

Bagging multiple comparisons from microarray data

Dimitris N. Politis*

Abstract

The problem of large-scale simultaneous hypothesis testing is revisited. Bagging and subbagging procedures are put forth with the purpose of improving the discovery power of the tests. The procedures are implemented in both simulated and real data. It is shown that bagging and subbagging significantly improve power at the cost of a small increase in false discovery rate with the proposed ‘maximum contrast’ subbagging having an edge over bagging, i.e., yielding similar power but significantly smaller false discovery rates.

1 Introduction

The problem of simultaneous statistical inference is not new; see Miller (1981) for an early treatment. In the last decade, however, the statistical community has been faced with huge amounts of data and a subsequent need to address *large-scale* simultaneous hypothesis testing problems.

The prototypical such dataset involves gene expression data but different applications, such as functional Magnetic Resonance Imaging, flight spectroscopy, flow cytometry, etc., all give rise to similar problems from a statistician’s perspective. The microarray set-up is described below in the context of the gene expression example with the understanding that the same ideas are applicable to a host of other two-sample, multiple comparison problems.

*Department of Mathematics, University of California at San Diego, La Jolla, CA 92093-0112; www.math.ucsd.edu/~politis.

A typical experiment may entail data on n_X normal subjects, and n_Y patients. An array of N measurements is obtained from each subject. Therefore, the data can be organized as a $N \times n_X$ data matrix X (control group), and a $N \times n_Y$ data matrix Y (patient group); the (i, j) entry of X is denoted X_{ij} , and that of Y is denoted Y_{ij} . Column i from X has the data from the i th normal subject, and column j from Y has the data from the j th patient.

The X data are assumed independent of the Y data. A general model for this set-up is to assume that, for each k ,

$$X_{k,1}, X_{k,2}, \dots, X_{k,n_X} \sim \text{i.i.d. } F_X^{(k)} \text{ and } Y_{k,1}, Y_{k,2}, \dots, Y_{k,n_Y} \sim \text{i.i.d. } F_Y^{(k)} \quad (1)$$

where $F_X^{(k)}, F_Y^{(k)}$ are some distribution functions. For each $k = 1, \dots, N$, the issue is to test $H_0 : F_X^{(k)} = F_Y^{(k)}$ vs. not; this is the set-up of multiple comparisons.

More often than not, the testing focuses on a potential difference in the means of the X and Y data. In that case, practitioners typically assume

$$X_{k,1}, X_{k,2}, \dots, X_{k,n_X} \sim \text{i.i.d. } N(\mu_k, \sigma_k^2) \quad (2)$$

and

$$Y_{k,1}, Y_{k,2}, \dots, Y_{k,n_Y} \sim \text{i.i.d. } N(\nu_k, \sigma_k^2). \quad (3)$$

The multiple comparisons now boil down to testing $H_0 : \mu_k = \nu_k$ vs. not, for $k = 1, \dots, N$. From the k th row, the familiar t -statistic $t^{(k)} = (\bar{Y}_k - \bar{X}_k) / (\hat{\sigma} \sqrt{n_Y^{-1} + n_X^{-1}})$ can be calculated where $\bar{Y}_k = n_Y^{-1} \sum_{j=1}^{n_Y} Y_{kj}$, $\bar{X}_k = n_X^{-1} \sum_{i=1}^{n_X} X_{ki}$, and $\hat{\sigma}^2 = (n_X + n_Y - 2)^{-1} \{ \sum_{i=1}^{n_X} (X_{ki} - \bar{X}_k)^2 + \sum_{j=1}^{n_Y} (Y_{kj} - \bar{Y}_k)^2 \}$ is the pooled variance.¹ A typical testing procedure then rejects H_0 from the k th row when $t^{(k)}$ is too large in absolute value.

Suppose that exactly n_0 rows (genes) conform to H_0 , i.e., they are “null”, and so $N - n_0$ rows (genes) do not, i.e., they are “non-null”. Collect the indices of the truly non-null rows in a list denoted by TRUELIST; similarly, collect the row indices corresponding to the rejected t -statistics in the LIST

¹The normality assumption is not crucial in practice, especially if the sample sizes n_X and n_Y are relatively large. The assumption of common variance on the k th row of X and Y is more important but can be addressed if required leading to a slightly different form of the t -statistic; in any case, the flavor of the testing problem remains unchanged.

of genes *declared* to be non-null. Then we can define the multiple comparisons *achieved discovery power* as

$$ADP = \frac{\#\{LIST \cap TRUelist\}}{\#\{TRUelist\}}$$

and the *achieved false discovery rate* as

$$AFDR = \frac{\#\{LIST \cap \overline{TRUelist}\}}{\#\{LIST\}}$$

where $\#\{A\}$ denotes number of elements in set A , and \bar{A} is the complement of A . The breakthrough method of Benjamini and Hochberg (1995) was designed to control the *expected value* of the AFDR; this expected value is usually called simply the false discovery rate (FDR).

2 Motivation

Suppose that two different groups perform the same scientific experiment and come up with two different lists of genes declared non-null, say $LIST_1$ and $LIST_2$. Let $AFDR_1$ and $AFDR_2$ denote the false discovery rates in the two experiments; recall that (the expected values of) $AFDR_1$ and $AFDR_2$ are controlled, i.e., bounded, in a typical multiple comparisons experiment.

How can the two lists, $LIST_1$ and $LIST_2$, be combined for better inference? The natural answer is to ‘heed’ the evidence from both experiments and declare as non-null all elements in the $BIGLIST = LIST_1 \cup LIST_2$. Since the BIGLIST is bigger than either $LIST_1$ or $LIST_2$, the combined experiment will have more power; but what is the AFDR associated with the BIGLIST?

