRNA measurements were collected from liver tissue. The RNA from each mouse was hybridized to a separate array. The data set consists of 16 arrays with measurements on 5548 expressed sequence tags (ESTs).

Callow *et al.* (2000) identified eight ESTs (representing four different genes) which are differentially expressed in the knockout group versus the control group. Smyth (2004) lists the top fifteen differentially expressed ESTs based on his LIMMA approach. Of these fifteen, the top eight coincide with the ones identified in Callow *et al.* (2000). In fact, Smyth writes "the top eight genes stand out clearly from the other genes and all methods clearly separate these genes from the others" (note that Smyth uses "gene" instead of "EST"). Our screening algorithm identified eleven ESTs, which included the eight ESTs identified in Callow *et al.* (2000).

5 DISCUSSION

Multivariate statistical procedures are useful for identifying DEGs because they account for gene interactions. Incorporating gene interactions enables multivariate tests to identify differentially expressed genes, which marginally are not detectable using univariate tests. In statistical terms, multivariate tests have improved power compared to univariate tests. It is important to make the distinction between *classical multivariate analysis*, where $n > p$, and *modern multivariate analysis*, where $n < p$. In classical multivariate analysis, Hotelling's T^2 statistic is used for two sample comparisons. For the two sample problem, Hotelling's T^2 statistic is defined as

$$T^2 \equiv \frac{n_1 n_2}{n_1 + n_2} \left(\bar{\mathbf{X}}_1 - \bar{\mathbf{X}}_2\right)^t \mathbf{S}^{-1} \left(\bar{\mathbf{X}}_1 - \bar{\mathbf{X}}_2\right),$$

where $\mathbf{S}$ is the the pooled covariance estimator (Anderson, 2003). The test is performed by comparing $\frac{n_1+n_2-p-1}{(n_1+n_2-2)p} T^2$ to the F distribution with numerator p degrees of freedom for the numerator and $n_1 + n_2 - p - 1$ for the denominator. From the denominator degrees of freedom, we see that this procedure breaks down when $p > n_1 + n_2$. Several authors have adopted modifications to handle this issue. For instance, Lu *et al.* (2005) combines Hotelling's T^2 with a forward search algorithm, which restricts the total number of identified genes to be less than $n_1 + n_2 - 2$. We suggest to instead use a genuinely modern multivariate statistical test which handles the case $p > n_1 + n_2$. Namely, we use the sup-norm test recently proposed by Kuelbs and Vidyashankar (2009), which was specifically developed for the large p, small n setting and is asymptotically justified in this context.

In this paper we presented a novel multivariate approach to identifying DEGs. We propose an algorithm which repeatedly applies the sup-norm test to screen for DEGs. Our simulation studies establish the validity of the sup-norm statistic in terms of Type I error and power. With simulated data sets, the screening algorithm retains the majority of DEGs under a variety of experimental conditions. We also used our methodology to analyze the ApoAI knockout experiment (Callow *et al.*, 2000). Our algorithm identified the biologically significant genes, as discussed by other authors (Callow *et al.*, 2000; Smyth, 2004). Additionally, our methodology is easily adapted to handle missing data.

ACKNOWLEDGEMENT

Funding: ANV's research was supported in part by a grant from NSF DMS 000-03-07057 and also by grants from the NDCHealth Corporation.

REFERENCES

Anderson, T. W. (2003). *An introduction to multivariate statistical analysis*. Wiley Series in Probability and Statistics. Wiley-Interscience [John Wiley & Sons], Hoboken, NJ, third edition.

Callow, M. J., Dudoit, S., Gong, E. L., Speed, T. P., and Rubin, E. M. (2000). Microarray expression profiling identifies genes with altered expression in HDL deficient mice. *Genome Research*, **10**, 2022 – 2029.

Dudoit, S. and van der Laan, M. J. (2008). *Multiple testing procedures with applications to genomics*. Springer Series in Statistics. Springer, New York.

