Paper 190-31

Estimating the False Discovery Rate using SAS [®]

Jason A. Osborne, North Carolina State University, Raleigh, NC

ABSTRACT

This paper gives an exposition of recently developed methods of estimating the false discovery rate (FDR) under multiple comparisons and discusses their implementation using SAS. For example, microarray experiments typically involve tests of significance for hundreds or thousands of genes. For biologists confronted by this problem of multiplicity, the FDR is an appealing quantification of error. Recent literature establishes theoretical properties of an estimator for FDR associated with any chosen critical region. SAS code using the SAS/MACRO, SAS/STAT, and SAS/IML facilities is presented to compute this estimate, and appears in the appendix. Q-values, which are to FDR as p-values are to the type I error rate, are also obtained and used to reproduce plots generated by the R package "qvalue" that may be helpful for conducting multiple tests. Two approaches to estimating the proportion of true null hypotheses are considered. The first arises by choosing several values of a tuning parameter and fitting a smoothing spline (e.g. using PROC TPSPLINE) to the resulting estimates. The second is to fit a two-component mixture to the sample of observed p-values via maximum likelihood (using PROC NLMIXED). A microarray example using p-values from tests on 384 genes is given.

INTRODUCTION

The false discovery rate (FDR) has received much attention as an alternative way of quantifying type I errors in multiple comparisons problems. One reason for this attention is the development of high throughput technology in the field of genomics that allow for experiments to test many hypotheses simultaneously. Microarray technology, for example, has facilitated the measurement of expression intensity for thousands of genes at once across different conditions in a single experiment. As an illustration, consider the "volcano" plot in Figure 1, which is taken from a microarray experiment Li et al., 2005 to investigate gene expression during gestation of *H. parviporum*, a fungal pathogen causing root rot in Norway Spruce. The experiment measured mRNA abundance from tissue harvested at timepoints 0, 18, 36, 72 and 120 hours post-germination in a completely randomized design with three biologically independent samples per timepoint. Single channel expression measurements on 384 cDNA clones ("genes") are made for each replicate. In this volcano plot, observed significance level ($\log_{10}$ scale) is plotted against fold-change, or difference of treatment means after $\log_2$ transformation. There is a point in the plot and corresponding test of significance for each of the 384 genes. The p-values on the vertical axis are computed from t-tests with 12 error degrees of freedom from the one-way ANOVA. Traditional approaches to controlling error rates in the presence of a large number of comparisons include strong and weak control of familywise error rates, using techniques such as the Bonferroni correction. Warnings about these error rates in the presence of multiplicity are automatically reported in output produced by PROC GLM or PROC MIXED whenever p-values for pairwise comparisons are requested without multiplicity adjustment in MEANS or LSMEANS statements. PROC MULTTEST can also be used to carry out a vast array of multiple comparisons procedures and provide multiplicity-adjusted p-values.
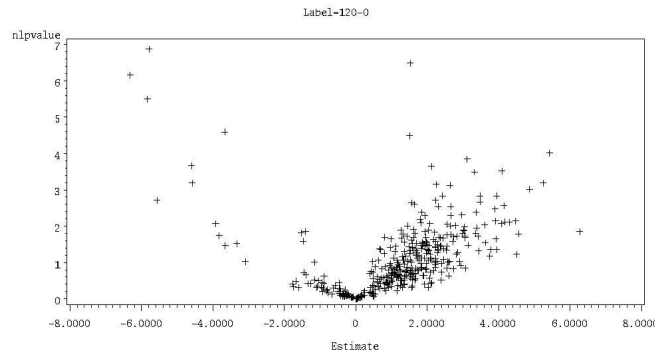
Figure 1. Volcano Plot

THE FALSE DISCOVERY RATE (FDR)

The FDR provides an alternative quantification of error under multiplicity of comparisons. Possible outcomes from multiple testing are given in Table 1, which adopts notation established in Storey (2002).

