Page One

Campus Computing News

EduTex 2002 Proceedings Available

Do you have something to tell *Everyone*?

Today's Cartoon

RSS Matters

SAS Corner

The Network Connection

Link of the Month

WWW@UNT.EDU

Short Courses

IRC News

Staff Activities

Subscribe to Benchmarks *Online*

Research and Statistical Support
University of North Texas

RSS Matters

Controlling the False Discovery Rate in Multiple Hypothesis Testing

The previous article in this series can be found in the December, 2001 issue of Benchmarks Online: Dealing with Outliers in Bivariate Data: Robust Correlation - Ed.

By Dr. Rich Herrington, Research and Statistical Support Consultant

This month we demonstrate multiple contrast adjustment using the False Detection Rate method (FDR). The GNU S language, "R" is used to implement this procedure. R is a statistical programming environment that is a clone of the S and S-Plus language developed at Lucent Technologies. In the following document we illustrate the use of a GNU Web interface to the R engine on the "rss" server, http://rss.acs.unt.edu/cgi-bin/R/Rprog. This GNU Web interface is a derivative of the "Rcgi" Perl scripts available for download from the CRAN Website, http://www.cran.r-project.org (the main "R" Website). Scripts can be submitted interactively, edited, and be re-submitted with changed parameters by selecting the hypertext link buttons that appear below the figures. For example, clicking the "Run Program" button below creates a vector of four numbers, displays the results, then sorts and displays the results. To view any text output, scroll to the bottom of the browser window. To view any graphical output, select the "Display Graphic" link. The script can be edited and resubmitted by changing the script in the form window and then selecting "Run the R Program". Selecting the browser "back page" button will return the reader to this document.

```
x<-c(.020,
.012, .048, .001,)
y<-sort(x)
x
y
```

Run Program

Introduction

False Discovery Rate (Benjamini & Hochberg, 1995) is a relatively new statistical procedure that controls the number of mistakes make when performing multiple hypothesis tests. False Discovery Rate (FDR) accomplishes this task by allowing one to control beforehand, the average fraction of false rejections made out of the total number of rejections performed. Furthermore, the FDR procedure is a simple procedure that can be adapted to work with correlated data sets.

It is common in statistical modeling to test whether data is consistent with the predictions of a particular statistical model. In this approach, one tests for overall differences between data and the model. For example, it is common in the social sciences to use mean difference testing (i.e. t-tests and ANOVA modeling) to search for statistically significant differences between group means, beyond those that were tested by a prior hypothesis (post-hoc comparisons (unplanned) as opposed to planned comparisons). In the case of a multi-way ANOVA (e.g. 3 way ANOVA), an "omnibus F-test" is performed. This overall statistical test ascertains whether there are statistically significant pairwise differences between means existing in the data set. In other words, the F-test informs researchers that at least one mean difference exists, but does not provide the information necessary to discern where these differences lie.

Multiple Hypothesis Testing

In accordance with usual ANOVA modeling practice, follow up "post-hoc" tests are performed to delineate which of the pairwise means contributed to the overall significant F-test. With a single t-test, if the mean difference is larger than twice the standard error of measurement, significance is declared. This approach allows one to declare significance erroneously with probability of about 0.05. That is, the usual "nominal" alpha level such as .05 or .01 is used for each test, as though no other comparisons were being made on the data. However, the making of such errors increases rapidly with the number of tests performed, so an adjustment is necessary to be applicable for multiple testing. This nominal alpha level is often referred to as the Type I error rate per comparison, or the PC error rate. In practice however, comparisons are usually tested in sets of comparisons based on the same set of data. This introduces the possibility of making at least one Type I error in the entire set or family of comparisons. This probability of making one or more Type I error errors in the set of comparison tests is know as the familywise error rate or FW error rate. For K independent tests, the FW error rate may be calculated:

$$\alpha_{FW}=1-(1-\alpha_{PC})^K$$

When testing a family of K dependent comparisons with a constant per-comparison error rate α_{PC}, the relation between the FW error rate and the PC error rate is more difficult to specify. Nonetheless, it is true that when we have any K tests using a constant PC for each test, the following relationship must hold:

$$\alpha_{FW} \leq K \cdot \alpha_{PC}$$

An investigator might employ a different PC level for each set of K tests. In this way, more power can be ensured for some sets of tests, presumably more important questions. This is done by making the designated alpha level larger for the more important tests than would be otherwise indicated. The familywise error rate must always be less than or equal to the sum of the error rates over the individual tests. If one wants to make the FW rate no larger than some value, say α_{FW}^*, then we can do so by setting the PC rate for each test at:

$$\alpha_{PC}=(\alpha_{FW}^*)/K$$

so that:

$$\alpha_{FW} \leq \alpha_{FW}^*$$

This approach is sometimes called the Bonferroni test, and can be applied to both independent and dependent tests. The Bonferroni method just outlined can be applied to

post hoc comparisons, although it becomes much too conservative to be practically applicable when many comparisons are made. Alternatively, multiple testing without adjustment allows too many false discoveries in return for more correct detections. While the Bonferroni method tightly controls the propensity for making false discoveries, it also misses many real detections. Testing without adjustment, and the Bonferroni approach represent two opposite extremes in multiple contrast adjustment. The False Detection Rate Method (FDR) represents an intermediate solution between these two extremes, when a large number of tests is conducted.