Dudoit, S., Fridlyand, J., and Speed, T. P. (2002a). Comparison of discrimination methods for the classification of tumors using gene expression data. *J. Amer. Statist. Assoc.*, **97**(457), 77–87.

Dudoit, S., Yang, Y. H., Callow, M. J., and Speed, T. P. (2002b). Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. *Statist. Sinica*, **12**(1), 111–139. Special issue on bioinformatics.

Friedman, A. (1982). *Foundations of modern analysis*. Dover Publications Inc., New York. Reprint of the 1970 original.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., and Bloomfield, C. D. (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, **286**, 531–537.

Kim, B. S., Kim, I., Lee, S., Kim, S., Rha, S. Y., and Chung, H. C. (2005). Statistical methods of translating microarray data into clinically relevant diagnostic information in colorectal cancer. *Bioinformatics*, **21**(4), 517–528.

Kuelbs, J. and Vidyashankar, A. N. (2009). Asymptotic inference for high dimensional data. *Annals of Statistics*. To Appear. Preprint available: http://www.stat.cornell.edu/~vidyashankar/.

Ledoit, O. and Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. *J. Multivariate Anal.*, **88**(2), 365–411.

Leung, Y. F. and Cavalieri, D. (2003). Fundamentals of cdna microarray data analysis. *Trends in Genetics*, **19**(11), 649 – 659.

Little, R. J. A. and Rubin, D. B. (2002). *Statistical analysis with missing data*. Wiley Series in Probability and Statistics. Wiley-Interscience [John Wiley & Sons], Hoboken, NJ, second edition.

Lu, Y., Liu, P.-Y., Xiao, P., and Deng, H.-W. (2005). Hotelling's T2 multivariate profiling for detecting differential expression in microarrays. *Bioinformatics*, **21**(14), 3105–3113.

Opgen-Rhein, R. and Strimmer, K. (2007). Accurate ranking of differentially expressed genes by a distribution-free shrinkage approach. *Statistical Applications in Genetics and Molecular Biology*, **6**(1), Article 9.

Plump, A., Erickson, S., Weng, W., Partin, J., Breslow, J., and Williams, D. (1996). Apolipoprotein A-I is required for cholesteryl ester accumulation in steroidogenic cells and for normal adrenal steroid production. *J. Clin. Invest.*, **97**, 2660 – 2671.

Schafer, J. and Strimmer, K. (2005). A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical Applications in Genetics and Molecular Biology*, **4**(1), Article 32.

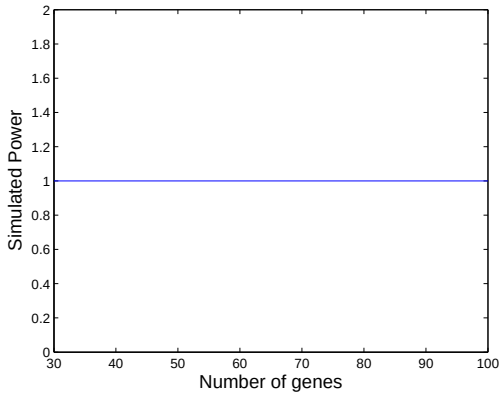
Smyth, G. (2004). Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, **3**(1). Article 3.

Smyth, G. K., Thorne, N., and Wettenhall, J. (2003). Limma: Linear Models for Microarray, User's Guide. Software manual available from http://bioinf.wehi.edu.au/limma.

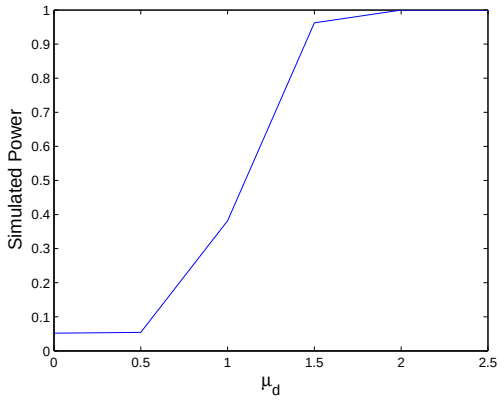
Szabo, A., Boucher, K., Jones, D., Tsodikov, A. D., Klebanov, L. B., and Yakovlev, A. Y. (2003). Multivariate exploratory tools for microarray data analysis. *Biostat*, **4**(4), 555–567.