Table 1: Outcome from multiple tests

Truth	Declared Significant	Not significant	Total
Null is true	F	$m_0 - F$	m_0
Alternative is true	S	$m_1 - S$	m_1
Total	$R = F + S$	$m - R$	m

Several quantifications of error rates are comparisonwise (CER), familywise (FWE) and false discovery (FDR). A procedure that controls these error rates satisfies the following inequalities:

$$E\left(\frac{F}{m_0}\right) \leq CER$$
$$\Pr(F > 0) \leq FWE$$
$$E\left(\frac{F}{R}|R > 0\right) \leq FDR$$

The FDR enjoys a straightforward interpretability for scientists outside of statistics. The same might not be true of comparisonwise or familywise error rates or even individual p-values themselves. This interpretation is as follows: suppose that R tests out of m are declared significant at an FDR of .05, then 5% of these declarations can be expected to be false positives, on average. This is in contrast to the FWE which controls the chance of at least one false positive among many tests for which the null hypothesis is true.

Weak control of the FDR, proposed in Benjamini and Hochberg (1995), has been available as an option in SAS PROC MULTTEST for some time. The steps of this procedure are as follows

1. Order the raw p-values: $p_{(1)} \leq \cdots \leq p_{(m)}$

2. Find $\hat{k} = \max\{k : p_{(k)} \leq k\alpha/m\}$

3. If $\hat{k}$ exists, reject tests corresponding to $p_{(1)}, \ldots, p_{(\hat{k})}$

Equivalently, the adjusted p-values are defined as

$$\tilde{p}_{(m)} = p_{(m)}$$
$$\tilde{p}_{(m-1)} = \min\{\tilde{p}_{(m)}, \frac{m}{m-1}p_{(m-1)}\}$$
$$\vdots \qquad \vdots$$
$$\tilde{p}_{(1)} = \min\{\tilde{p}_{(2)}, mp_{(1)}\}$$

An example with 10 independent p-values, taken from Westfall et al. (1999), illustrates how the step-up procedure works. Suppose a SAS dataset called `pvalues` contains the variable `raw_p`, then the code

```
proc multtest pdata=pvalues bon fdr;
run;
```

will produce output similar to that in Table 2:

Table 2: PROC MULTTEST output

Test	Raw	Bonferroni	False Discovery Rate
1	0.0001	0.0010	0.0010
2	0.0058	0.0580	0.0290
3	0.0132	0.1320	0.0440
4	0.0289	0.2890	0.0723
5	0.0498	0.4980	0.0996
6	0.0911	0.9110	0.1518
7	0.2012	1.0000	0.2874
8	0.5718	1.0000	0.7148
9	0.8912	1.0000	0.9011
10	0.9011	1.0000	0.9011

Upon comparison of the column entitled "Raw" with the sequence of cutpoints $\{.05, .045, \ldots .01, .005\}$, it can be seen that $\hat{k} = 3$ of these tests may be declared significant while controlling the (expected) FDR at .05, and only 1 may be declared significant while using the Bonferroni method to control FWE at .05.

ESTIMATION OF FDR

Storey (2002) takes a different philosophy towards multiple testing, advocating that an *estimate* of the FDR (or any other quantification of error) be reported, along with a standard error, for any particular significance region for an observed dataset. This is a different approach than the one taken above, where the critical region itself ($\hat{k} = 3$) was estimated via the step-up procedure in such a way that the expected FDR is some specified target value such as .05. As shown later, the two approaches may be reconciled, and the list of "significant" genes in a microarray experiment obtained using the FDR option in PROC MULTTEST may be obtained as a special case of a more general procedure.