The False Detection Rate Method (FDR)

Benjamini & Hochberg (1995) suggested the FDR method as an improvement on existing multiple contrast adjustment approaches. FDR has higher power than Bonferroni, and it controls errors better than testing without adjustment, by controlling a different measure of error than Bonferroni and other post-hoc comparison techniques. Bonferroni seeks to control the chance of even a single false discovery among all tests performed. The FDR method controls the proportion of errors among those tests whose null hypothesis were rejected. Thus, FDR attains higher power by controlling the most relevant errors.

The FDR procedure is as follows. First select an alpha between zero and one, $0 \le \alpha \le 1$. Let $P_1, \ldots P_N$ denote the p-values from the N tests, listed from smallest to largest. Let:

$$d = \max\left\{ j : \quad P_j < \frac{j \cdot \alpha}{C_N \cdot N} \right\}$$

where C_N is a constant defined below. Reject all hypothesis whose p-values are less than or equal to P_d. When the p-values are based on statistically independent tests, we take $C_{N=1}$. When the tests are dependent, we take:

$$C_N = \sum_{i=1}^{N} \frac{1}{i}$$

Benjamini & Hochberg (1995) show that the proportion of errors among the rejected tests are no larger than α. That is, $FDR \le \alpha$. As an algorithm, the procedure can be described as (for 10 tests and critical alpha=.05) :

1) Create the vector A by sorting the observed pvalues.
2) Create the vector B by computing $j \cdot \frac{\alpha}{10}$ for j=1,.....10 and α=.05 (in the case of independent tests).
3) Subtract vector A from vector B; call this vector C.
4) Find the largest index, d, (from 1 to 10) for which the corresponding number in vector C is negative.
5) Reject all null hypothesis whose p-value are less than or equal to P_d (d indexes vector A). The null hypothesis for the other tests are not rejected.

An Example Using GNU S ("R")

```
pvalue<-c(.021,
.001, .017, .041, .005, .036, .042, .023, .070, .10)
sorted.pvalue<-sort(pvalue)
sorted.pvalue
j.alpha<-(1:10)*(.05/10)
j.alpha
diff<-sorted.pvalue-j.alpha
diff
neg.diff<-diff[diff<0]
pos.diff<-neg.diff[length(neg.diff)]
index<-diff==pos.diff
index
p.cutoff<-sorted.pvalue[index]
p.cutoff
p.sig<-pvalue[pvalue<=p.cutoff]
p.sig

### A test case comparing the FDR script ### against the "multtest"
#   library. "BH" is the FDR for independent p-values

library(multtest)
procs<-c("Bonferroni","Holm","Hochberg","SidakSS","SidakSD","BH","BY")
rawp<-c(.049, .049, .049)
res2<-mt.rawp2adjp(rawp,procs)
res2
```

Run Program

Results and Conclusion

The resulting vector, "p.sig" is the final vector containing all of the rejections from the null hypothesis - 5 rejections out of 10 statistical tests; "p.cutoff" is the new alpha criterion used to assess significance - 0.023. These statistical detections or discoveries contain at most 5% errors or false rejections. The FDR method increases the power to detect differences while maintaining control of a meaningful measure of error rate. The Bonferroni approach would have the alpha criterion at .005 (.05/10), whereby only 2 of the tests would have been deemed statistically significant. The FDR method is a relatively simple method for multiple contrast adjustment that keeps type II error low (high power), while maintaining control over the number of decision errors for the rejected tests (less than 5% for an alpha criterion of .05).

The results below compare the FDR script in this article against the R library, "multtest", for three p-values: .049, .049, and .049. The results for both are equivalent.

```
> pvalue<-c(.049, .049, .049)
> sorted.pvalue<-sort(pvalue)
> j.alpha<-(1:3)*(.05/3)
> diff<-sorted.pvalue-j.alpha
> neg.diff<-diff[diff<0]
> pos.diff<-neg.diff[length(neg.diff)]
> index<-diff==pos.diff
> p.cutoff<-sorted.pvalue[index]
> p.cutoff
[1] 0.049
> p.sig<-pvalue[pvalue<=p.cutoff]
> p.sig
[1] 0.049 0.049 0.049
```

```
> ### Comparing to "multtest" library
>
> library(multtest)
> procs<-c("Bonferroni","Holm","Hochberg","SidakSS","SidakSD","BH","BY")
> rawp<-c(.049, .049, .049)
> res2<-mt.rawp2adjp(rawp,procs)
> res2
$adjp
      rawp Bonferroni  Holm Hochberg   SidakSS   SidakSD    BH    BY
[1,] 0.049      0.147 0.147    0.049 0.1399146 0.1399146 0.049 0.049
[2,] 0.049      0.147 0.147    0.049 0.1399146 0.1399146 0.049 0.049
[3,] 0.049      0.147 0.147    0.049 0.1399146 0.1399146 0.049 0.049
```

References

Benjamini, Y., Hochberg, Y. (1995). J.R. Stat. Soc. B, Vol 57, page 289.

Benjamini, Y., Yekutieli (1999). J. Stat. Plan. Infer., Vol 82, page 171.

Announcement

GNU S ("R") on SOL

R version 1.4.1 (2002-01-30) is now installed on SOL, UNT's research UNIX computer. To invoke R within your session, type:

~ % /usr/local/R/bin/R

To quit out of an R session, type:

> q( )