To proceed with the analysis, let us make the simplifying assumption that genes declared non-null in both studies are very likely truly non-null, i.e., that $SMALLLIST \subset TRUelist$ with high probability where we denote $SMALLLIST = LIST_1 \cap LIST_2$. Also let $FALSE_1$ denote the subset of $LIST_1$ that consists of false discoveries, i.e., genes falsely declared non-null; similarly for $FALSE_2$. Therefore, we have

$$AFDR_1 = \frac{\#\{FALSE_1\}}{\#\{LIST_1\}} \quad \text{and} \quad AFDR_2 = \frac{\#\{FALSE_2\}}{\#\{LIST_2\}} \quad (4)$$

from which the numbers $\#\{FALSE_1\}$ and $\#\{FALSE_2\}$ can be calculated as functions of $AFDR_1$ and $AFDR_2$.

Consequently, the AFDR associated with BIGLIST is given by:

$$\begin{aligned} AFDR_{BIG} &= \frac{\#\{FALSE_1\} + \#\{FALSE_2\}}{\#\{LIST_1\} + \#\{LIST_2\} - \#\{SMALLLIST\}} \\ &= \frac{AFDR_1 \times \#\{LIST_1\} + AFDR_2 \times \#\{LIST_2\}}{\#\{LIST_1\} + \#\{LIST_2\} - \#\{SMALLLIST\}}. \end{aligned} \quad (5)$$

Taking expectations in the above, we see that eq. (5) is satisfied with the expected false discovery rates (FDR) in place of the AFDRs, i.e., that:

$$FDR_{BIG} = \frac{FDR_1 \times \#\{LIST_1\} + FDR_2 \times \#\{LIST_2\}}{\#\{LIST_1\} + \#\{LIST_2\} - \#\{SMALLLIST\}}. \quad (6)$$

In experiments with low power it is not uncommon to have $LIST_1$ and $LIST_2$ be totally disjoint; see Efron (2006) for a discussion. Suppose we are in such a low-power set-up, and also suppose—for the sake of argument—that the two experiments have similar design, i.e., that $FDR_1 = FDR_2$. Then, the above equations show that $FDR_{BIG} = FDR_1 = FDR_2$. So, in this case, the combined experiment has more power with the *same* FDR, i.e., a win-win situation.

In general, however, $LIST_1$ and $LIST_2$ might not be disjoint, and the increase in power associated with BIGLIST will come at the price of an increase in FDR. However, it is the thesis of this paper that the increase in power may be well worth a small increase in FDR.

Before proceeding further, let us momentarily consider the generalization to the case of having M different groups perform the same experiment and coming up with their respective non-null lists, say $LIST_1, LIST_2, \dots, LIST_M$; let $AFDR_1, AFDR_2, \dots, AFDR_M$ denote the respective AFDRs. Under the same simplifying assumption, namely that genes declared non-null in at least two studies are very likely truly non-null, a similar calculation as before yields:

$$FDR_{BIG} = \frac{\sum_{i=1}^M FDR_i \times \#\{LIST_i\}}{\#\{BIGLIST\}} \quad (7)$$

where again FDR_{BIG} is the expected false discovery rate associated with $BIGLIST = \cup_{i=1}^M LIST_i$. Finally, note that the number of elements in

BIGLIST can be calculated as:

$$\begin{aligned} \#\{BIGLIST\} &= \sum_i \#\{LIST_i\} - \sum_{i \neq j} \#\{LIST_i \cap LIST_j\} \\ &+ \sum_{i \neq j \neq k \neq i} \#\{LIST_i \cap LIST_j \cap LIST_k\} + \dots + (-1)^{M-1} \times \#\{\cap_{i=1}^M LIST_i\}. \end{aligned}$$

3 Bootstrap and bagging

In Section 2, having multiple experiments (with their associated rejection *LISTS*) was discussed. In practice, however, the statistician is faced with a single dataset. Nonetheless, *resampling* and *subsampling* methods can be utilised in order to create additional (pseudo)samples.

Efron's (1979) *bootstrap* is one of the most prominent resampling methods. For i.i.d. data Z_1, \dots, Z_n , the bootstrap amounts to sampling randomly with replacement from the set $\{Z_1, \dots, Z_n\}$ to create the (pseudo)sample Z_1^*, \dots, Z_n^* ; see Efron and Tibshirani (1993) for a review. The bootstrap is closely related to Tukey's (1958) 'delete-1' *jackknife* which was generalized to a 'delete- d ' jackknife by Shao and Wu (1989). For i.i.d. data Z_1, \dots, Z_n , the delete- d jackknife is equivalent to *subsampling* with sample size $b = n - d$, i.e., sampling randomly *without* replacement from the set $\{Z_1, \dots, Z_n\}$ to create the (pseudo)sample Z_1^*, \dots, Z_b^* ; see Politis, Romano and Wolf (1999).

'*Bagging*', i.e., **bootstrap aggregation**, was put forth by Breiman (1996) in order to improve the accuracy of statistical predictors. The idea is to evaluate the predictor in question on a number of bootstrap (pseudo)datasets, and to combine the resulting predictors in an aggregate predictor. It has been shown that bagging indeed helps improve predictor accuracy in particular when the predictor is relatively unstable, i.e., when small changes in the data result in greatly perturbed predictions; see Bühlmann and Yu (2002). Bagging can alternatively be implemented in conjunction with subsampling in which case the term '*subbagging*' was suggested by Bühlmann and Yu (2002); see also Bühlmann (2003).

4 Balanced bagging and subbagging for microarrays

As discussed in Section 2, it is possible to have two different low-power experiments produce disjoint or almost disjoint rejection lists; this is evidence of *instability*. Thus, bagging and/or subbagging may be helpful for multiple comparisons as they have been shown to be helpful in prediction and classification.

We now elaborate on how to perform bagging and subbagging in the multiple comparisons, microarray set-up of Section 1; the main idea is to re/sub-sample subjects, i.e., columns of the matrices X and Y . Throughout this section it is assumed that the practitioner is using a fixed multiple hypothesis testing procedure, e.g., the procedure of Benjamini and Hochberg (1995) or Efron (2005), for *any* dataset that he/she may encounter.

The bagging and subbagging algorithms described below are termed ‘balanced’; the reason for this term will become more apparent in Section 6. Let $\underline{x}_1, \dots, \underline{x}_{n_X}$ and $\underline{y}_1, \dots, \underline{y}_{n_Y}$ denote the columns of X and Y respectively; B is an integer denoting the number of (pseudo)samples generated.