Tsai, C.-A. and Chen, J. J. (2009). Multivariate analysis of variance test for gene set analysis. *Bioinformatics*, **25**(7), 897–903.

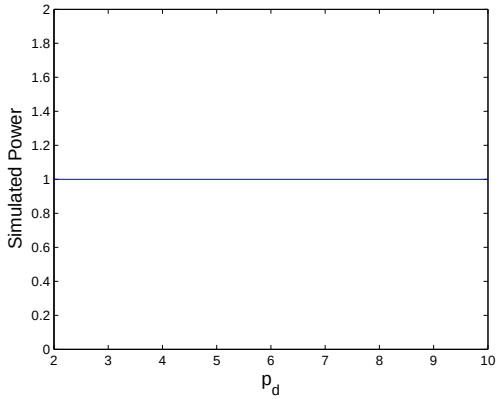
Williamson, R., Lee, D., Hagaman, J., and Maeda, N. (1992). Marked reduction of high density lipoprotein cholesterol in mice genetically modified to lack apolipoprotein A-I. *Proceedings of the National Academy of Sciences of the United States of America*, **89**(15), 7134–7138.



(a) Simulated power versus the number of genes (p). Here $p_d = 1$ with $\mu_d = 2$.

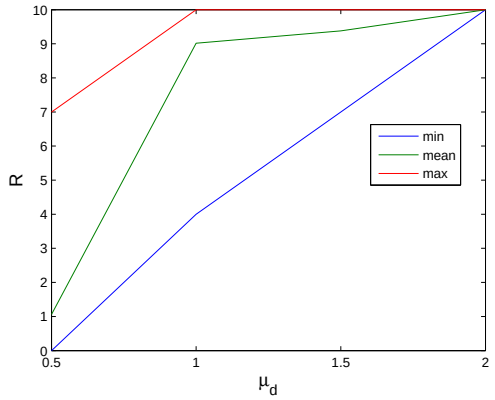


(b) Simulated power versus the DE mean (μ_d). Here $p = 100$ and $p_d = 1$.

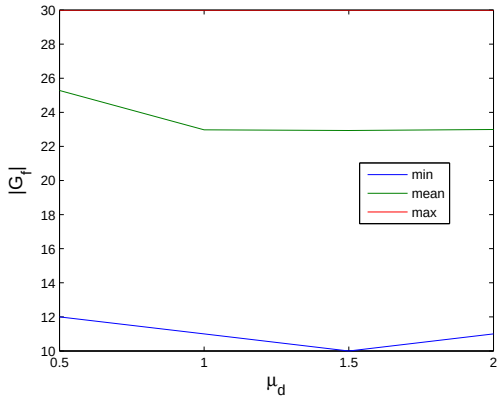


(c) Simulated power versus the number of DEGs (p_d). Here $p = 100$ and $\mu_d = 2$.

Fig. 5. Plots of simulated power. In all cases, $n_1 = 10$, $n_2 = 15$, and $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_{L,p}$ and $\mathbf{\Sigma}_2 = \mathbf{\Sigma}_{M,p}$.



(a) Number of retained DEGs (R) versus the DE mean (μ_d). The minimum, maximum, and average (over the 5000 simulated data sets) are all given.



(b) Final number of genes ($|G_f|$) versus the DE mean (μ_d). The minimum, maximum, and average (over the 5000 simulated data sets) are all given.

Fig. 6. Performance of the screening algorithm as the DE mean (μ_d) changes. In these simulations there are 10 DEGs and $p = 2000$ total genes. Additionally, In all cases there are $n_1 = 10$, $n_2 = 15$, and $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_{L,2000}$ and $\mathbf{\Sigma}_2 = \mathbf{\Sigma}_{M,2000}$