GS01 0163
Analysis of Microarray Data

Keith Baggerly and Bradley Broom
Department of Bioinformatics and Computational Biology
UT M. D. Anderson Cancer Center
kabagg@mdanderson.org
bmbroom@mdanderson.org

30 September 2010

Lecture 9: Differential Expression

- One Last Caution on Normalization
- Student's t-test, and Simulations
- Family-wise error rate
- Permutation tests
- Is FWER too conservative?
- Significance Analysis of Microarrays (SAM)
- Beta-uniform mixture model
- Empirical Bayes

Project Normal: A Cautionary Tale

Pritchard, Hsu, Delrow and Nelson

Project Normal: Defining Normal Variance in Mouse Gene Expression
PNAS **98** (2001), 13266-13271.

Data set used for the third annual Critical Analysis of Microarray Data (CAMDA 2002)

Pritchard et al.'s Initial Goals

The goal of many microarray studies is to identify genes that are "differentially expressed".

Relative to what?

Differences larger in scale than those that would be encountered due to "normal" or technical variation.

Try to assess the fraction of genes exhibiting a large mouse-to-mouse heterogeneity in the absence of structure.

Pritchard et al.'s Experimental Design

Eighteen Samples

- Six C57BL6 male mice
- Three organs: kidney, liver, testis

Reference Material

- Pool all eighteen mouse organs

Replicate microarray experiments using two-color fluorescence with common reference and dye swaps

- Four experiments per mouse organ, 2 each dye

Pritchard et al.'s Analysis

Print-tip specific intensity dependent loess normalization

Perform F-tests on log(Exp/Ref) for each gene to see if mouse-to-mouse variance exceeds the array-to-array variance

The Data Supplied

Images

One quantification file each for kidney, liver and testis.

```
CDNA ID, Cluster ID, Title,
Block, Column, Row

F635 Median M1K3_1, B635 Median M1K3_1
F532 Median M1K3_1, B532 Median M1K3_1
```

Mouse 1, Kidney Sample in Cy3 channel, first replicate.

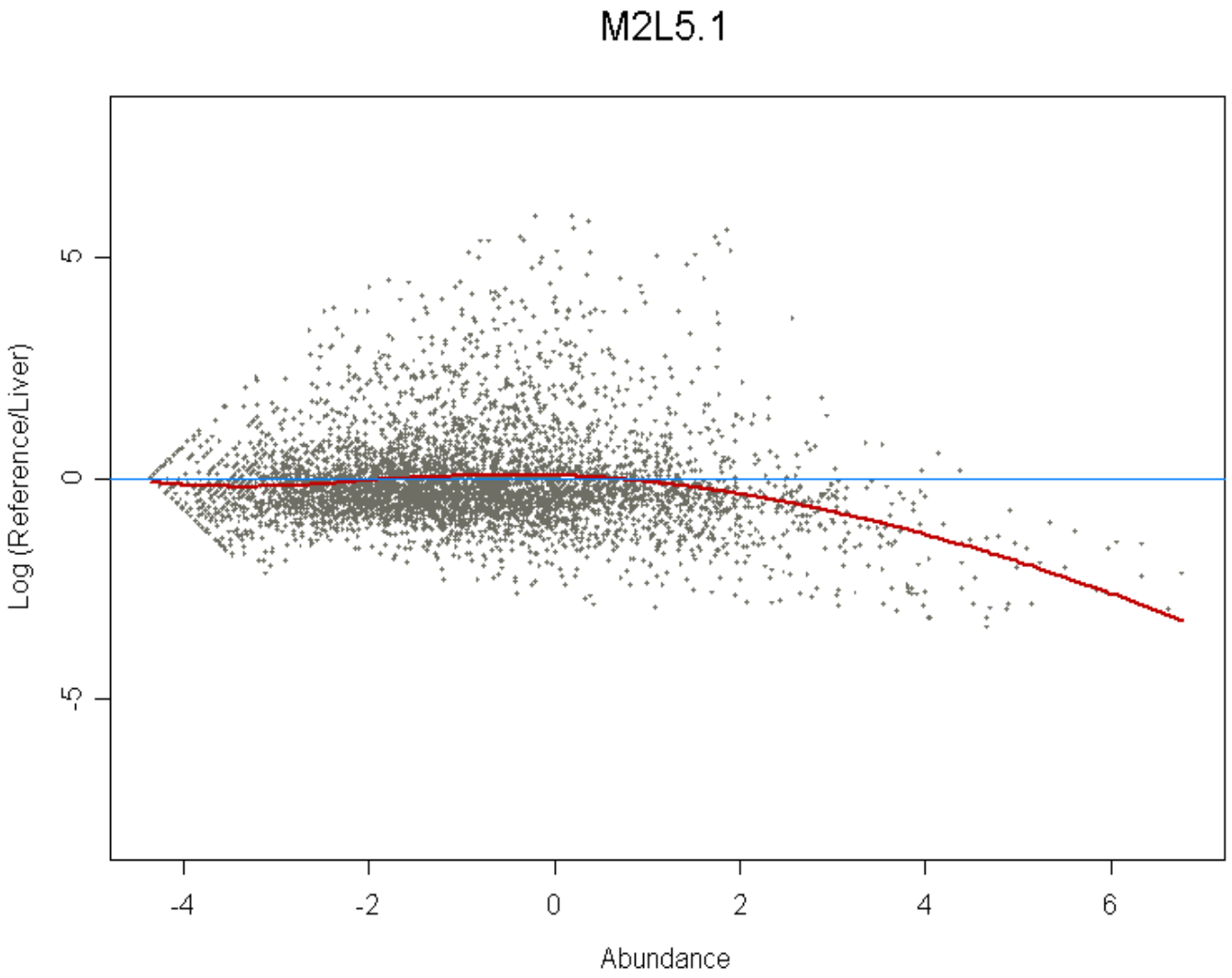
Why We Got Involved

All in all, the analysis described looks pretty good. F-tests on log ratios seem reasonable, and the preprocessing steps they used are fairly standard. Furthermore, the images looked fairly clean.

"Fairly standard" $\neq$ correct

For this data, we think that loess normalization is incorrect.

What Loess Looks Like for 1 Array



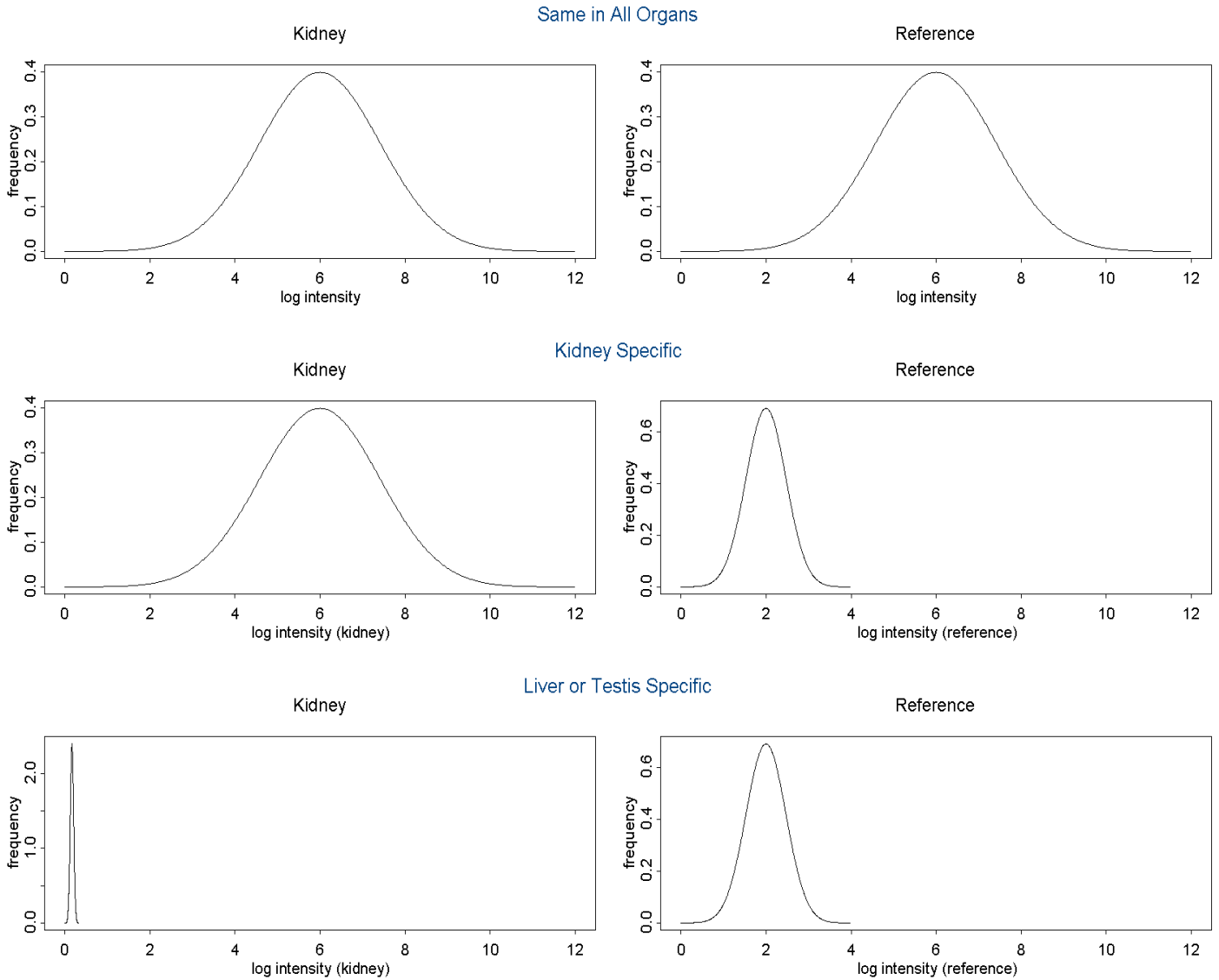
Why Loess Normalization?

Most normalization methods assume:

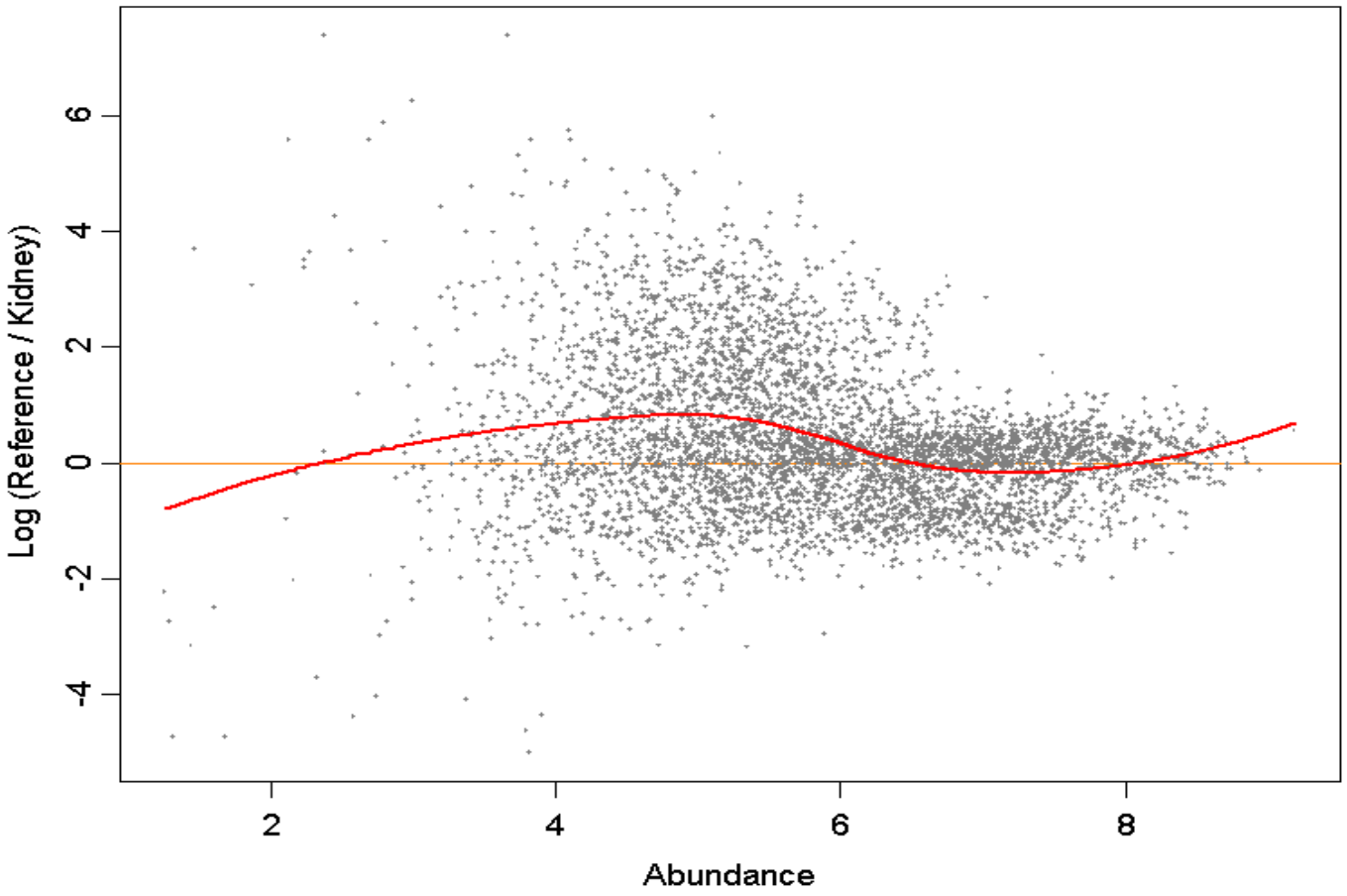
- Distributions of intensities are the same in the two channels
- Most genes do not change expression
- The number of overexpressed genes is about the same as the number of underexpressed genes

Loess normalization tries to force the distributions in the two channels to match, believing that differences are attributable to technology.

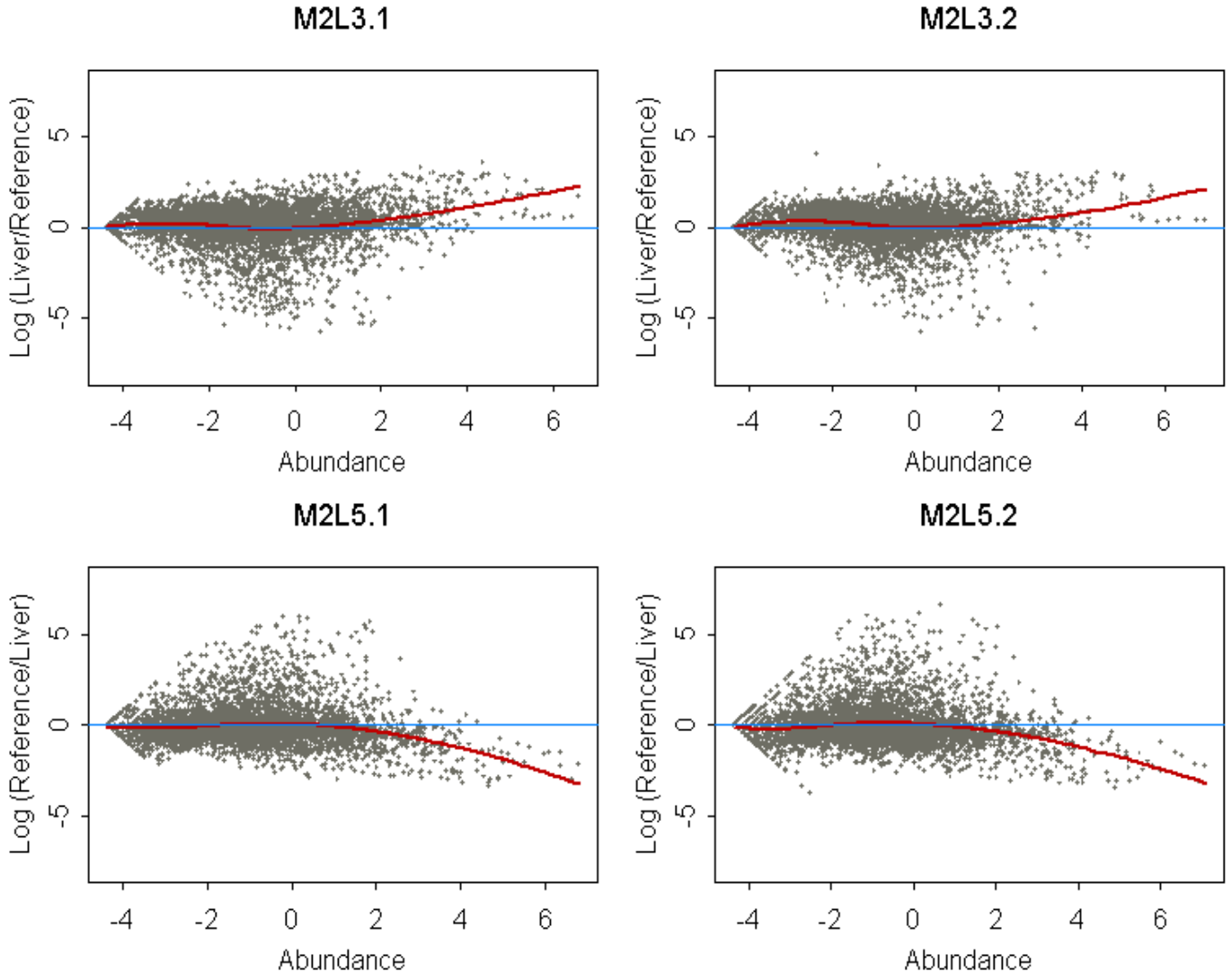
Why We Think It's Wrong



Simulated Data



What's In the Data? Checking Dye Swaps



Interpretation

- Distributions of intensities are different in the two channels
- Difference is NOT caused by arrays, dyes, or technology
- Difference is inherent in the choice of reference material

Class Comparison

Perhaps the most common use of microarrays is to determine which genes are differentially expressed between prespecified groups of samples. We refer to this as the `class comparison` problem. We begin with the simplest case:

- Given microarray experiments on
  - N_A samples of type A
  - N_B samples of type B
- Decide which of the G genes on the microarray are differentially expressed between the two groups.

Student's t-test

In many cases, we analyze microarrays "one gene at a time". That is, we first analyze the same problem for one gene, and then figure out how to adapt that method to thousands of genes.

The one-gene version of the class comparison problem with two classes is Student's t-test. We first estimate the mean and standard deviation in both classes:

$$\bar{x}_A = \frac{1}{N_A} \sum_{i=1}^{N_A} x_i, \qquad s_A^2 = \frac{1}{N_A - 1} \sum_{i=1}^{N_A} (x_i - \bar{x})^2.$$

Weighted difference in means

Next, we "pool" the estimated group standard deviations:

$$s_P^2 = \frac{(N_A - 1)s_A^2 + (N_B - 1)s_B^2}{N_A + N_B - 2}.$$

The two-sample t-statistic is the difference in means, weighted by the pooled standard deviation and the number of samples:

$$t = \frac{\bar{x}_B - \bar{x}_A}{s_P\sqrt{1/N_A + 1/N_B}}.$$

Question: Why not just use the difference in means?

T-statistics

Three ways to get a larger t-statistic:

- Bigger difference in means
- Smaller standard deviation
- More samples

Microarray aside: which scale is best?

Before answering the question, we offer a slight reinterpretation. Most (but not all) analysts believe that microarray data should be transformed by computing logarithms before testing for differential expression. The key mathematical fact supporting this belief is that the logarithm turns multiplication into addition:

$$\log(xy) = \log(x) + \log(y).$$

In particular

$$\log(2x) = \log(x) + \log(2), \qquad \log(x/2) = \log(x) - \log(2).$$

Differences on the log scale can be interpreted as "fold change" on the original scale of the data. Increases and decreases by the same fold change are treated equally on the log scale.

Something New: p-values

Null hypothesis: The mean difference between the two groups is zero.

Two-sided alternative hypothesis: The mean difference is non-zero.

P-value = probability of seeing a t-statistic this extreme under the null hypothesis = area in both tails of the distribution.

Interpretation: if you repeat the experiment many times (with the same numbers of samples), the p-value represents the proportion of times you expect to see a t-statistic this large.

BUT: Computing a t-statistic for each gene on a microarray is like performing the same experiment many times.

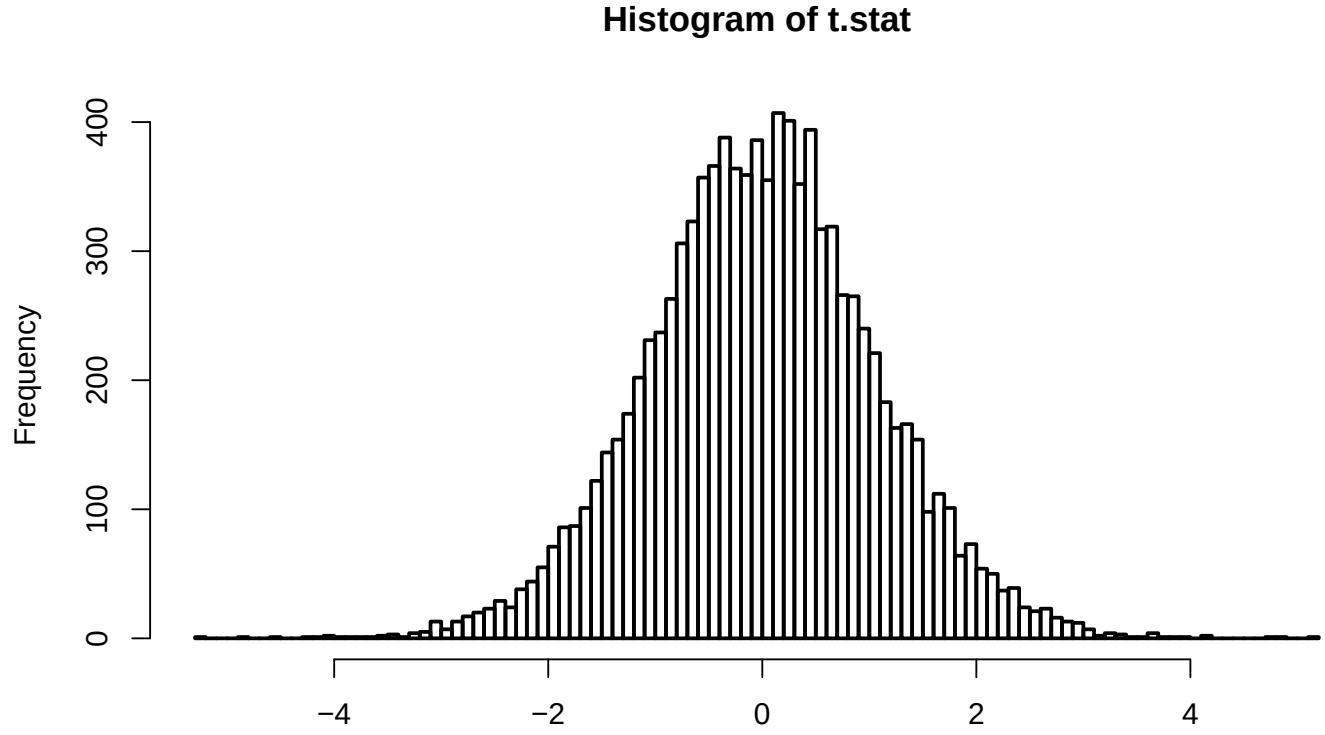
Simulating nothing

We simulated a microarray data set with no differences:

```
> n.genes <- 10000
> n.samples <- 20
> an <- n.samples/2
> bn <- n.samples/2
> type <- factor(rep(c("A", "B"), times = c(an, bn)))
> data <- matrix(rnorm(n.genes * n.samples), nrow = n.ge
> am <- apply(data[, type == "A"], 1, mean)
> bm <- apply(data[, type == "B"], 1, mean)
> av <- apply(data[, type == "A"], 1, var)
> bv <- apply(data[, type == "B"], 1, var)
> sp2 <- ((an - 1) * av + (bn - 1) * bv)/(an + bn -
+     2)
```