- **Balanced bagging.** For $k = 1, \dots, B$, construct the k th bootstrap (pseudo)sample $X^{(k)}$ and $Y^{(k)}$; the columns of $X^{(k)}$ and $Y^{(k)}$ respectively are given as $\underline{x}_{I_1}, \dots, \underline{x}_{I_{n_X}}$ and $\underline{y}_{J_1}, \dots, \underline{y}_{J_{n_Y}}$ where I_1, \dots, I_{n_X} are numbers drawn randomly with replacement from the index set $\{1, \dots, n_X\}$ and J_1, \dots, J_{n_Y} are numbers drawn randomly with replacement from the index set $\{1, \dots, n_Y\}$ and independently of I_1, \dots, I_{n_X} . From this k th (pseudo)sample, the rejection list $LIST_k$ is created.

To define subbagging, subsample sizes b_X and b_Y must be specified. Note that there is no reason here to have the subsample sizes be of smaller order of magnitude as compared to the original sample sizes; this is only required for estimation consistency which is not the objective here—see e.g. Politis et al. (1999). So, the subsample sizes for subbagging could (and should) be taken relatively large; furthermore, it is intuitive that a choice satisfying $b_X/b_Y \simeq n_X/n_Y$ might be fruitful as being more representative of the original dataset. Thus, a good rule-of-thumb may be to let $b_X \simeq a n_X$ and $b_Y \simeq a n_Y$ where the constant a is close to (but less than) one.

- **Balanced subbagging—random version.** For $k = 1, \dots, B$, construct the k th subbagging (pseudo)sample $X^{(k)}$ and $Y^{(k)}$; the columns of $X^{(k)}$ and $Y^{(k)}$ respectively are given as $\underline{x}_{I_1}, \dots, \underline{x}_{I_{b_X}}$ and $\underline{y}_{J_1}, \dots, \underline{y}_{J_{b_Y}}$ where I_1, \dots, I_{b_X} are numbers drawn randomly *without* replacement from the index set $\{1, \dots, n_X\}$ and J_1, \dots, J_{b_Y} are numbers drawn randomly *without* replacement from the index set $\{1, \dots, n_Y\}$ and independently of I_1, \dots, I_{b_X} . As before, from this k th (pseudo)sample, the rejection list $LIST_k$ is created.
- **Balanced subbagging—nonrandom version.** Let \mathcal{S}_X denote the set of all size b_X subsets of the index set $\{1, \dots, n_X\}$, and \mathcal{S}_Y denote the set of all size b_Y subsets of the index set $\{1, \dots, n_Y\}$ where b_X and b_Y are as above. A subbagging (pseudo)sample is given by $X^{(k_1)}$ and $Y^{(k_2)}$ where the columns of $X^{(k_1)}$ are the columns of X with indices given by the k_1 th element of set \mathcal{S}_X , and the columns of $Y^{(k_2)}$ are the columns of Y with indices given by the k_2 th element of set \mathcal{S}_Y . Since the set \mathcal{S}_X contains $\binom{n_X}{b_X}$ elements and the set \mathcal{S}_Y contains $\binom{n_Y}{b_Y}$ elements, it is apparent that there are $B = \binom{n_X}{b_X} \cdot \binom{n_Y}{b_Y}$ possible (pseudo)samples.

Of course, $\binom{n_X}{b_X} \cdot \binom{n_Y}{b_Y}$ can be a prohibitively large number, so considering *all* possible (pseudo)samples seems out of the question. The aforementioned random subbagging procedure side-steps this difficulty but so does the following scheme that has the additional benefit of nonrandom selection of ‘maximum contrast’ subsamples, i.e., subsamples that are ‘most’ different from one another in their composition.

It is easier to describe this idea in the ‘delete- d ’ framework (with $d = n - b$) as opposed to ‘choose- b ’; of course, now the game is delete- d *columns* from one of our data matrices.

- **‘Maximum contrast’ nonrandom subbagging.** Let m_X, m_Y be two positive integers, and divide the index set $\{1, \dots, n_X\}$ into the m'_X subsets $S_X^{(1)}, \dots, S_X^{(m'_X)}$ where $S_X^{(1)} = \{1, \dots, d_X\}, S_X^{(2)} = \{d_X + 1, \dots, 2d_X\}, \dots$, etc. where $d_X = \lceil n_X/m_X \rceil$ and $m'_X = \lceil n_X/d_X \rceil$; here $\lceil a \rceil$ is the smallest integer that is bigger or equal to a . The last set, i.e., $S_X^{(m'_X)}$, may have size less than d_X if m_X does not divide n_X but that poses no problem. Similarly, divide the index set $\{1, \dots, n_Y\}$ into the m'_Y subsets $S_Y^{(1)}, \dots, S_Y^{(m'_Y)}$ where $S_Y^{(1)} = \{1, \dots, d_Y\}, S_Y^{(2)} =$

$\{d_Y + 1, \dots, 2d_Y\}, \dots$, etc.

A subbagging (pseudo)sample is now given by $X^{(k_1)}$ and $Y^{(k_2)}$ where the columns of $X^{(k_1)}$ are the columns of X with indices given by the set $\{1, \dots, n_X\} - S_X^{(k_1)}$, and the columns of $Y^{(k_2)}$ are the columns of Y with indices given by the set $\{1, \dots, n_Y\} - S_Y^{(k_2)}$. Since the possible values of k_1 are $\{1, \dots, m'_X\}$, and those for k_2 are $\{1, \dots, m'_Y\}$, it is apparent that there are $m'_X \cdot m'_Y$ possible such (pseudo)samples; thus rejection lists $LIST_1, \dots, LIST_B$ can be created with $B = m'_X \cdot m'_Y$.

5 Combining the rejection lists

Let $LIST$ denote the rejection list of the original dataset X and Y , and $LIST_1, \dots, LIST_B$ the rejection lists corresponding to B (pseudo)samples from one of the algorithms of Section 4.