Consider fixing the critical region by rejecting hypotheses with p-values less than some small $t > 0$. From Table 1,

$$FDR(t) \approx \frac{E[F(t)]}{E[R(t)]} = \frac{tm_0}{E[\#\{p_i \leq t\}]}$$

leading to the estimate

$$\widehat{FDR}(t) \approx \frac{t\hat{m}_0}{\#\{p_i \leq t\}} = \frac{t\hat{\pi}_0 m}{\#\{p_i \leq t\}}$$

where $\hat{\pi}_0$ denotes an estimate of the parameter $\pi_0 = m_0/m$. To obtain such an estimate, consider the probability histogram in the top left panel of Figure 3. This histogram differs from the theoretical uniform distribution because some of the null hypotheses being tested do not hold. Many of the p-values near 0 correspond to tests for genes that are differentially expressed. As long as the experiment has reasonable power to detect differences, most of the rest of the p-values correspond to genes where the null hypothesis holds and there is no underlying differential expression. To estimate the proportion of such genes, π_0, Storey (2002) introduces a tuning parameter λ with $0 < \lambda < 1$. The sample proportion of p-values exceeding λ contains information about π_0 leading to the estimating equation

$$E\left(\frac{\#\{p_i > \lambda\}}{m}\right) \approx (1 - \lambda)\pi_0.$$

An estimate of π_0 may be taken as the value that solves this equation:

$$\hat{\pi}_0(\lambda) = \frac{\#\{p_i > \lambda\}}{m(1 - \lambda)}.$$

For small λ, this estimate will be positively biased, as many of the p-values will correspond to tests of genes that are truly differentially expressed. As $\lambda \to 1$, the estimates have larger and larger variance, since the estimating equation calls for division by $(1 - \lambda)$. Consider, for example, the choices $\lambda = 0.05$ and $\lambda = 0.5$. The proportions of p-values exceeding these choices are $311/384 = .81$ and $113/384 = .294$, respectively. Upon division by $1 - \lambda$, the estimates are $\hat{\pi}_0(\lambda = 0.05) = .85$ and $\hat{\pi}_0(\lambda = 0.5) = .59$, Storey (2002) considers fitting a smooth function to the sequence $\{\hat{\pi}_0(\lambda); \lambda = .5, .55, .6, \ldots, .9\}$ via the `smooth.spline` function available in the R software package (Ihaka and Gentleman, 1996). The same function may be fit using PROC TSPLINE in SAS. Storey's `qvalue` software (Storey, 2002), developed for R, adopts the fitted value of this smooth curve at $\lambda = .9$ for substitution into the expression for $\widehat{FDR}$ and this convention is adopted in the SAS program in the appendix. The plot itself appears in the upper left panel of Figure 2.
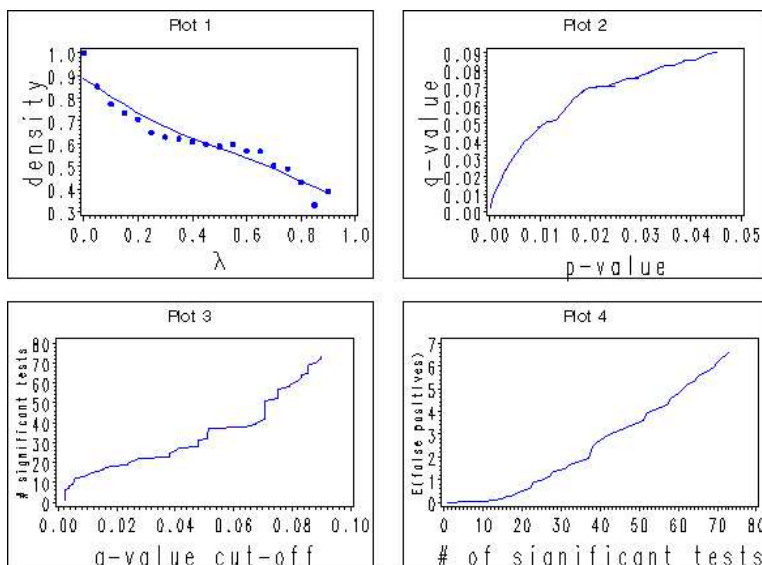

Figure 2 - *H. parviporum* data

Consider the critical region obtained by rejecting hypotheses with p-values less than $t = 0.01$. Upon substitution of this smoother estimate $\hat{\pi}_0(\lambda = 0.9) = 0.379$ and the $R(0.01) = 30$ observed rejections, an estimate of the FDR associated with this critical region is

$$\widehat{FDR}(t = 0.01) = \frac{\hat{\pi}_0 mt}{\#\{p_i < t\}} = \frac{0.379(384)(0.01)}{30} = 0.049$$

A bootstrap standard error associated with this estimate may be obtained by resampling all of the p-values many times. The results of $B = 1000$ such samples, achieved using the "`point=`" option within a datastep, appears in the table below

Table 3: Bootstrap ($B = 1000$) estimation of FDR and π_0

Parameter	Mean (SE)	Median	95% c.i.
π_0	.380(.079)	.378	(.237,.529)
$FDR(.01)$	.050(.015)	.049	(.026,.083) .