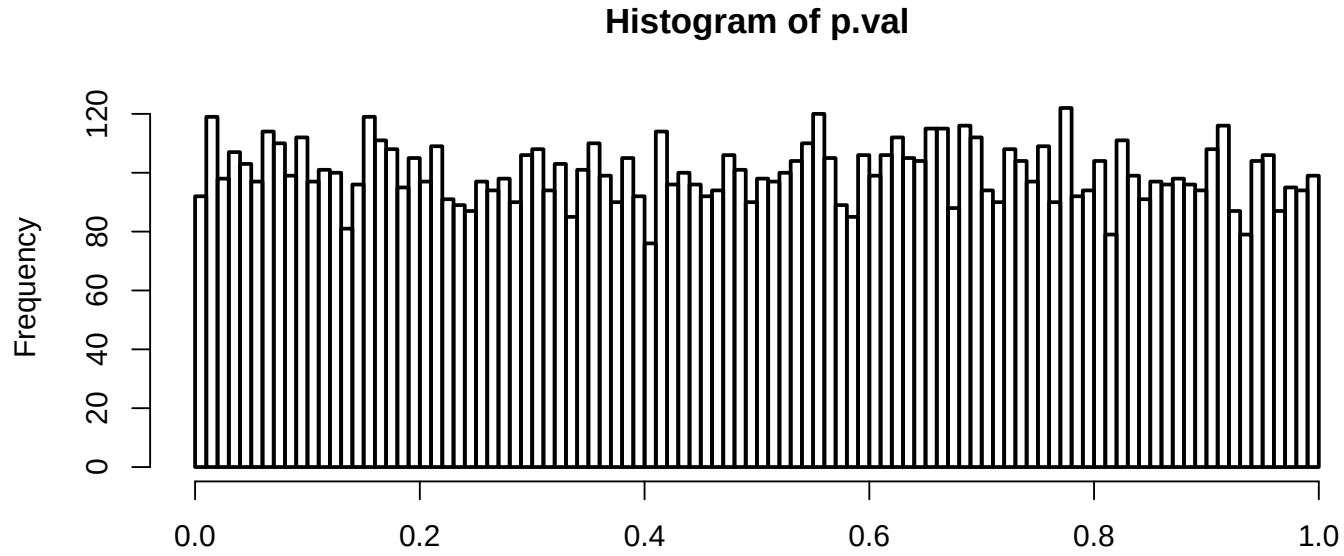

The t Distribution

```
> t.stat <- (bm - am)/sqrt(sp2)/sqrt(1/an + 1/bn)
> hist(t.stat, breaks = 100, xlab = "")
```



P-values are Uniformly Distributed

```
> p.val <- sapply(t.stat, function(tv, df) {
+     2 * (1 - pt(abs(tv), df))
+ }, an + bn - 2)
> hist(p.val, breaks = 100, xlab = "")
```



How "Significant" is Nothing?

Are we finding anything other than what we expect?

```
> sum(p.val < 0.05) # observed

[1] 519

> 0.05 * n.genes    # expected

[1] 500
```


Estimating Nothing

If there are no real differences, and if we can treat different genes as though they are "replicates" of the same experiment, then

1. Number of genes with $p < \alpha$ is approximately αN.

2. The distribution of p-values is uniform.

Statistical Error Types

Statisticians (on average) are obsessed with errors. They also use circumlocutions that make it difficult for non-statisticians to understand them. E.g., "rejecting the null hypothesis" means "calling a gene differentially expressed".

Test Result	**Truly Different**	**Truly Unchanged**
Positive	True Positive (TP)	False Positive (FP) Type I Error
Negative	False Negative (FN) Type II Error	True Negative (TN)

P-value = Prob(Type I Error)

To control Type II Errors (FN), you have to increase the sample size to ensure enough power to detect the true differences.

The Family-Wise Error Rate (FWER)

$FWER =$ probability of getting ≥ 1 FP when performing many statistical tests $=$ probability of making ≥ 1 mistake.

Bonferroni adjustment: To achieve $FWER \leq \alpha$ when looking at G genes, use $p \leq \alpha/G$ on a per-gene basis.

```
> bonferroni <- 0.05/n.genes
> bonferroni

[1] 5e-06

> sum(p.val < bonferroni)

[1] 0
```


What Happens with Real Data?

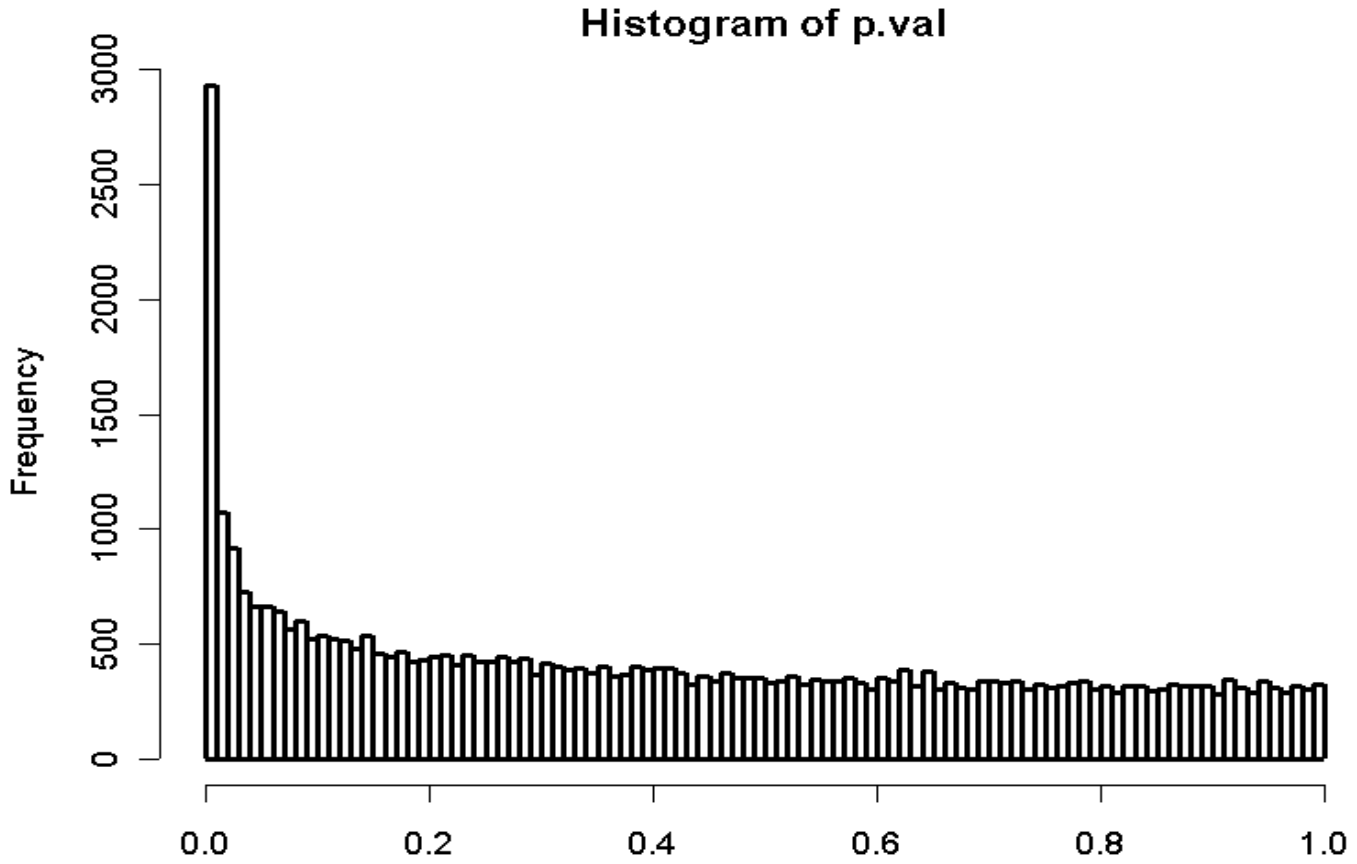
Reference: Lapointe et al. Gene expression profiling identifies clinically relevant subtypes of prostate cancer. *Proc Natl Acad Sci USA*. 2004; **101**: 811–816.

Two-color arrays processed with local background subtraction, loess normalization, and taking log ratios to a reference channel.

- 41 samples of apparently normal prostate
- 62 samples of prostate cancer
- 9 samples of lymph node metastases from prostate cancer

We randomly selected ten samples of normal prostate and ten samples of prostate cancer, and performed two-sample t-tests.

Real p-values



There seems to be an overabundance of small p-values, causing the distribution to differ considerably from uniform.

Counting Small

```
> n.genes <- nrow(data)
> n.genes
[1] 42129
> sum(p.val < 0.05)  # observed
[1] 6316    (2931 if p < 0.01)
> 0.05 * n.genes     # expected
[1] 2106.45 (421.29 if p < 0.01)
> bonferroni <- 0.05/n.genes
> bonferroni
[1] 1.186831e-06
> sum(p.val < bonferroni)
[1] 42
```

Simulating Something

We also simulated two data sets with differences:

1. Data Set I
   - 10 arrays per group, 2000 genes per array
   - Gene expressions in each group are independent, $N(\mu, 1)$.
   - In group A, take all $\mu_A = 0$.
   - 50 genes are different, with $|\mu_A - \mu_B| \sim 5 * Beta(2, 8)$.
2. Data Set II
   - 10 arrays per group, $10,000$ genes per array
   - Mean expression $\mu_A \sim Exp(1/20)$.
   - 100 genes are different, with $\mu_A/\mu_B \sim 1 + 9 * Beta(3, 7)$.

Bonferroni Correction: Results

- Data Set I (normal model)
  - Truth: 50 genes differ out of 2000
  - With $\alpha = 0.05$, makes 21 positive calls, 21 correct.
- Data Set II (exponential + noise)
  - Truth: 100 genes differ out of $10,000$
  - With $\alpha = 0.05$, makes 25 positive calls, 25 correct.