As in Section 2, the simplest suggestion is to combine the lists by a union, i.e., to define the aggregate/combined list as:

$$LIST.AGG = LIST \cup LIST_1 \cup LIST_2 \cup \dots \cup LIST_B. \quad (8)$$

However, other alternatives exist; their description is facilitated by the notion of ‘voting’ where a list is said to ‘vote’ that the i th gene is non-null when the i th gene is an element of the list.

Let $V(i)$ denote the number of votes the i th gene received from the ‘voting’ lists $LIST, LIST_1, \dots, LIST_B$. With this terminology, rejecting every gene in $LIST.AGG$ corresponds to the formula:

(i) declare the i th gene as non-null if $V(i) \geq 1$, i.e., it got at least one vote.

A more conservative approach might require to ‘second’ a vote, i.e., it would

(ii) declare the i th gene as non-null if $V(i) \geq 2$, i.e., it got at least two votes.

One might even raise the rejection threshold at a level higher than two although we will not consider that here. However, it is informative to see which genes received more votes than others in the sense that getting more votes corresponds to more evidence for being truly non-null. Thus, a plot of $V(i)$ vs. i may be a helpful diagnostic tool.

As a further diagnostic, we may define $N(h)$ as the number of genes that received at least h votes, i.e., $N(h)$ is the size of the non-null list obtained

from a criterion of the type: reject gene i if $V(i) \geq h$. A plot of $N(h)$ vs. h is another way to quantify the ‘strength of evidence’ towards proclaiming each gene on *LIST.AGG* as non-null.

Note that formula (ii) treats *LIST* as ‘equal’ to $LIST_1, \dots, LIST_B$, and carries the implicit risk that not all of the genes found in *LIST* will be finally rejected. To remedy this, we may give the original *LIST* more weight in the aggregation. The easiest way of doing this is giving the original *LIST* a double vote, i.e., defining $V^*(i)$ to equal the number of votes the i th gene got from $LIST_1, \dots, LIST_B$ plus a double vote from the original *LIST* (if indeed *LIST* gave it a vote),² and then

(*ii**) declaring the i th gene as non-null if $V^*(i) \geq 2$.

As above, we can define $N^*(h)$ as the number of genes that received at least h votes from formula (*ii**) above, i.e., $N^*(h)$ is the size of the non-null list obtained from a criterion of the type: reject gene i if $V^*(i) \geq h$. A plot of $N^*(h)$ vs. h has an interpretation similar to that of plot of $N(h)$ vs. h .

6 Comparison to bagging for classification

Microarray data, such as the ones arising in gene expression data, lend themselves to analysis with the objective of classifying future observations; in other words, using the data to decide if a future observation belongs to the control or the patient group—the decision being based on the new observation’s ‘features’ (i.e. gene expressions) only. Since Breiman’s (1996) original bagging was aimed at improving predictors and classifiers, it is of no surprise that there is already a body of literature on bagging and subbagging microarrays with the purpose of classification; a partial list includes Dettling (2004), Dudoit and Fridlyand (2003), and Dudoit, Fridlyand and Speed (2002).

Although related at the outset, classification is a very different problem than hypothesis testing; their objectives are quite different, and so are the methods involved. To illustrate this point, we now give a brief description of the bagging/subbagging procedures as used for microarray classification.

To start with, concatenate the X and Y matrices into a big $N \times n$ matrix denoted by W where $n = n_X + n_Y$. Let $\underline{w}_1, \dots, \underline{w}_n$ denote the columns of

²Note that we can also get $V^*(i)$ by computing $V(i)$ counting single votes from *LIST*, *LIST*, and $LIST_1, \dots, LIST_B$, i.e., having *LIST* double-up and—in effect—vote twice.

W , and define new variables U_1, \dots, U_n such that $U_i = 0$ for $i \leq n_X$, and $U_i = 1$ for $i > n_X$; in this sense, the variable U_i is an indicator of which group (normal or patient) the i th subject belongs to. Finally, define $Z_i = (\underline{w}_i, U_i)$ for $i = 1, \dots, n$.

The Z_i data are multivariate but they constitute a *single* sample. This sample can be bootstrapped—by sampling with replacement from the set $\{Z_1, \dots, Z_n\}$, or subsampled—by sampling without replacement from the same set $\{Z_1, \dots, Z_n\}$, in order to create (pseudo)samples. In all the above-referenced works, bagging/subbagging for microarray classification follows the above paradigm.

Note, however, that the above single-sample bootstrap scheme can generate (pseudo)samples that are *unbalanced* in terms of the two groups (normal/patient). To elaborate, let $Z_i^* = (\underline{w}_i^*, U_i^*)$ for $i = 1, \dots, n$ be the bootstrap (pseudo)sample. Then, it is not unlikely that $\sum_{i=1}^n U_i^*$ turns out quite different from its expected value of n_Y ; in fact, it is even possible (although very unlikely) that $\sum_{i=1}^n U_i^*$ is 0 or n , i.e., the (pseudo)sample consisting of data from one group only.

The above discussion refers to bootstrap and bagging but similar ideas hold for single-sample subbagging. Let us define a (pseudo)sample to be *balanced* if the proportion of patients to control subjects within the (pseudo) sample is equal to that found in the original sample, i.e., n_Y/n_X . If we let $Z_i^* = (\underline{w}_i^*, U_i^*)$ for $i = 1, \dots, b$ be the subsampling (pseudo)sample, then it is still possible to have $\sum_{i=1}^n U_i^* = 0$ provided of course that $b \leq n_X$. But even barring such extreme events, it is clear that there is no guarantee that the above subsampling (pseudo)sample would be balanced.

In conclusion, the possibility of unbalanced (pseudo)samples might not adversely influence the properties of bagging/subbagging for classification purposes but it is problematic in our hypothesis testing setting. The balanced bagging/subbagging procedures of Section 4 are devoid of this deficiency, since they yield—by design—exactly balanced (pseudo)samples.