Adopting this critical region $p < .01$ leads to declaring $R(.01) = 30$ genes as exhibiting significant differential expression, with $\widehat{FDR} = .05$ and a standard error of .015. An estimate of the expected number of these 30 genes that are false leads, or false discoveries, is $30 \times .05 = 1.5$, which might be acceptably small for many investigators. Alternatively, the Bonferroni correction to control $FWE = .05$ leads to 6 genes exhibiting significant differential expression. This procedure has the property that the chance of at least one false positive, or $\Pr(F \geq 1)$ is at most .05. If no adjustment is made for multiplicity and CER is controlled at .05, 73 genes out of $m = 384$ are declared significant, so that in the complete absence of any differential expression among any of the genes, $.05 \times 384 = 19$ false positives are expected.

4

Q-VALUES

In multiple testing, the multiplicity-adjusted p-value for a particular null hypothesis being tested is the smallest FWE at which the test may be declared significant. Analagously, the q-value (Storey, 2002) is the smallest estimated FDR at which the test may be declared significant:

$$q\text{-value}(p_i) = \min_{t \geq p_i} \widehat{FDR}(t).$$

For genes in the germination data with small p-values, a plot of the q-values against the (unadjusted) p-values appears in the upper right panel of Figure 2. Note that while the q-values exceed the p-values over this range, the q-values fall below the p-values further towards unity on the horizontal axis (corresponding to genes without much evidence of differential expression). Plots which can be helpful for adaptive choice of FDR are given in the bottom of Figure 2. The plot on the left tells the investigator how many genes may be declared significant for different choices of a q-value cutoff. The plot on the right reports the consequent expected number of false positives associated with such a cutoff. This adaptive approach may seem like anathema to researchers with classical training in multiple testing. However, there are problems associated with the traditional approach of *estimating* a critical region so that, on average, the FWE has some target such as .05 as its expected value in repeated experimentation. The approach of considering several critical regions and estimating the statistical properties and consequences of such considerations does not seem unreasonable.

As illustrated in Storey (2002) there is an equivalence between the procedure carried out using the `FDR` option in PROC MULTTEST and the method of estimating FDR described earlier. For m p-values, the method of Benjamini and Hochberg (1995) finds $\hat{k}$ such that

$$\hat{k} = \max\{k : p_{(k)} \leq (k/m)\alpha\}$$

and rejects $p_{(1)}, \ldots, p_{(\hat{k})}$. In repeated sampling, this procedure has expected FDR not exceeding α. The method described here, which requires the estimator $\hat{\pi}_0$ finds $\hat{l}$ such that

$$\hat{l} = \max\{l : \widehat{FDR}(p_{(l)}) \leq \alpha\}.$$

But

$$\widehat{FDR}(t = p_{(l)}) = \frac{\hat{\pi}_0 p_{(l)}}{l/m}.$$

With $\hat{\pi}_0 = 1$ this is equivalent to

$$\hat{l} = \max\{l : p_{(l)} \leq (l/m)\alpha\}.$$

If $\hat{\pi}_0 < 1$, then $\hat{l} > \hat{k}$ with high probability. For example, in the *H. parviporum* experiment with $\alpha = 0.05$, $\hat{k} = 18$ but $\hat{l} = 31$. This suggests that if there are fewer than $m = 384$ tests where the genes are not differentially expressed, then there are fewer tests to focus on, and more power to declare significance. It is worth noting that without formal quantification through some prior distribution or Bayesian approach, it is typically believed that there are some genes that are differentially expressed. In fact, the majority of genes under study are placed on the genechip and included in the experiment because there is prior evidence to suggest their importance and their likely differential expression. A much smaller number of "housekeeping" genes that are not believed to be differentially expressed are also frequently included on these chips. Further, in addition to this *a priori* expectation of differential expression, inspection of the histogram of p-values strongly suggests that $\pi_0 << 1$. Proceeding with a multiple comparisons procedure that fixes $\pi_0 = 1$ is likely to be unnecessarily conservative.