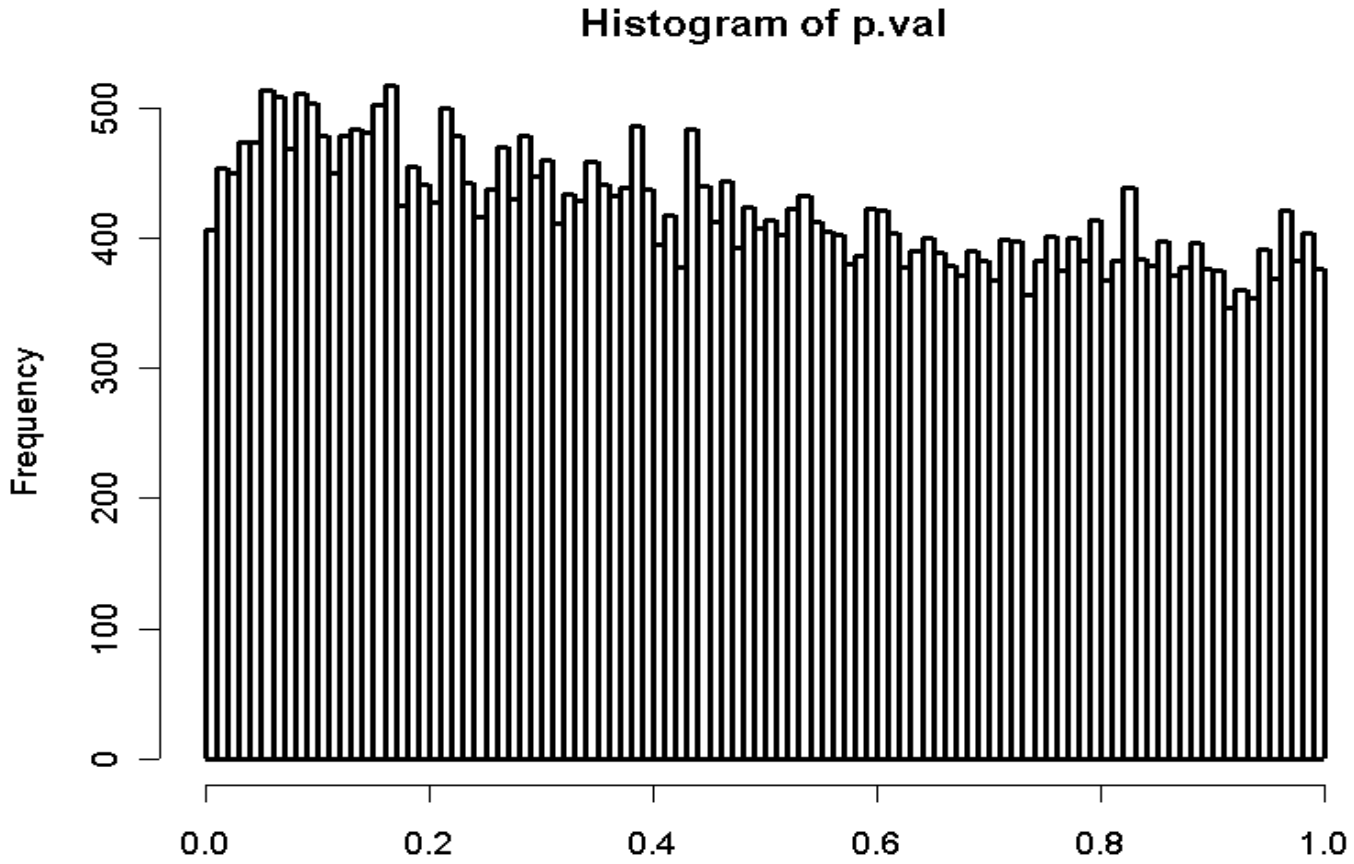
Beginning to assess the model

A key assumption of the Bonferroni approach is that a uniform distribution adequately describes the p-values when there are no differentially expressed genes present.

We can test how good this model is by performing a permutation test. In this case, we simply scramble the labels on the samples.

In the prostate example, we have ten normal and ten cancer samples. We choose ten samples at random to call "normal", and call the other ten "cancer", and we repeat the analysis with the two-sample t-test.

P-values with Scrambled Sample Labels



Nearly uniform, with a slight bulge near $p = 0.01$. This might be attributable to an imbalance of "truth" in the permuted groups.

Scrambled Data is Insignificant

```
> sum(p.val < 0.05)  # observed
[1] 2257
> 0.05 * n.genes     # expected
[1] 2106.45
> sum(p.val < 0.01)  # observed
[1] 406
> 0.01 * n.genes     # expected
[1] 421.29
> sum(p.val < bonferroni)
[1] 0
```

Should We Believe the p-values?

There is another potential difficulty with using the Bonferroni approach: in order to get a significant gene, we need extremely small p-values. That means we have to very accurately estimate the tails of the distribution, which is a fairly difficult thing to do unless one of two fairly unlikely things happens:

1. The number of samples is extremely large, or

2. The distribution of expression values is almost perfectly described by a normal distribution.

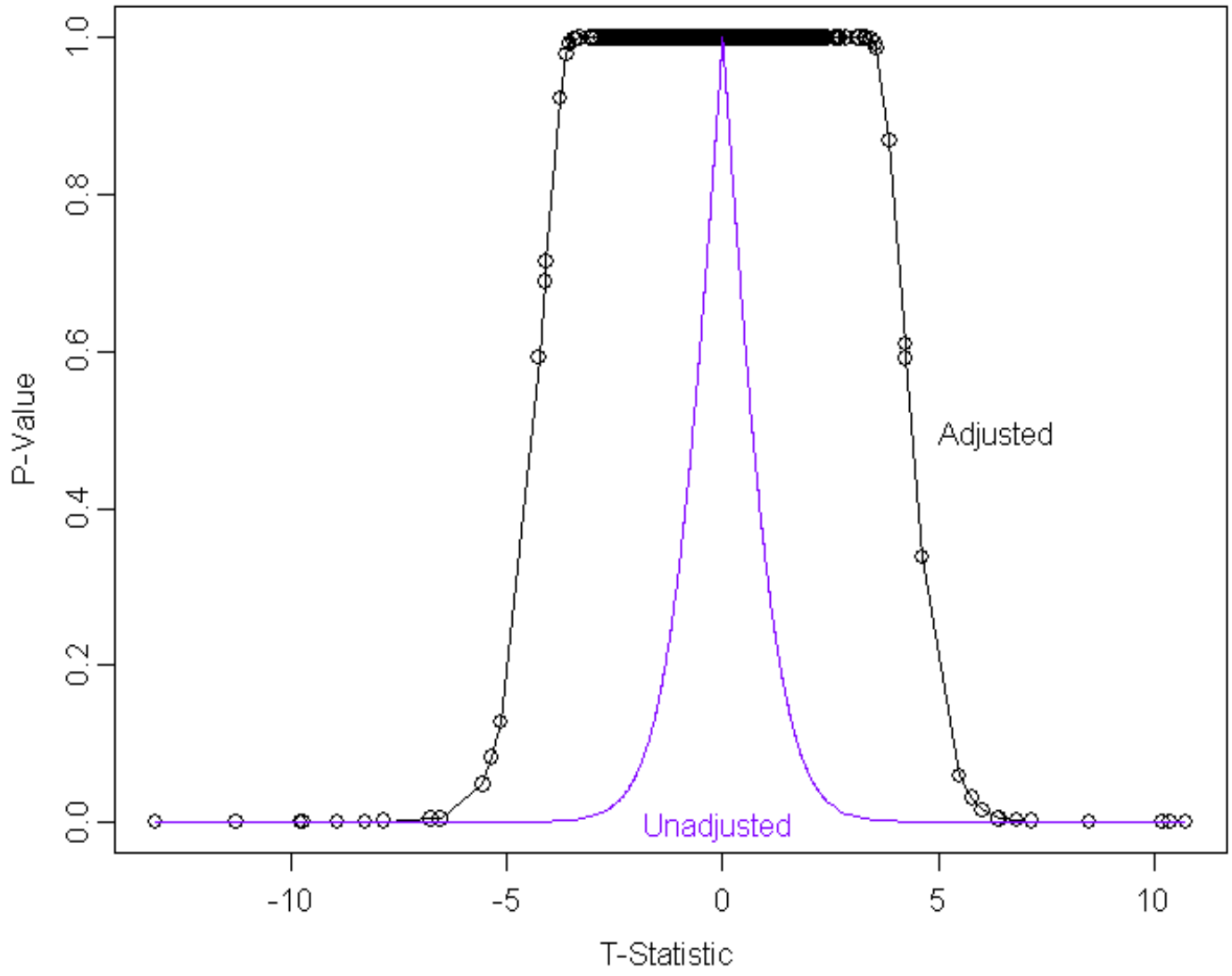
We can use permutations to get around the second problem, but that only makes the first problem worse.

Dudoit's Permutation p-values

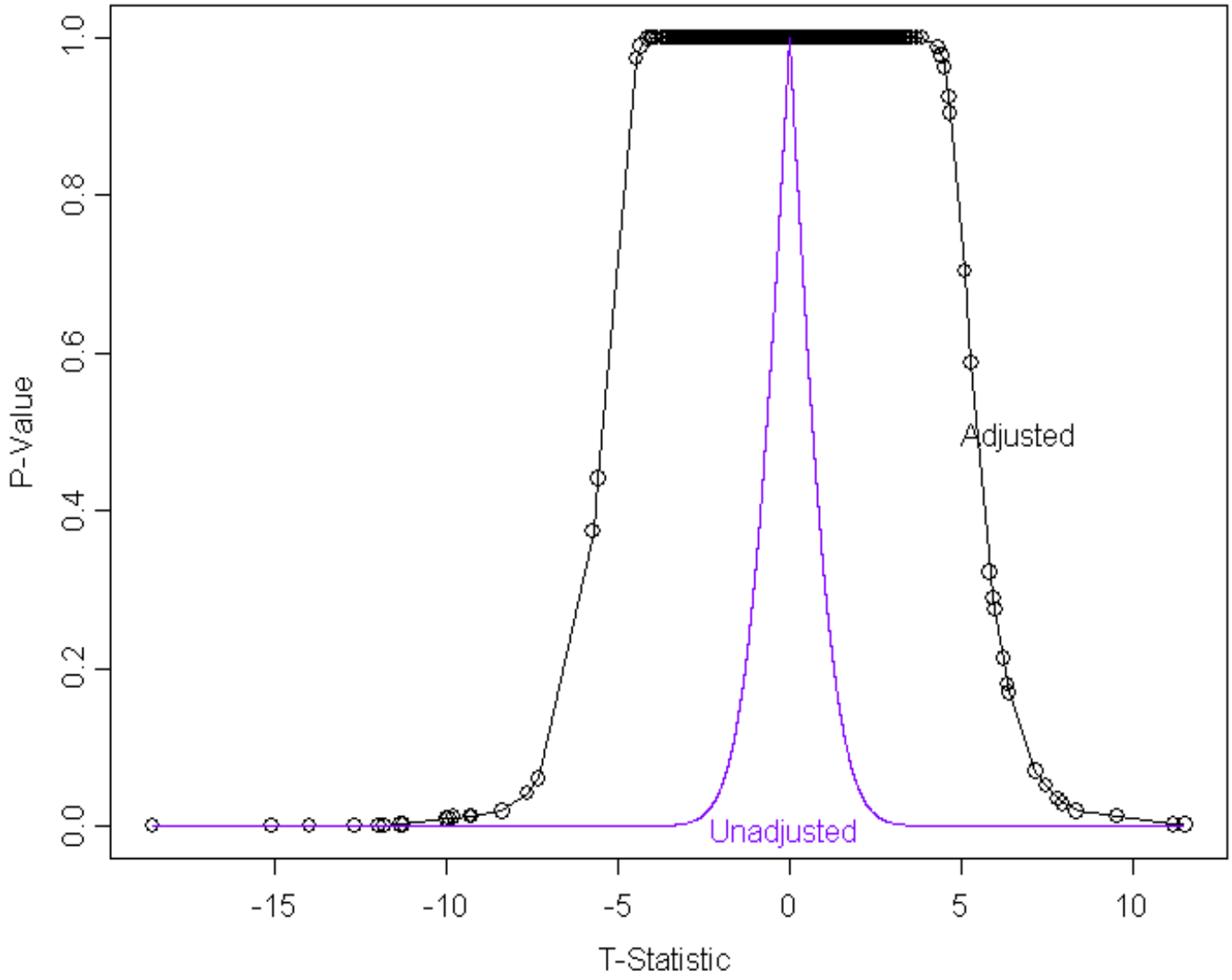
Reference: Dudoit et al. Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. *Statistica Sinica*, 2004; **12**: 111-139.

- Perform t-test for each gene g. Sort the absolute t-statistics, $|t_g|$.
- Repeat many times:
  - Randomly permute sample labels.
  - Compute new t-statistics
- Adjust p-values based on empirical joint distribution of t-statistics to control $FWER$.