Finally, note that different resampling methods have been used in connection with multiple comparisons—the most popular of which involving permutation tests; see e.g. Westfall and Young (1993), Ge, Dudoit and Speed (2003), or Romano and Wolf (2004). In addition, the re-calculation of rejection lists over subsamples was considered by Newton et al. (2004) for the purpose of validating the stability of a particular list-forming method. Nevertheless, the approach of Section 4 constitutes the first—to our knowledge—

application of the notion of bagging/subbagging for the purpose of increasing detection power in multiple comparisons.

7 A simulation experiment

The balanced bagging/subbagging procedures of Section 4 are now implemented in the context of a small simulation. 199 ‘true’ datasets satisfying eq. (2) and (3) were generated using $N = 1,000$, $n_X = 18$, $n_Y = 24$, $\mu_k = 0$ and $\sigma_k^2 = 1$ for all k ; also $\nu_j = c$ for $j = 1, \dots, 75$, $\nu_j = -c$ for $j = 76, \dots, 150$, and $\nu_j = 0$ for $j > 150$ for some value c . Note that there are 150 truly non-null genes among the 1,000.

	DP	FDR
$c = 1/2$	0.04	0.08
$c = 2/3$	0.09	0.03
$c = 3/4$	0.18	0.03
$c = 4/5$	0.26	0.04
$c = 0.9$	0.44	0.03
$c = 1.0$	0.60	0.03

Table 1. Average discovery power (DP) and FDR as a function of c ; unbagged experiments.

For each of the ‘true’ datasets, the multiple comparisons were carried using the R program `locfdr` of Efron (2005) using its default settings. Table 1 shows the average achieved discovery power (DP), and average achieved false discovery rate (FDR) among the 199 ‘true’ datasets. The FDR is effectively controlled at a level about 0.03. It is apparent that $c = 1$ corresponds to good power, whereas $c = 2/3$ corresponds to low power. Values of c lower than $2/3$ give power that is so low that it is comparable to (or lower than) the FDR and the experiment is practically useless. We illustrate this phenomenon by including the case $c = 1/2$; note that even FDR control does not work well in this problematic case, and—as will be seen below—bagging/subbagging can not help remedy this case.

‘Maximum contrast’ nonrandom subbagging was implemented; since n_X, n_Y are of the same order of magnitude, the simple choice $m_X = m_Y = m = 6$

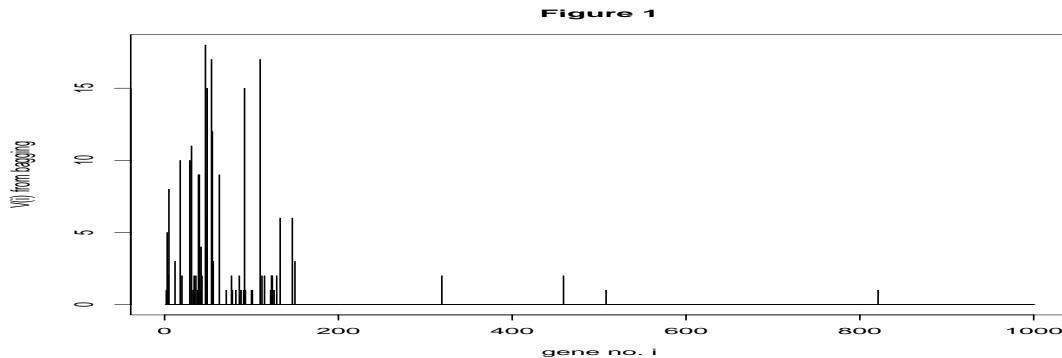


Figure 1: Plot of function $V(i)$ in a typical bagging experiment with $B=36$; case $c = 4/5$.

was used, i.e., subbagging involved splitting each of the two groups in six equal parts, and deleting one of those parts each time. Balanced bagging was also implemented; the choice $B = m^2 = 36$ was used for the purpose of comparing bagging and subbagging based on the same number of (pseudo)samples generated.

Figures 1 and 2 show plots of the voting function $V(i)$ defined in Section 5 in a typical simulation with $c = 4/5$ involving bagging and subbagging respectively. Recall that the truly non-null genes are the ones with indices $i = 1, \dots, 150$; both plots bring this out, and could be used as effective diagnostic tools as suggested in Section 5. The two figures are quite different, however; the most prominent difference is that in bagging you see a gene getting as many as 17 votes, whereas the maximum is 7 votes in subbagging. This is a manifestation of the ‘maximum contrast’ phenomenon related to the particularly designed subbagging algorithm. In other words, the subbagging (pseudo)samples are chosen to be very different from one another, and hence ‘vote’ differently; by contrast, in random bagging, there are many (pseudo)samples of a similar composition that tend to ‘vote in unison’.

This phenomenon is further manifested in Table 2 that shows the average discovery power (DP) and FDR associated with bagging and subbagging as a function of c ; formula (i) from Section 5 was used to combine the votes, i.e., at least one vote gets a gene rejected. As expected, bagging and subbagging both improve power at the cost of an increased FDR. Subbagging is seen to

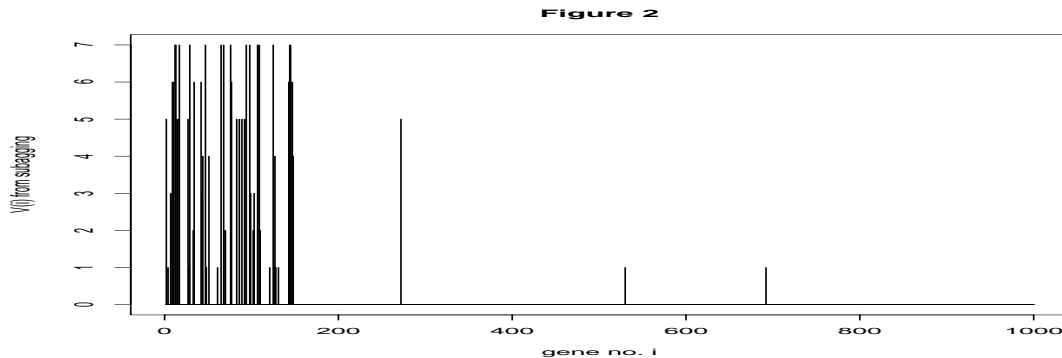


Figure 2: Plot of function $V(i)$ in a typical ‘maximum contrast’ subagging with $m = 6$; case $c = 4/5$.

be uniformly better than bagging with same number of (pseudo)samples in the sense that it has both better power and smaller FDR.