A TWO-COMPONENT MIXTURE MODEL FOR THE DISTRIBUTION OF P-VALUES

Introducing the tuning parameter λ allows the information contained in the distribution of p-values to be used for inference about π_0 without making any parametric assumptions about that distribution. To do this requires choosing a value of the tuning parameter at which to predict the fitted value for π_0. An alternative approach (Allison et al., 2002) is to model the distribution of p-values as a mixture of a uniform distribution, with weight π_0, for genes where the null hypothesis of no differential expression holds and some other distribution, with weight $1 - \pi_0$, corresponding to the other genes. The beta distribution has a fairly

rich variety of shapes and may provide a reasonable fit for this mixture. The resulting density function for a p-value is then

$$f(p; a, b) = \pi_0 + (1 - \pi_0)\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}p^{a-1}(1-p)^{b-1}$$

where the support is $0 < p < 1$ and a, b parameterize the beta component. This likelihood function is easily maximized using the code below, which assumes the variable `raw_p` contains the p-values:

```
PROC NLMIXED DATA=pvalues;
   PARAMETERS pi0=.5 a=2 b=2;
   pi1=1-pi0;
   loglikelihood=LOG(pi0+pi1*PDF('BETA',raw_p,a,b));
   MODEL raw_p ~ GENERAL(loglikelihood);
RUN;
```

The MLEs from the *parviporum* data are $(\hat{\pi_0}, \hat{a}, \hat{b}) = (0.44, 0.51, 2.03)$. The fitted density is overlaid on the histogram of the p-values in Figure 3. As an additional illustration of agreement, a random sample of size $m = 384$ is generated from the fitted two-component mixture. The quantiles of this sample may be plotted against the observed p-values in a two-sample QQ-plot, as in the top right panel of Figure 3. Neither this plot nor the histogram with overlaid fitted density indicate any obvious lack-of-fit of the mixture model.
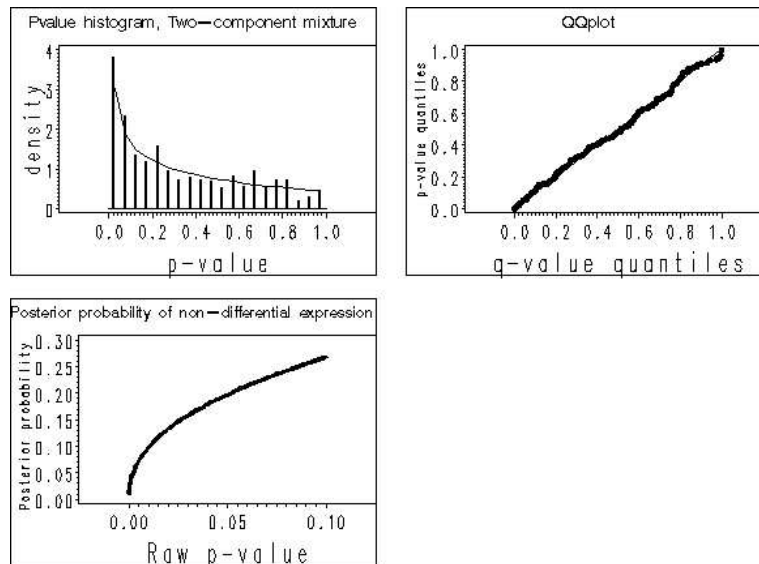

Figure 3- *H. parviporum* data

This parametric model then provides an alternative estimator for π_0 which can be substituted in the expression for $\widehat{FDR}$ and also used for comparison with the smoother estimate based on some choices for the tuning parameter λ. Additionally, the two-component mixture approach leads to posterior probabililty of equal means (or non-differential expression in microarray) conditionally on an observed p-value for test i:

$$\Pr(H_0|p_i) = \frac{\pi_0}{\pi_0 + (1 - \pi_0)f(p_i; \hat{a}, \hat{b})}$$

While this estimated probability may be used to quantify an error rate, it is important that it not be confused with the observed level of significance, p_i. The bottom left panel of Figure 3 shows a plot of these posterior probabilities against observed p-values.