Adjusted p-values, Data Set I



Adjusted p-values, Data Set II



Dudoit's Method: Results

- Data Set I (normal model)
  - Truth: 50 genes differ, out of 2000.
  - With $\alpha = 0.05$, makes 21 positive calls, 21 correct.
- Data Set II (exponential + noise)
  - Truth: 100 genes differ, out of $10,000$
  - With $\alpha = 0.05$, makes 21 positive calls, 21 correct.

Is FWER too conservative?

1. In the prostate data set, Bonferroni with $FWER \leq 5\%$ flagged 42 genes.

2. With an uncorrected $p \leq 1\%$, the model underlying the Bonferroni correction predicts only 421 genes, but we actually observe 2931.

3. With an uncorrected $p \leq 5\%$, the model underlying the Bonferroni correction predicts only 2106 genes, but we actually observe 6316.

Are there only 42 differentially expressed genes among the 42129 spots on this array, or are there $2510 = 2931 - 421$? Or maybe even $4210 = 6316 - 2106$?

Opportunity Cost

The Bonferroni correction only considers Type I Errors. Microarray experiments, however, are often used for discovery. Findings are usually confirmed by performing additional experiments (typically, real-time PCR). In some cases, the "opportunity cost" of missing out on a discovery (by making a Type II Error) is greater than the "validation cost" of finding some false positives (Type I Errors) in your list of genes.

Like anything else, there are trade-offs. By choosing a smaller significance cutoff for the p-values, you get fewer false positives but more false negatives. By choosing a larger cutoff, you get more false positives and fewer false negatives.

The False Discovery Rate (FDR)

$FDR = FP/(TP + FP) =$ fraction of false positives among all genes called differentially expressed by the test. Here is a crude way to estimate FDR: Assume the uniform p-value model holds under the null. Then the expected number of false discoveries at a given cutoff is pG. If the total number of discoveries is V, then we can estimate $FDR = pG/V$. In the prostate example, this gives

- When $p = 0.05$, $FDR = 2106/6316 = 0.3334$.
- When $p = 0.01$, $FDR = 421/2931 = 0.1436$.

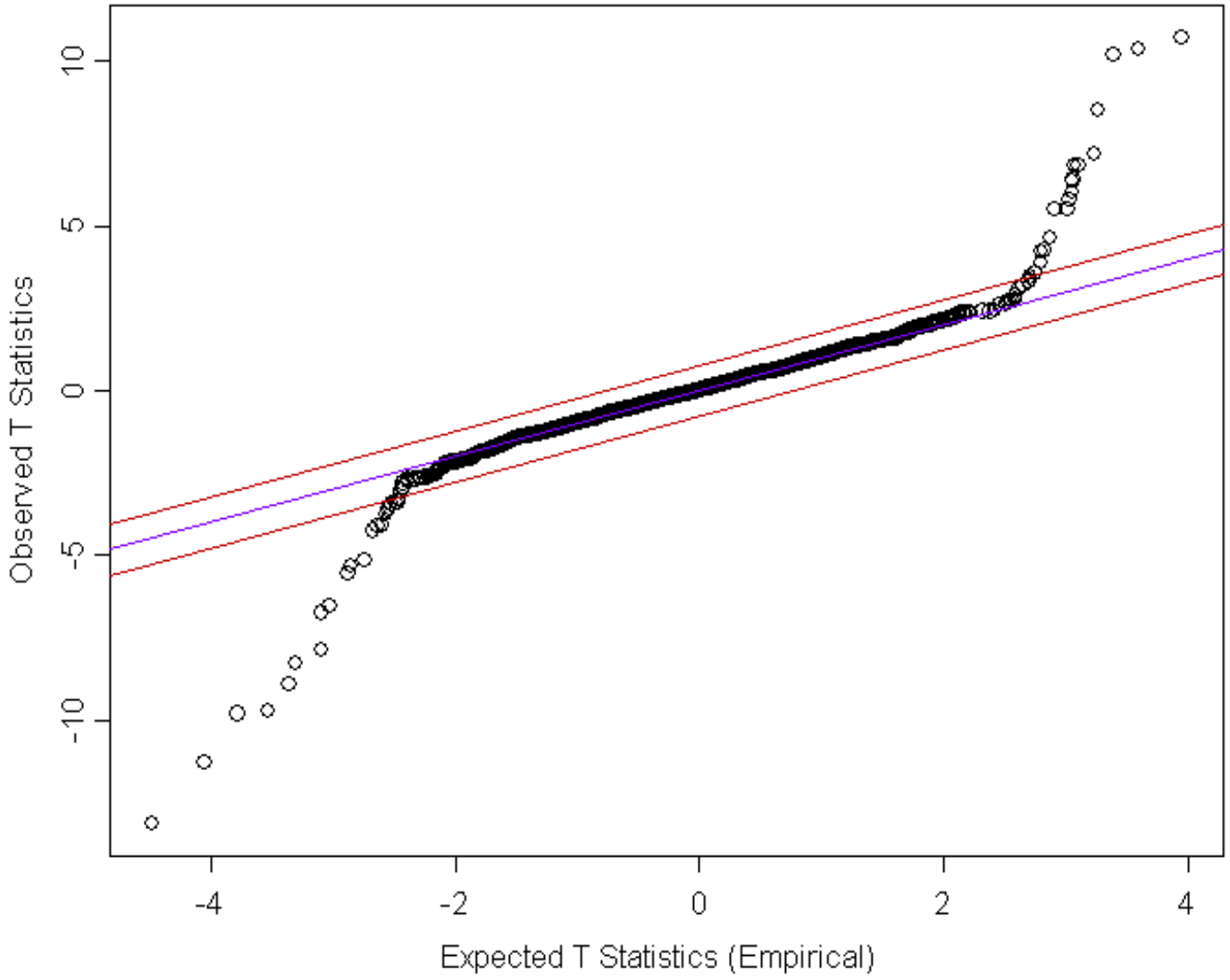
This estimate isn't very good. It overestimates the number of errors by not allowing for true discoveries.

Significance Analysis of Microarrays (SAM)

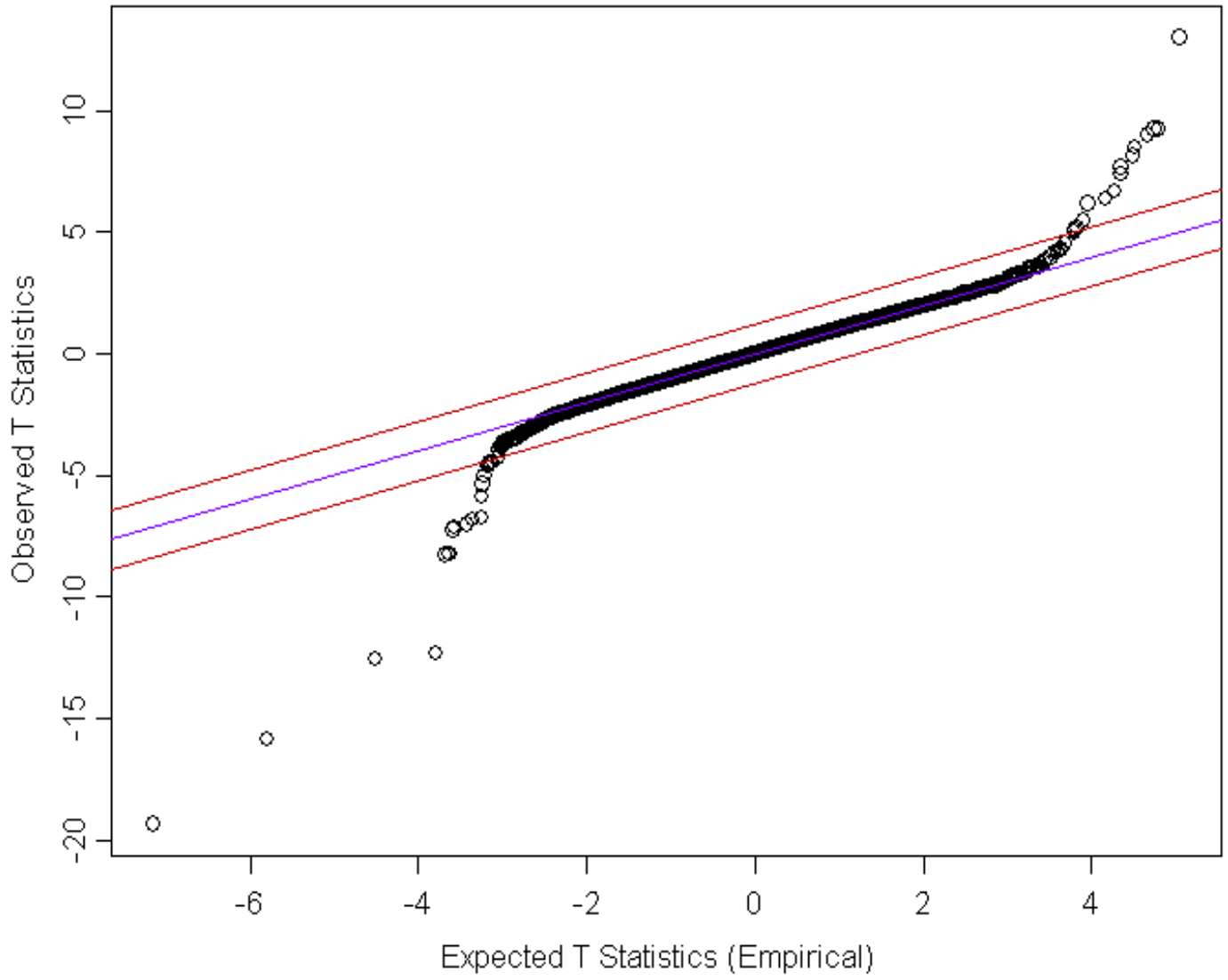
Reference: Tusher et al. Significance analysis of microarrays applied to the ionizing radiation response. *PNAS*, 2001; **98**: 5116–5121.

- Compute modified t-statistics (increase σ to minimize coefficient of variation across the array).
- Recompute t-statistics based on balanced permutations (each group equally represented) of the sample labels.
- Decide on significance cutoff based on quantile-quantile plot of observed versus expected t-statistics.
- Estimate FDR from the permutations.