Comparing subagging to the original experiment of Table 1, it is apparent that subagging is most helpful in low power situations, i.e. $c \in [2/3, 4/5]$, where it almost doubles the power while controlling the FDR to about 0.10. As previously mentioned, subagging does not help much in the problematic case $c = 1/2$ where the power is almost zero. Subagging does improve the power in the ‘good-power’ cases $c = 0.9$ and 1, but in those cases the improvement may not be worth the cost in FDR.

	Bag DP	Bag FDR	Sub DP	Sub FDR
$c = 1/2$	0.19	0.36	0.13	0.21
$c = 2/3$	0.24	0.19	0.21	0.10
$c = 3/4$	0.32	0.15	0.33	0.10
$c = 4/5$	0.40	0.13	0.45	0.11
$c = 0.9$	0.57	0.13	0.64	0.12
$c = 1.0$	0.73	0.16	0.78	0.13

Table 2. Average discovery power (DP) and FDR associated with bagging with $B = 36$, and ‘maximum contrast’ subagging with $m = 6$; rejection is declared with at least one vote.

To use bagging/subagging with a smaller potential increase in FDR, we

now employ formula (ii) from Section 5 that needs two votes to get a gene rejected; Table 3 shows the corresponding powers and FDR. As expected, both power and FDR are decreased comparing to Table 2 since formula (ii) is more conservative and does less aggregation.

Recall that in Section 2, the simplifying assumption was made that a gene declared non-null in two different studies is very likely truly non-null. However, the plausibility of such an assumption only holds when the two studies are independent not recycled as in the bagging/subbagging case considered here. So it is of no surprise that the FDRs reported in Table 3 are bigger than zero.

	Bag DP	Bag FDR	Sub DP	Sub FDR
$c = 1/2$	0.14	0.21	0.14	0.18
$c = 2/3$	0.14	0.07	0.18	0.08
$c = 3/4$	0.21	0.05	0.28	0.07
$c = 4/5$	0.28	0.05	0.39	0.07
$c = 0.9$	0.44	0.05	0.58	0.08
$c = 1.0$	0.63	0.06	0.75	0.09

Table 3. Average discovery power (DP) and FDR associated with bagging with $B = 36$, and ‘maximum contrast’ subbagging with $m = 6$; rejection is declared when a gene gets at least *two* votes.

As evidenced by Table 3, subbagging performs uniformly better than bagging with same number of (pseudo)samples even under the formula (ii) voting scheme. Comparing formula (ii) subbagging to the original data of Table 1 the results are quite encouraging; barring the problematic case of $c = 1/2$, subbagging is seen to *substantially* improve power while controlling the FDR to a low level of about 0.07 or 0.08.

For completeness, it is interesting to see how bagging performs with an increased B , i.e., number of (pseudo)samples. Table 4 contains power and FDR associated with bagging with $B = 99$ using both formulas (i) and (ii); as expected, the power is improved by increasing B but so is the FDR. It still seems that our ‘maximum contrast’ subbagging has an edge over bagging, yielding similar power but significantly less FDR, even when bagging uses a higher B .

	DP (i)	FDR (i)	DP (ii)	FDR (ii)
$c = 1/2$	0.17	0.47	0.11	0.37
$c = 2/3$	0.36	0.27	0.25	0.14
$c = 3/4$	0.43	0.22	0.31	0.10
$c = 4/5$	0.49	0.21	0.38	0.10
$c = 0.9$	0.66	0.22	0.55	0.10
$c = 1.0$	0.82	0.27	0.75	0.14

Table 4. Average discovery power (DP) and FDR associated with *bagging* with $B = 99$; first two columns correspond to formula (i) and last two columns to formula (ii).

Finally, formula (ii*) from Section 5 was also tried out where still two votes are needed to get a gene rejected but the original rejection list casts a double vote. We expected results that would be intermediate between Tables 2 and 3 but, surprisingly, the resulting table was almost identical—with the subbagging part being *exactly* identical—to Table 3; thus, the formula (ii*) table is omitted to save space. The interpretation of this finding is that each gene rejected by the original *LIST* was also rejected by at least one of the (pseudo)sample lists $LIST_1, \dots, LIST_B$. It is informative to know that formulas (ii) and (ii*) are inter-changeable but forced to choose between the two, formula (ii*) seems like a better bet.

8 Concluding remarks and real data example

As expected from the discussion in Sections 2 and 4 and confirmed by the simulation results of Section 7, bagging and subbagging generally succeed in improving the experiment’s discovery power at a small cost in increased FDR. Thus, if the objective is to control the FDR of the bagged/subagged experiment to a certain level α , say, then the target FDR of each (pseudo)sample experiment must be chosen to be less than α ; the choice of FDR for the (pseudo)sample experiments would be the result of a *calibration* procedure for which simulation experiments like the above can be helpful. Note also that ‘maximum contrast’ subbagging seems to generally have an edge over bagging, yielding similar power but significantly less FDR.

To conclude, we now apply subagging to the well-known prostate cancer dataset of Singh et al. (2002) that has been analyzed extensively by Efron (2006); this is a ‘low power’ experiment, and thus could potentially benefit most from subagging. In the prostate dataset, there are $n_X = 50$ normal subjects, and $n_Y = 52$ patients; on each subject expression levels for $N = 6033$ are recorded.