CONCLUSION

The false discovery rate associated with choosing a significance region for a large collection of p-values provides an interpretable quantification of error in multiple comparisons problems. An estimate of the proportion of true nulls, π_0, may be based on the nonparametric approach of Storey (2002) or the mixture model approach of Allison et al. (2002). The SAS code in the appendix carries out these computations

leading to estimates of the false discovery rate. A bootstrap procedure in which the p-values are resampled enables users to estimate their standard errors as well. The plots in Figures 2 and 3 are easily generated and can be useful tools for multiple testing problems such as microarray experiments investigating potential differential expression in many genes.

SAS and all other SAS Institute Inc. product or servicce names are registered trademarks or trademarks of SAS Institute, Inc. in the USA and other countries. $^{\circledR}$ indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

1. APPENDIX: SAS CODE

```
/* Thanks to Joy Smith, N.C. State University, for help with the coding */
OPTIONS LS=75  PS=1000;
%LET seed=123;
DATA one;
    INFILE "pvalues.txt" END=last;    *a file containing a column of p-values;
    INPUT raw_p;
    IF last THEN CALL SYMPUT("m",_n_);
RUN;
%MACRO pi0comp;    %DO i=0 %TO 18;
DATA tmp;
    SET one;
    lambda=.05*&i;
    pbiggerl=(raw_p>=lambda)/(1-lambda);
RUN;
PROC MEANS DATA=tmp NOPRINT;
    ID lambda;
    VAR pbiggerl;
    OUTPUT OUT=tout&i MEAN=pi0hat;
RUN;
DATA tout&i;  SET tout&i;  KEEP pi0hat lambda; RUN;
PROC APPEND BASE=tout DATA=tout&i; RUN;
%END;    %MEND;
%pi0comp;
DATA tout; SET tout;  KEEP lambda pi0hat; RUN;
PROC TPSPLINE ;
    MODEL pi0hat=(lambda)/DF=3;
    OUTPUT OUT=splot;
RUN;
DATA _null_;
    SET splot END=last;
    IF last THEN CALL SYMPUT("pi0",P_pi0hat);
RUN;
PROC RANK DATA=one OUT=rone TIES=high;
    VAR raw_p;
    RANKS v;
RUN;
DATA rone;
    SET rone;
    preqvalue=&pi0*&m*raw_p/v;
    orderp=_n_;
RUN;
PROC SORT DATA=rone OUT=orderedbyp; BY raw_p; RUN;
PROC IM1;
PRINT &pi0;
USE orderedbyp;
READ ALL VAR{orderp} INTO torderp;
orderp=(torderp)`;
```

```
USE rone;
READ ALL VAR{preqvalue} INTO qvalue;
READ ALL VAR{raw_p} INTO pvalue;
qvalue[orderp[&m]] = MIN(qvalue[orderp[&m]],1);
DO i=(&m-1) TO 1 BY -1;
    qvalue[orderp[i]] = min(qvalue[orderp[i]],qvalue[orderp[i+1]],1);
END;
pqmtx=pvalue||qvalue;
CREATE pqvalues VAR{pvalue qvalue};
APPEND;
CLOSE pqvalues;
QUIT;
DATA pqr;
    SET pqvalues;