SAM, Data Set I



SAM, Data Set II



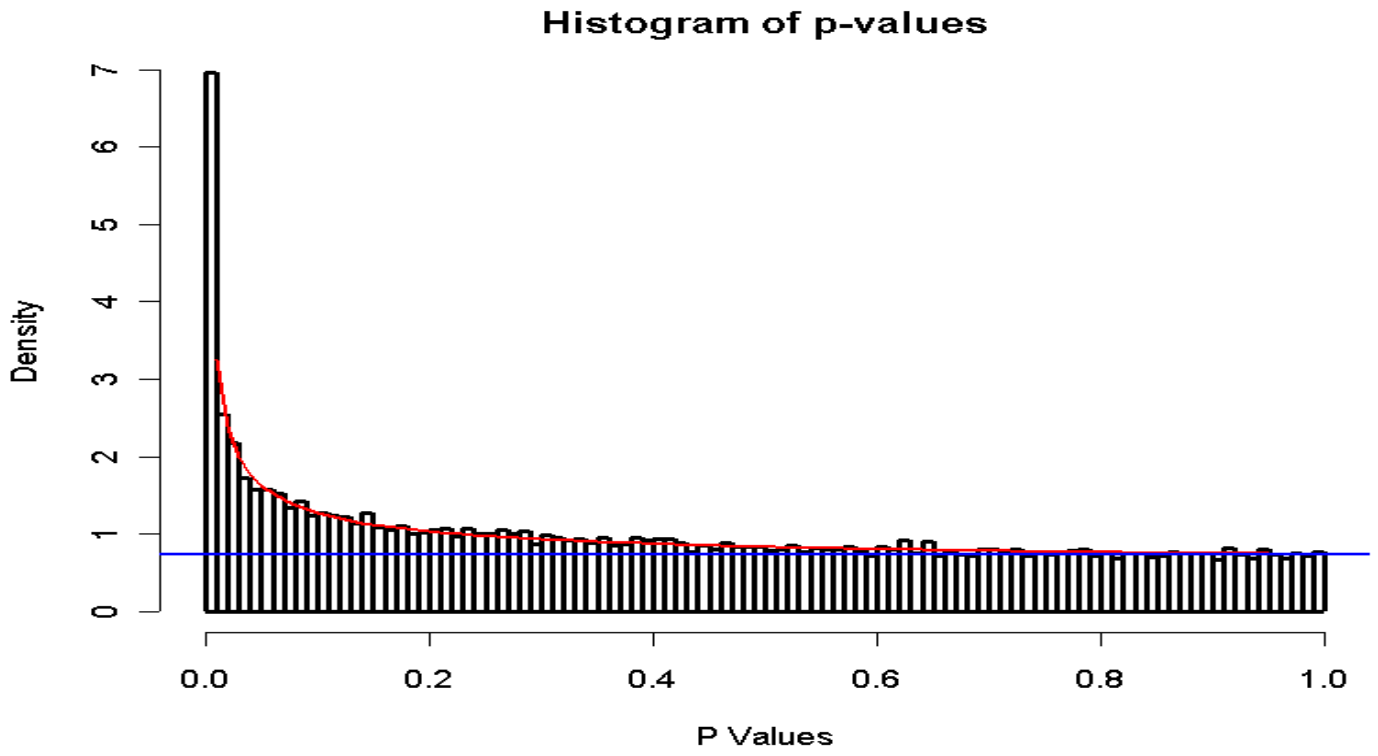
SAM: Results

- Data Set I (normal model)
  - Truth: 50 genes differ, out of 2000.
  - With $FDR = 0.10$, makes 32 positive calls, 30 correct.
- Data Set II (exponential + noise)
  - Truth: 100 genes differ, out of $10,000$
  - With $FDR = 0.10$, makes 41 positive calls, 37 correct.

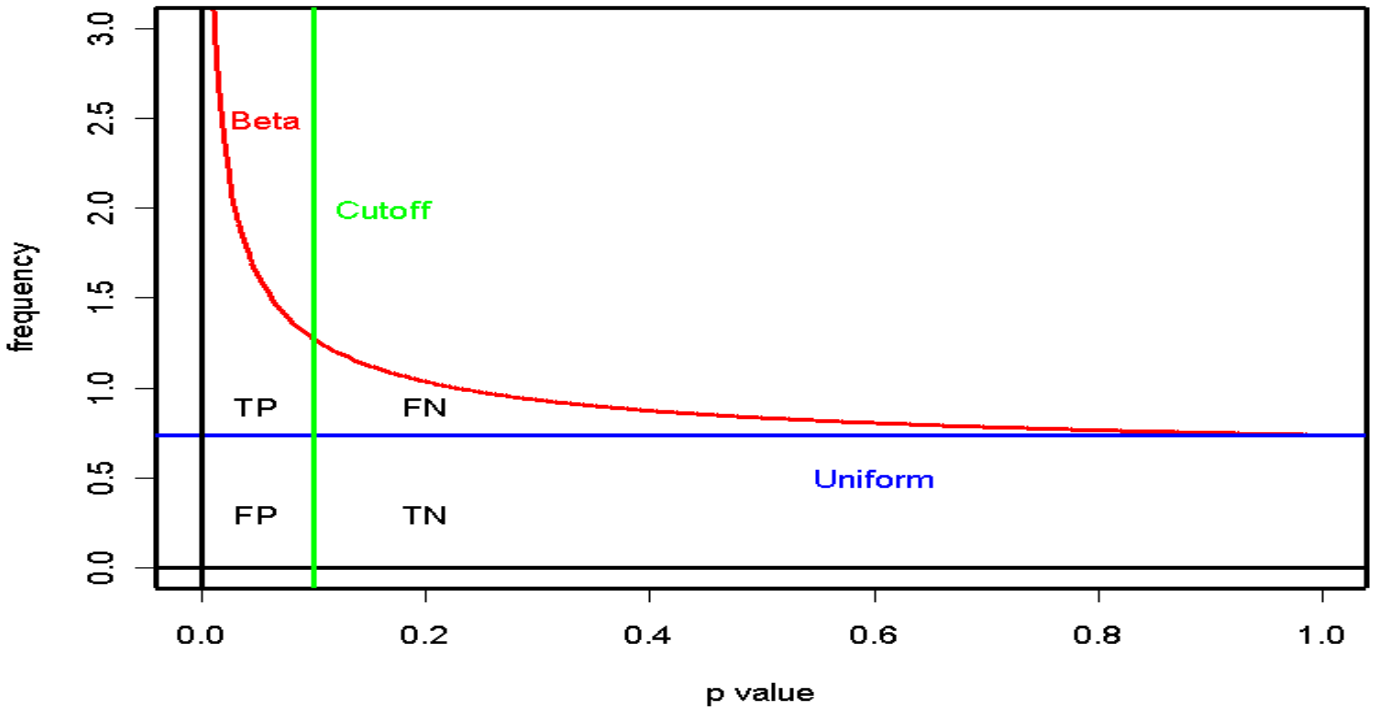
Detects more true positives in simulated data than Bonferroni or Dudoit, at some cost in false positives. Like Dudoit's method, it is computationally intensive.

A Beta-Uniform Mixture Model (BUM)

Reference: Pounds and Morris. Estimating the occurrence of false positives and false negatives in microarray studies by approximating and partitioning the empirical distribution of p-values. *Bioinformatics*, 2003; **19**: 1236–1242.

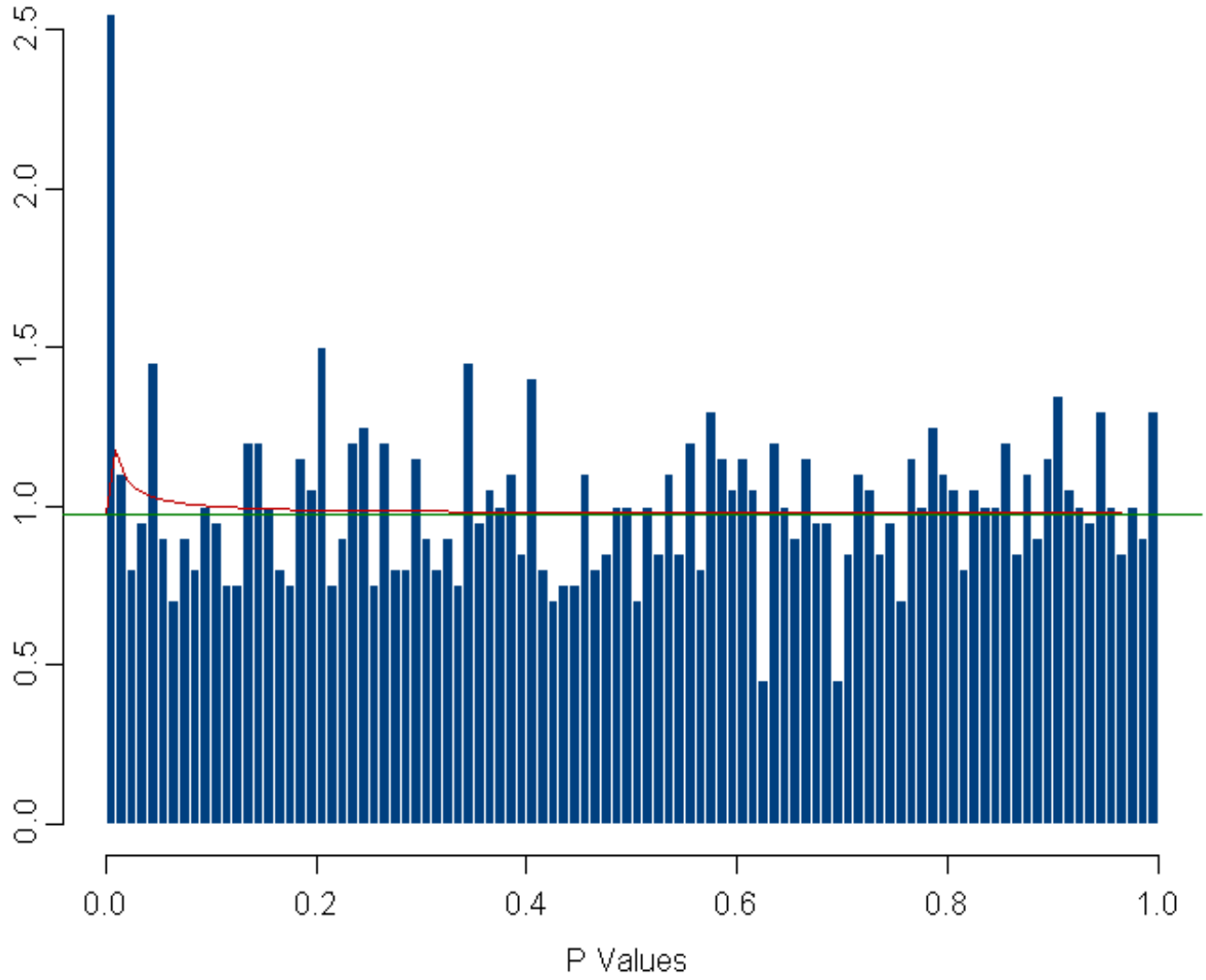


The BUM Theory

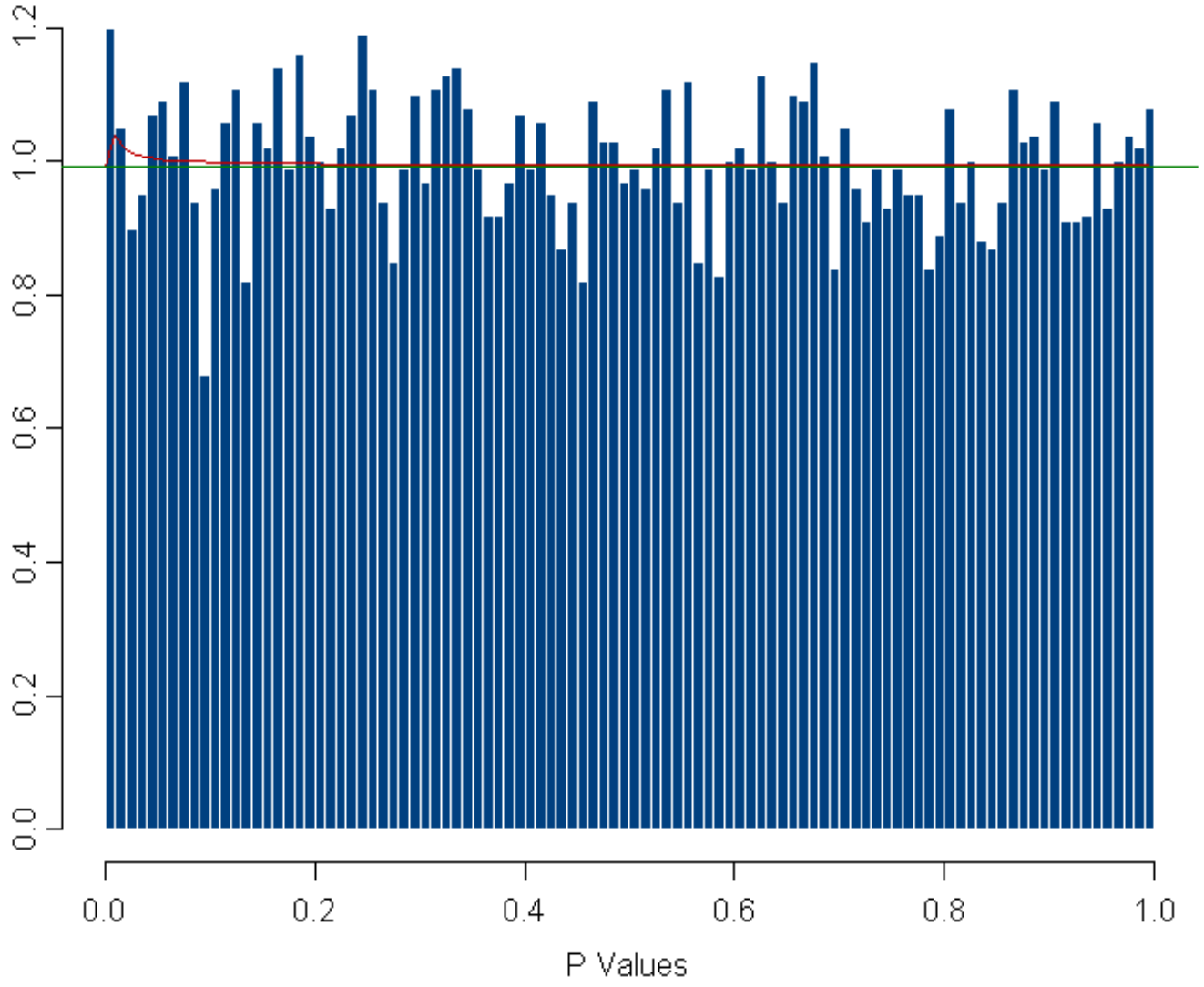


Idea: Model the p-values as a mixture of a uniform distribution and a beta distribution. Estimate mixture parameters. Obtain estimates of TP, FP, FN, TN as a function of significance cutoff.