To apply ‘maximum contrast’ subagging, the simple choices $m_X = 10$ and $m_Y = 13$ were used mostly for divisibility purposes; they correspond to delete- d with $d_X = 5$ and $d_Y = 4$. The data were pre-processed via a cube-root transformation as in, for example, Tusher, Tibshirani and Chu (2001). Efron’s (2005) `locfdr` method was used to perform the multiple comparisons using two different thresholds, $\text{thr}=0.2$ and $\text{thr}=0.3$. The rejection lists for the original data, and formula (i) and (ii*) ‘maximum contrast’ subagging were compiled and given in the Appendix; their sizes are given in Table 5 where is seen that subagging roughly *triples* the number of genes declared non-null.

	data	subag (i)	subag (ii*)
thr=0.2	34	113	101
thr=0.3	51	142	123

Table 5. Numbers of non-null genes as found by Efron’s `locfdr` method in combination with ‘maximum contrast’ subagging; ‘thr’ indicates the `locfdr` threshold.

Because of the potential increase in FDR that comes with bagging, the lower threshold $\text{thr}=0.2$ might be recommended—which is also `locfdr`’s default. Of the subagging formulas, one might prefer formula (ii*) subagging for reasons of being conservative. The plot of function $V^*(i)$ corresponding to the default threshold is given in Figure 3 where it is apparent that there are many genes that got an enormous number of votes; in fact, there are seven genes that were voted by the original list as well as *every* subagging list. This appears to be a major difference between the real data—where some genes are indeed more relevant than others, and our simulation—where the rows are exchangeable in nature; compare Figure 3 to Figure 2 where the most prominent genes only received a vote from a fifth of the subagging lists.

This phenomenon is shown clearly in the plot of function $N^*(h)$ of Fig-

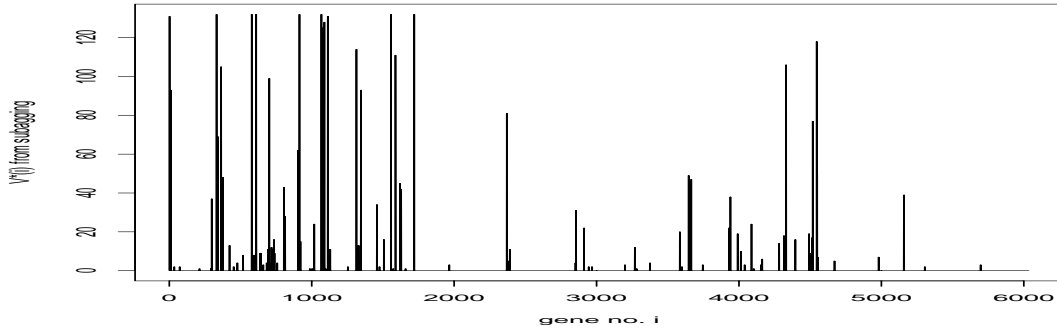


Figure 3: Plot of function $V^*(i)$ in subbagging the prostate data.

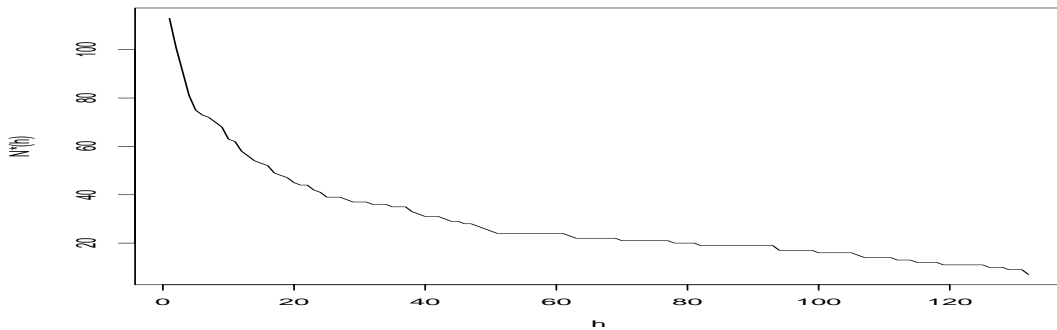


Figure 4: Plot of function $N^*(h)$ vs. h in subbagging the prostate data.

ure 4. The left hand side end of the plot (where h equals one or two) corresponds to the respective sizes (113 and 101) of the formula (i) and (ii*) lists mentioned above. The right hand side end of the plot corresponds to the case $N^*(132) = 7$, i.e., the seven genes voted by every list.

Acknowledgement. The idea for this paper was inspired by Brad Efron’s talk “Doing Thousands of Hypothesis Tests at the Same Time”; the author is grateful to Prof. Efron for many helpful discussions, and for sharing his software and data. The support of NSF via grant SES-04-18136 is also gratefully acknowledged.

References

- [1] Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing, *J. Roy. Statist. Soc., Ser. B*, 57, 289-300.
- [2] Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24, 123-140.
- [3] Bühlmann, P. (2003). Bagging, subbagging and bragging for improving some prediction algorithms, *Recent Advances and Trends in Nonparametric Statistics*, (M.G. Akritas and D.N. Politis, Eds.), Elsevier (North Holland), pp. 19-34.
- [4] Bühlmann, P. and Yu, B. (2002). Analyzing bagging, *Ann. Statist.*, 30, 927-961.
- [5] Dettling, M. (2004). BagBoosting for tumor classification with gene expression data, *Bioinformatics*, vol. 20, no. 18, 3583-3593.
- [6] Dudoit, S. and Fridlyand, J. (2003). Bagging to improve the accuracy of a clustering procedure, *Bioinformatics*, vol. 19, no. 9, 1090-1099.
- [7] Dudoit, S. Fridlyand, J. and Speed, T. (2002). Comparison of discrimination methods for the classification of tumors using gene expression data, *J. Amer. Statist. Assoc.*, vol. 97, no. 457, 77-87.
- [8] Efron, B. (1979). Bootstrap methods: another look at the jackknife, *Ann. Statist.*, 7, 1-26.
- [9] Efron, B. (2005). Local false discovery rates, Preprint available from: <http://www-stat.stanford.edu/~brad/papers>.
- [10] Efron, B. (2006). Size, power, and false discovery rates, Preprint available from: <http://www-stat.stanford.edu/~brad/papers>.
- [11] Efron, B. and Tibshirani, R.J. (1993), *An Introduction to the Bootstrap*, Chapman and Hall, New York.
- [12] Ge, Y., Dudoit, S. and Speed, T. (2003). Resampling-based multiple testing for microarray data analysis, *Test*, vol. 12, no. 1, 1-77.