    IF pvalue<.05;
RUN;
PROC SORT DATA=pqr; BY qvalue; RUN;
DATA pqrg3;
    SET pqr;
    sigtests=_n_;
    sigXq=sigtests*qvalue;
RUN;
GOPTIONS RESET=all;
SYMBOL1 I=NONE COLOR=BLUE VALUE=DOT;
SYMBOL2 I=JOIN COLOR=BLACK VALUE=DOT;
SYMBOL3 I=JOIN COLOR=BLACK VALUE=DOT;
SYMBOL4 I=JOIN COLOR=BLACK VALUE=DOT;
SYMBOL5 I=JOIN COLOR=BLUE VALUE=NONE;
AXIS1 LABEL=(FONT=GREEK H=3 'l') VALUE=(H=2.0);
AXIS2 LABEL=(ANGLE=90 H=3 'density') VALUE=(H=2.0);
AXIS3 LABEL=(H=3 'p-value')  VALUE=(H=2.0);
AXIS4 LABEL=(ANGLE=90 H=3 'q-value')  VALUE=(H=2.0);
AXIS5 LABEL=(H=3 'q-value cut-off') VALUE=(H=2.0);
AXIS6 LABEL=(ANGLE=90 H=1.5 '# significant tests') VALUE=(H=2.0);
AXIS7 LABEL=(H=3 '# of significant tests') VALUE=(H=2.0);
AXIS8 LABEL=(ANGLE=90 H=1.5 'E(false positives)') VALUE=(H=2.0);
PROC GPLOT DATA=splot;
    TITLE "Plot 1";
    PLOT pi0hat*lambda=1 p_pi0hat*lambda=5/OVERLAY HAXIS=AXIS1 VAXIS=AXIS2;
RUN;QUIT;
PROC GPLOT DATA=pqr;
    TITLE "Plot 2";
    PLOT qvalue*pvalue=5/HAXIS=AXIS3 VAXIS=AXIS4;
RUN;QUIT;
PROC GPLOT DATA=pqrg3;
    TITLE "Plot 3";
    PLOT sigtests*qvalue=5/HAXIS=AXIS5 VAXIS=AXIS6;
RUN;QUIT;
PROC GPLOT DATA=pqrg3;
    TITLE "Plot 4";
    PLOT sigXq*sigtests=5/HAXIS=AXIS7 VAXIS=AXIS8;
RUN; QUIT;
PROC GREPLAY IGOUT=GSEG NOFS;
TC=sashelp.templt;
TEMPLATE=l2r2s;
    TREPLAY 1:GPLOT
            2:GPLOT2
            3:GPLOT1
            4:GPLOT3;
```

```
RUN; QUIT;
PROC NLMIXED DATA=one;
   PARAMETERS pi0=0.5 a=1.5 b=1.5;
   pi1=1-pi0;
   like=LOG(pi0+pi1*PDF('BETA',raw_p,a,b));
   MODEL raw_p ~ GENERAL(like);
   ODS OUTPUT PARAMETERESTIMATES=parmz;
RUN;
DATA plot5;  /* for plot5 */
   INFILE "pvalues.txt";
   SET one;
   ARRAY counts count1-count20;
   ARRAY midpts midpt1-midpt20;
   INPUT raw_p;
   count1=(raw_p<=.05);
   midpt1=.025;
   DO i=2 to  20;
      binedge=.05*i;
      counts{i}=(raw_p<=binedge) * (raw_p>binedge-.05);
      OUTPUT;
   END;
RUN;
PROC MEANS DATA=plot5 NOPRINT NWAY;
   VAR count1-count20;
   OUTPUT OUT=hts MEAN=ht1-ht20;
RUN;
DATA hts;  SET hts;  KEEP ht1-ht20;RUN;
PROC TRANSPOSE DATA=hts OUT=thts PREFIX=height;RUN;
DATA thts;
   SET thts;
   midpt=_n_*.05-.025;
   KEEP height1 midpt;
RUN;
PROC TRANSPOSE DATA=parmz OUT=ests;
   ID Parameter;
   VAR Estimate;
RUN;
DATA plot5;
   IF _n_=1 THEN SET ests;
   SET thts;
   height2=pi0+(1-pi0)*PDF('BETA',midpt,a,b);
   height1=height1*20;
RUN;
DATA two; * for plot6*;
   SET ests;
   DO i=1 TO &m;
      u=RANUNI(&seed);
      IF u<pi0 THEN pvalue=RANUNI(&seed);
      ELSE DO; u=RANUNI(&seed);