BUM, Data Set I



BUM, Data Set II



BUM: Results

- Data Set I (normal model)
  - Truth: 50 genes differ, out of 2000.
  - With $FDR = 0.10$, makes 33 positive calls, 31 correct.
  - Estimates that 2.8% of genes are different (truth $= 2.5\%$)
- Data Set II (exponential + noise)
  - Truth: 100 genes differ, out of $10,000$
  - With $FDR = 0.10$, makes 40 positive calls, 37 correct.
  - Estimates that 0.7% of genes are different (truth $= 1.0\%$)

Results equivalent to SAM, with much less computation.

BUM Results on Prostate Data

We have already seen the histogram, and the fit of the beta-uniform mixture.

- With $FDR < 0.01$, calls 427 genes differentially expressed.
- With $FDR < 0.05$, calls 1513 genes differentially expressed.
- With $FDR < 0.10$, calls 2727 genes differentially expressed.

Overall, BUM estimates that 26% of the genes are differentially expressed at some level. (That's more than $10,000$ genes!)

Empirical Bayes

There are other ways to exploit the fact that we have thousands of tests in order to improve our estimates of what's going on. Here, we encounter a synthesis of two views of statistics: frequentist and Bayesian.

Reference: Efron and Tibshirani. Empirical Bayes methods and false discovery rates for microarrays. *Genetic Epidemiology*, 2002; **23**: 70–86.

Basic Idea

Assume that there are two classes of genes, Different and Not Different. We assume prior probabilities

- $p_0 =$ Prob(Not Different)
- $p_1 = 1 - p_0 =$ Prob(Different)

and density functions

- $f_0(y)$, known, if Not Different (Null)
- $f_1(y)$, unknown, if Different

Mixtures

The overall probability density function is a mixture

$$f(y) = p_0 f_0(y) + p_1 f_1(y).$$

Bayes' Theorem: $P(H|D) = P(D|H)P(H)/P(D)$

Applying Bayes' Theorem gives posterior estimates:

$$p_0(y) \equiv Prob(\text{Not Diff}|Y = y) = p_0 f_0(y)/f(y)$$

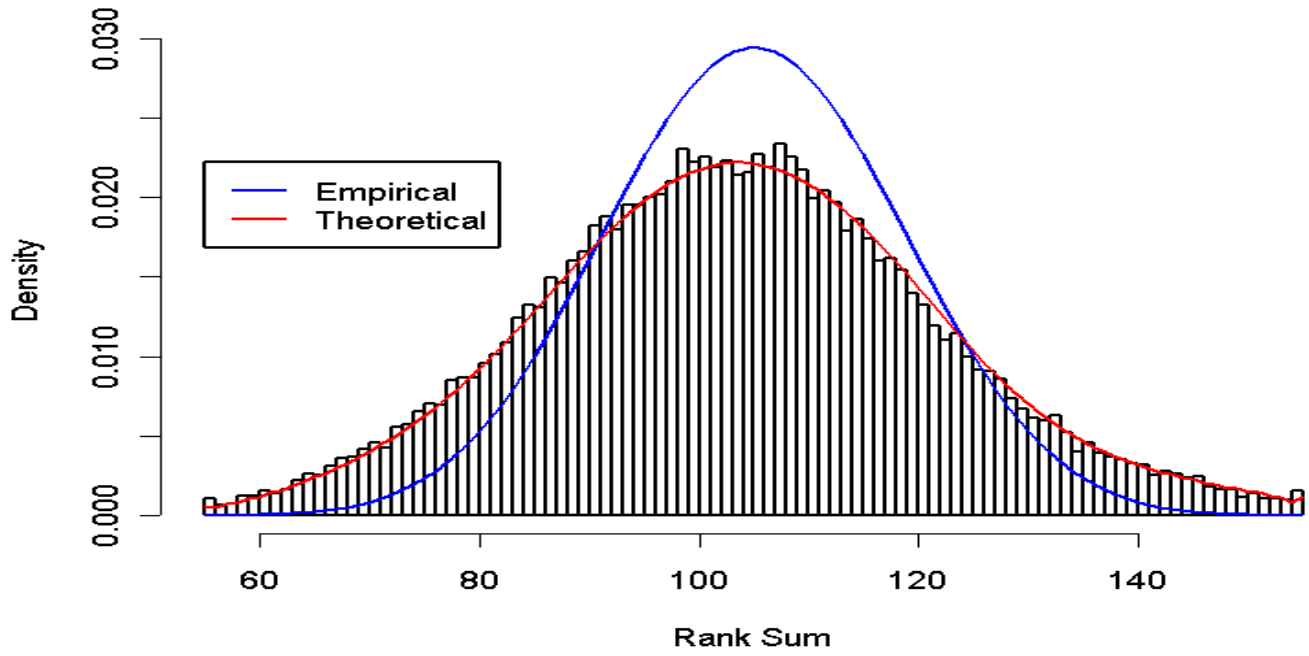
and

$$p_1(y) \equiv Prob(\text{Diff}|Y = y) = 1 - p_0 f_0(y)/f(y)$$

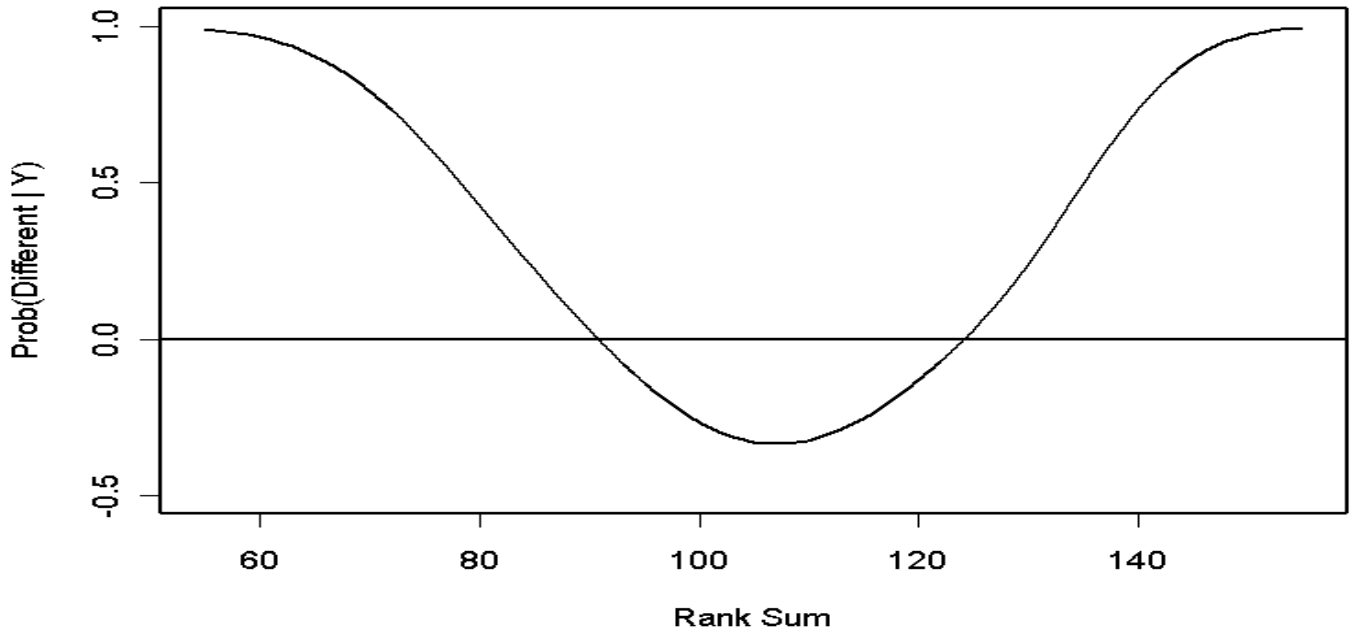
We can use the observed data to estimate the overall density function by $\hat{f}(y)$ (typically by log-transforming the observed function and fitting a curve.)

Empirical Bayes

The "empirical" nature of this Bayesian idea is that we can adjust the "prior" p_0 after looking at the data, and thus obtain some reasonable values for it. First, here is how well we fit the distribution (mentally swap the labels, since they're wrong):