- [13] Miller, R.G. (1981). *Simultaneous Statistical Inference*, 2nd Ed., Springer, New York.
- [14] Newton, M.A., Noueir, A., Sarkar, D., and Ahlquist, P. (2004). Detecting differential gene expression with a semiparametric hierarchical mixture method *Biostat.*, 5, 155-176.
- [15] Politis, D.N., Romano, J.P. and Wolf, M. (1999), *Subsampling*, Springer Verlag, New York, 1999.
- [16] Romano, J.P. and Wolf, M. (2004). Exact and approximate stepdown methods for multiple hypothesis testing. *Journal of the American Statistical Association*, 100, 94-108.
- [17] Shao, J. and Wu, C.F. (1989). A general theory of jackknife variance estimation, *Ann. Statist.*, 17, 1176-1197.
- [18] Singh, D., Febbo, P.G., Ross, K., Jackson, D.G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A.A., D'Amico, A.V., Richie, J.P., Lander, E.S., Loda, M., Kantoff, P.W., Golub, T.R., and Sellers, W.R. (2002). Gene expression correlates of clinical prostate cancer behavior, *Cancer Cell*, vol. 1, no. 2, 203-209.
- [19] Tukey, J.W. (1958). Bias and confidence in not quite large samples, (Abstract) *Ann. Math. Statist.*, 29, 614.
- [20] Tusher, V.G., Tibshirani, R., and Chu, G. (2001). Significance analysis of microarrays applied to the ionizing radiation response, *Proc. Natl. Acad. Sci. U.S.A.*, 98, 5116-5121.
- [21] Westfall, P. and Young, S. (1993). *Resampling-based multiple testing: examples and methods for p-value adjustment*, Wiley, New York.

APPENDIX

The full rejection lists are given below, i.e., the gene (row) numbers declared non-null corresponding to the methods reported in Table 5.

Case thr= 0.2:

Data LIST: 2 11 298 332 341 364 377 579 610 702 805 905 914 1068 1077 1089
1113 1314 1346 1557 1588 1589 1620 1720 2370 3647 3665 3940 4331 4518 4546
4549 5158 5159

Subag (i) LIST: 2 11 35 73 212 292 298 332 341 364 377 423 452 478 518 579
594 610 611 637 642 660 684 692 694 698 702 709 718 721 731 735 739 758 805 813
905 914 921 987 1003 1018 1019 1068 1077 1082 1089 1090 1097 1113 1130 1254
1314 1329 1345 1346 1458 1476 1507 1557 1572 1588 1589 1620 1628 1659 1720
1966 2370 2385 2391 2852 2856 2912 2945 2968 3200 3269 3282 3375 3585 3600
3647 3665 3746 3930 3940 3991 4013 4040 4088 4104 4154 4163 4282 4316 4331
4396 4492 4496 4500 4515 4518 4546 4549 4552 4554 4671 4981 5158 5159 5305
5697

Subag (ii*) LIST: 2 11 35 73 298 332 341 364 377 423 452 478 518 579 594 610
611 637 642 660 684 692 694 698 702 709 718 721 731 735 739 758 805 813 905
914 921 1018 1068 1077 1089 1090 1113 1130 1254 1314 1329 1345 1346 1458 1476
1507 1557 1588 1589 1620 1628 1720 1966 2370 2385 2391 2852 2856 2912 2945
2968 3200 3269 3375 3585 3600 3647 3665 3746 3930 3940 3991 4013 4040 4088
4154 4163 4282 4316 4331 4396 4492 4496 4500 4515 4518 4546 4549 4552 4671
4981 5158 5159 5305 5697

Case thr= 0.3:

Data LIST: 2 11 298 332 341 364 377 579 610 611 637 702 735 805 813 905 914
1068 1077 1089 1113 1130 1314 1345 1346 1458 1507 1557 1588 1589 1620 1628
1720 2370 2856 2912 3647 3665 3940 3991 4088 4316 4331 4396 4492 4515 4518
4546 4549 5158 5159

Subag (i) LIST: 2 11 35 44 73 78 212 249 263 270 292 298 332 341 364 377 423
452 478 493 518 579 594 610 611 626 637 642 660 684 692 694 698 702 709 718 721
731 735 739 742 758 805 813 832 844 905 913 914 921 987 1003 1018 1019 1068
1077 1082 1089 1090 1097 1113 1130 1132 1254 1314 1329 1345 1346 1362 1458
1476 1491 1507 1508 1557 1566 1572 1588 1589 1620 1628 1643 1659 1702 1720
1872 1966 2370 2385 2391 2785 2852 2856 2912 2945 2968 3200 3208 3260 3269
3282 3375 3585 3600 3647 3665 3746 3930 3940 3961 3991 4013 4040 4057 4073

4088 4104 4154 4163 4282 4316 4331 4386 4396 4492 4496 4500 4510 4515 4518
4546 4549 4552 4554 4671 4981 5158 5159 5305 5547 5647 5697

Subag (ii*) LIST: 2 11 35 73 212 292 298 332 341 364 377 423 452 478 493 518
579 594 610 611 637 642 660 684 692 694 698 702 709 718 721 731 735 739 742 758
805 813 844 905 913 914 921 987 1003 1018 1019 1068 1077 1082 1089 1090 1097
1113 1130 1254 1314 1329 1345 1346 1362 1458 1476 1507 1508 1557 1588 1589
1620 1628 1659 1720 1966 2370 2385 2391 2852 2856 2912 2945 2968 3200 3208
3260 3269 3282 3375 3585 3600 3647 3665 3746 3930 3940 3991 4013 4040 4073
4088 4104 4154 4163 4282 4316 4331 4396 4492 4496 4500 4515 4518 4546 4549
4552 4554 4671 4981 5158 5159 5305 5547 5647 5697