      pvalue=BETAINV(u,a,b);
      END;
   OUTPUT;
   END;
   KEEP pvalue;
RUN;
PROC SORT DATA=two;  BY pvalue;RUN;
PROC SORT DATA=one;  BY raw_p;RUN;
DATA plot6; MERGE one two; RUN;
DATA plot7;
```

```
   IF _n_=1 THEN SET ests;
   SET one;
   pdata=pi0+(1-pi0)*PDF('BETA',raw_p,a,b);
   p0data=pi0/pdata;
   p1data=1-p0data;
   IF raw_p>0.1 THEN DELETE;
   KEEP p0data p1data raw_p pi0 a b;
RUN;
GOPTIONS RESET=all;
AXIS1 LABEL=(ANGLE=90 H=2.5 'density' ) VALUE=(H=2);
AXIS2 ORDER=0 TO 1 BY .2 OFFSET=(10) LABEL=(H=3 'p-value' ) VALUE=(H=2);
AXIS3 LABEL=(ANGLE=90 H=1.5 'p-value quantiles' ) VALUE=(H=2);
AXIS4 OFFSET=(10) LABEL=(H=3 'q-value quantiles') VALUE=(H=2);
AXIS5 ORDER=0 to 0.3 by 0.05 LABEL=(ANGLE=90 H=1.5 'Posterior probability' ) VALUE=(H=2);
AXIS6 OFFSET=(10) LABEL=(H=3 'Raw p-value' ) VALUE=(H=2);
SYMBOL1 C=BLACK I=NEEDLE V=NONE W=5;
SYMBOL2 C=BLACK I=J V=NONE;
PROC GPLOT DATA=plot5;
   TITLE "Pvalue histogram, Two-component mixture";
   PLOT height1*midpt height2*midpt/OVERLAY VAXIS=AXIS1 HAXIS=AXIS2 ;
RUN; QUIT;
SYMBOL1 I=NONE VALUE=DOT;
SYMBOL2 I=JOIN VALUE=NONE;
PROC GPLOT DATA=plot6;
   TITLE "QQplot";
   PLOT pvalue*raw_p=1 raw_p*raw_p=2/OVERLAY VAXIS=AXIS3 HAXIS=AXIS4;
RUN; QUIT;
SYMBOL1 I=JOIN VALUE=NONE;
PROC GPLOT DATA=plot7;
   TITLE "Posterior probability of non-differential expression";
   PLOT p0data*raw_p/VAXIS=AXIS5 HAXIS=AXIS6;
RUN; QUIT;
PROC GREPLAY IGOUT=GSEG NOFS;
TC=sashelp.templt;
TEMPLATE=l2r2s;
   TREPLAY 1:GPLOT4
           2:GPLOT6
           3:GPLOT5;
RUN; QUIT;
```

REFERENCES

Allison, D., Gadbury, G., Heo, M., Fernández, J., Lee, C.-K., Prolla, T. and Weindruch, R. (2002). A mixture model approach for the analysis of microarray gene expression data. *Computational Statistics & Data Analysis* **39**, 1–20.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society, Series B* **57**, 289–300.

Ihaka, R. and Gentleman, R. (1996). R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics* **5**, 299–314.

Li, G., Osborne, J. and Asiegbu, F. (2005). Gene expression during conidiospore germination and development in the conifer pathogen *heterobasidion parviporum*. *Applied and Environmental Microbiology (AEM)* **in preparation**.

Storey, J. (2002). A direct approach to false discovery rates. *Journal of the Royal Statistical Society, Series B* **64**, 479–498.

Westfall, P., Tobias, R., Rom, D., Wolfinger, R. and Hochberg, Y. (1999). *Multiple Comparisons and Multiple Tests Using the SAS ® System*. SAS Institute, Inc., Cary, NC.