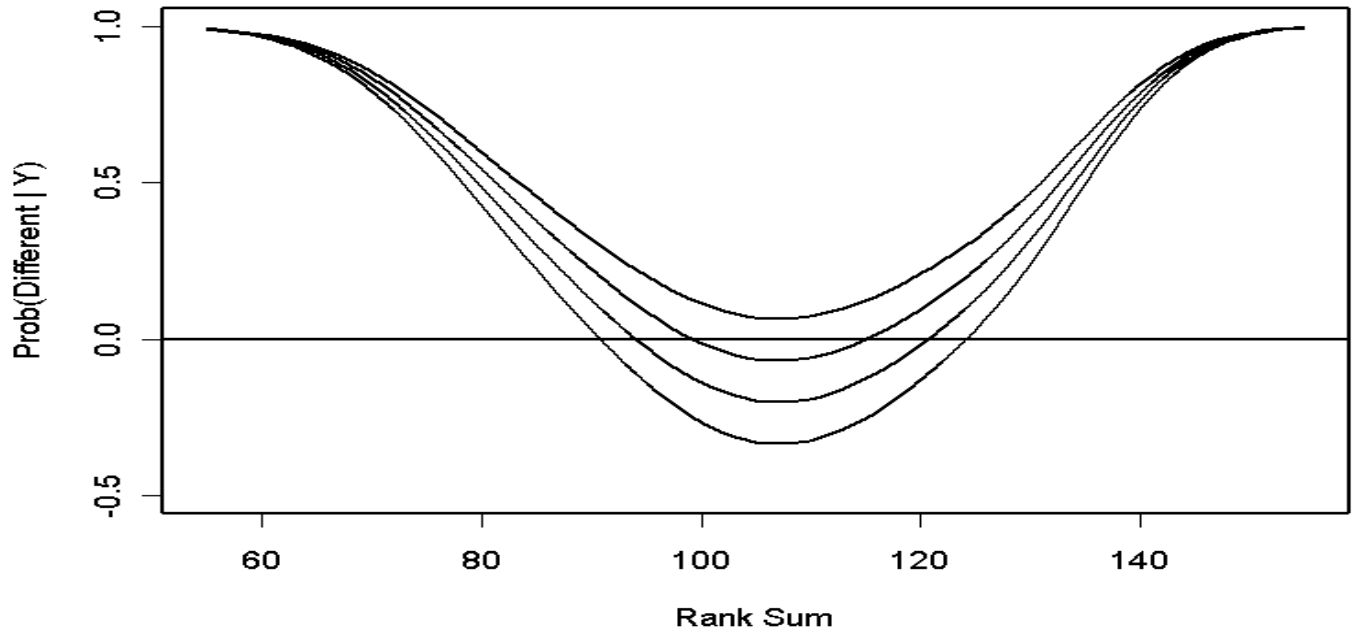


Plot of Posterior Probability of Difference



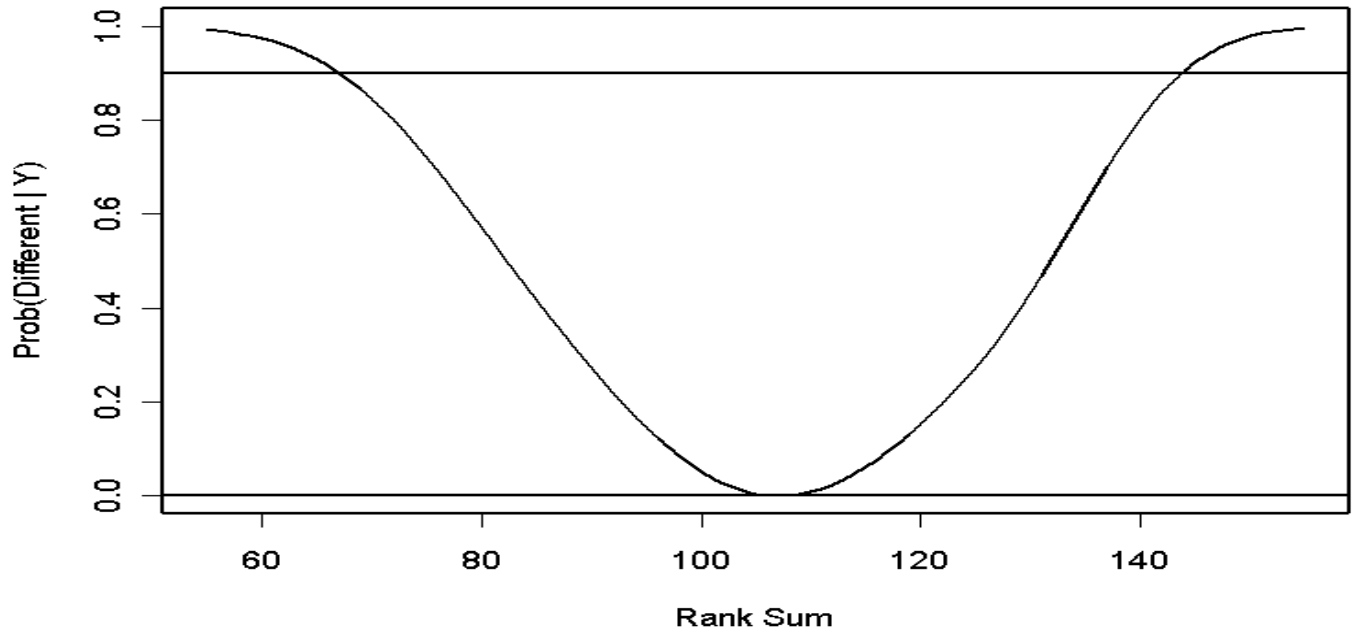
This graph assumes $p_0 = 1$, so no genes are different. The posterior probability of difference goes negative! This results from the "empirical" nature of the estimate without imposing a full model. We can, however, adjust p_0 to prevent negative probabilities.

Plot of Posterior Probability of Difference



This shows posterior probabilities with $p_0 = 0.7, 0.8, 0.9, 1.0$. Somewhere between $p0 = 0.7$ and $p_0 = 0.8$, all the posterior probabilities become positive.

Plot of Posterior Probability of Difference

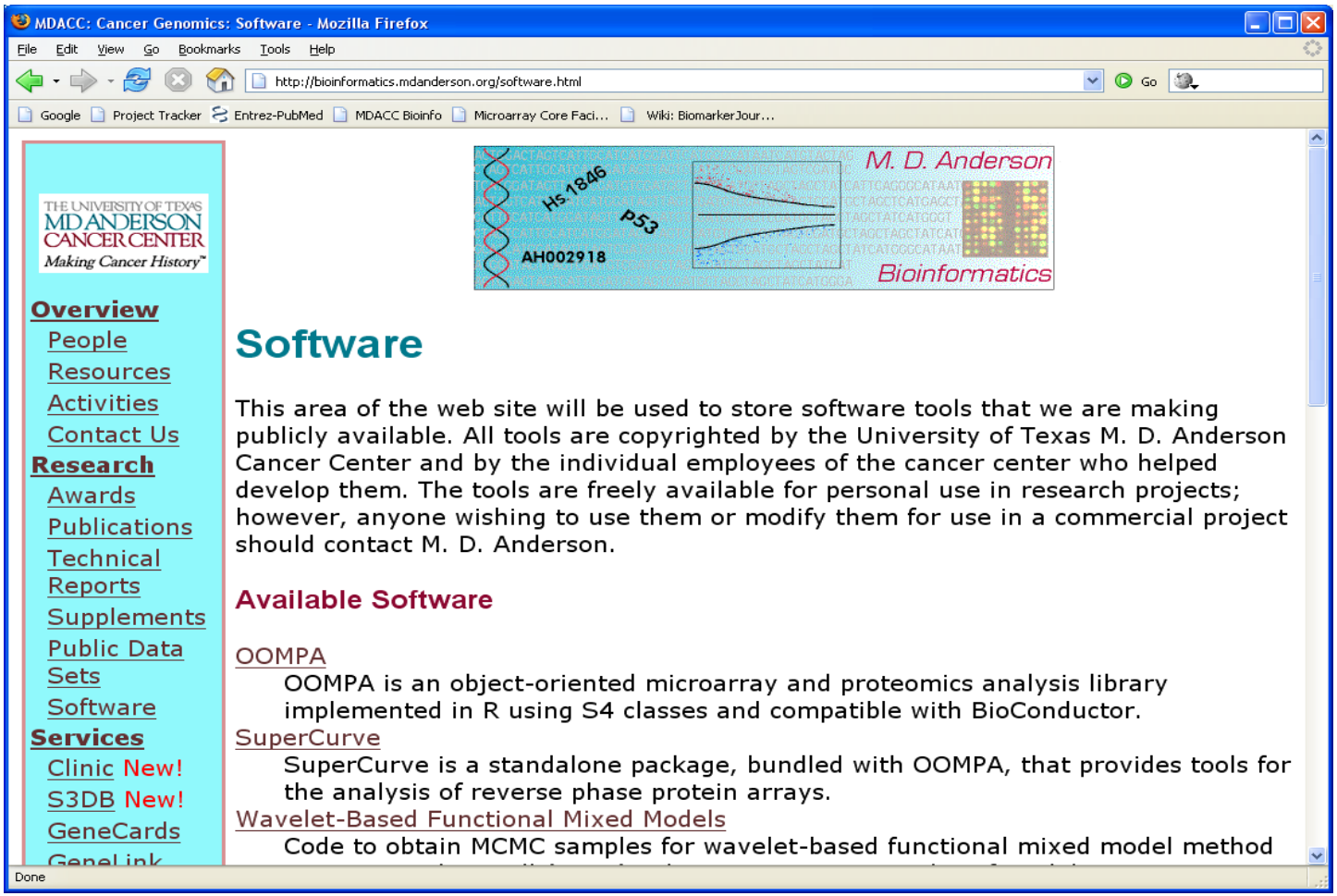


This plot uses $p_0 = 0.75$, which is essentially the largest value we can use for p_0 and ensure that all the posterior probabilities are positive. The horizontal line indicates a posterior probability of 90% that a gene is differentially expressed.

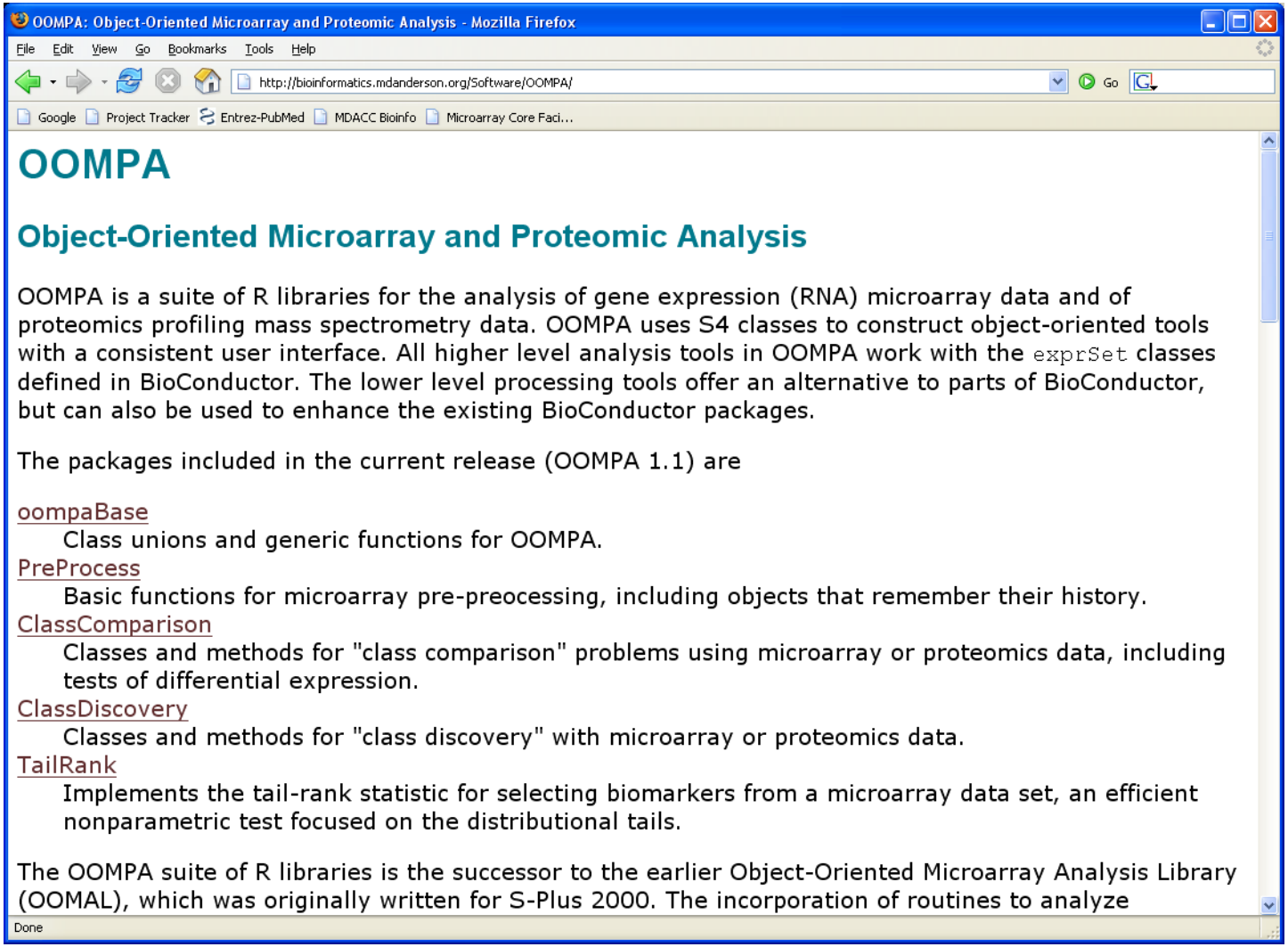
How does this work in R?

We have implemented this idea in an R package:

`http://bioinformatics.mdanderson.org/software.html`



The OOMPA home page



The ClassComparison package