R routines for partial mixture estimation and differential expression analysis

David Rossell *

1 Introduction

The following R routines are provided in the file `ebayes.l2e.r` (available at http://www.stat.rice.edu/ rusi).

- `ebayes.l2e`: core routine that performs 2 group differential expression analysis via partial mixture estimation. Additionally, it implements pairwise t and Wilcoxon tests with Benjamini & Hochberg's p-value adjustment.
- `qqnorm.pme`: creates qq-normal plot to assess the partial mixture assumption that the equally expressed genes are normally distributed.
- `ma.boxplot`: creates MA-plot to assess the assumption that the differences between group means are identically distributed.
- `find.fdr`: auxiliary routine that, given the posterior probability that each gene is differentially expressed, computes the Bayesian FDR (1).
- `find.threshold`: auxiliary routine that, given the posterior probability that each gene is differentially expressed, computes the optimal threshold to declare significance while controling the Bayesian FDR (2).
- `wupdc`: fits a partial Normal mixture component via weighted L_2E distance minimization.
- `rewupdc`: fits a partial Normal mixture component via iteratively reweighted L_2E distance minimization.

*Department of Biostatistics, The University of Texas M. D. Anderson Cancer Center, Houston, TX.

- `wupdc.findw`: fits a partial Normal or partial T component via weighted L_2E minimization, fixing the parameters of the component so that only its weight (*i.e.* the proportion of differentially expressed genes) needs to be estimated.

These routines only implement the weighted L_2E criterion for partial mixture estimation. For them to work correctly one should also download and source the routines that implement the non-weighted L_2E criterion. These can be found in the file `mpdc.r`, which is available at David W. Scott's page (http://www.stat.rice.edu/ scottdw/code/l2e).

2 Partial mixture estimation estimating the null distribution

Let's start by simulating expression data for 10,000 genes (we set the seed for the random number generator so that you can reproduce exactly the results). We define mean expression values in `a` ranging from 5 to 10, and we define differences in group means `m` to be a decreasing function of `a`. We then draw 2 observations for each group, with the variance also being a decreasing function of `a`. We have observed this sort of decrease in mean differences and variances in several real datasets, which is why we set up the simulation this way. Finally, we randomly set approximately 5% of the genes to be differentially expressed by adding 1/4 to the expression values in the first group.

```
> source("~/projects/l2e/1 R Routines/ebayes.l2e.R")
> source("~/projects/l2e/1 R Routines/mpdc.R")
> set.seed(1)
> n <- 10000
> a <- seq(5, 10, length = n)
> m <- 2/a - 0.2
> mu1 <- 0.5 * (a + m)
> mu2 <- 0.5 * (a - m)
> x <- matrix(rnorm(n * 2, mu1, 1/a), ncol = 2)
> y <- matrix(rnorm(n * 2, mu2, 1/a), ncol = 2)
> truede <- (runif(n) < 0.05)
> x[truede, ] <- x[truede, ] + 0.25
> groups <- rep(0:1, each = 2)
```

Before obtaining any results, we should check whether the partial mixture assumption that the difference between group means are identically dis-

tributed is reasonable (in fact we know it not to hold because of the way we simulated the data). Figure 1(a) is generated with the following code using the function `ma.boxplot`.

```
> m <- ma.boxplot(cbind(x, y), groups, plot = TRUE, xlab = "Average expression
+     ylab = "Difference in means (M)")
> abline(h = 0)
```

The overall mean expression A=`rowMeans(x)+rowMeans(y)` are categorized into 50 groups, and for each group a boxplot of the differences between groups M=`rowMeans(x)-rowMeans(y)` is presented. We observe that both the mean and the variance of M depends on the value of A, therefore indicating a violation of the identically distributed assumption. A simple-minded way to fix this is by centering and scaling each M value according to its estimated mean and standard deviation. The routine `ma.boxplot` accomplishes this by obtaining mean and variance estimates for each boxplot separately.

```
> m.norm <- ma.boxplot(cbind(x, y), groups, centerx = TRUE, scalex = TRUE,
+     plot = TRUE, xlab = "Average expression (A)", ylab = "Difference in means
> abline(h = 0)
```

The argument `centerx=TRUE` indicates to center the data and `scalex=FALSE` indicates to divide by the estimated standard deviation. The normalized values are saved in `m.norm`. The resulting plot is presented in 1(b).

At this point we are ready to fit a partial mixture to the values in `m.norm`, and from it obtain a list of differentially expressed genes. We use the routine `ebayes.l2e`. The arguments `wl2e=TRUE` and `wl2e.adjP=TRUE` indicate that the weighted L_2E criterion should be used (as opposed to non-weighted L_2E), and that two lists of genes should be obtained: one controlling for the Bayesian and the other the frequentist FDR. By default the routines keeps the FDR below 0.05.

```
> ebayes.fit <- ebayes.l2e(x = m.norm, wl2e = TRUE, wl2e.adjP = TRUE)
> ebayes.fit$w0.wl2e

[1] 0.9883232

> table(ebayes.fit$rej.wl2e, truede)

   truede
    FALSE TRUE
  0  9465  528
  1     0    7
```

(a) (b)

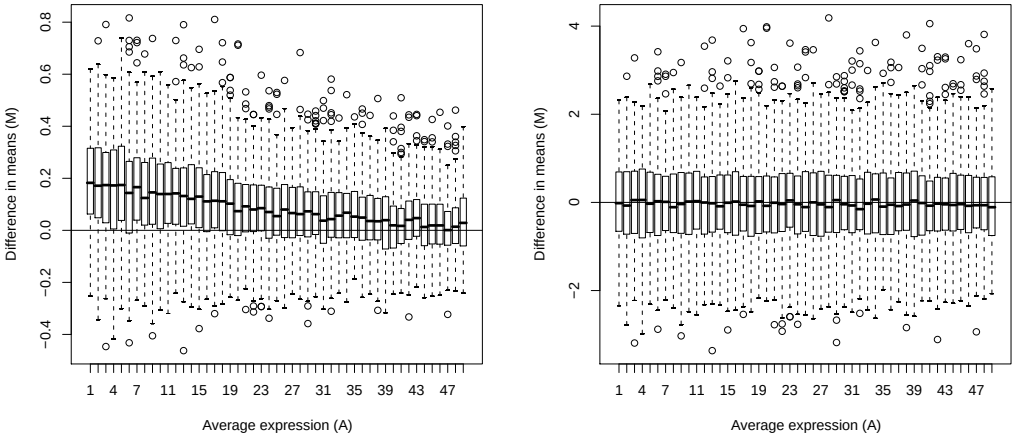

Figure 1: Assessing identically distributed assumption

```
> table(ebayes.fit$rej.wl2e.adjP, truede)

   truede
    FALSE TRUE
  0  9465  535
```

The weighted L_2E fit estimates 98.8% of the genes to be equally expressed, while in reality this proportion is slightly lower than 95%. When we compare the obtained gene lists with the real differential expression status (saved in `truede`) we see that the pseudo-Bayesian procedure finds 7 genes (all of which are truly differentially expressed), while the frequentist procedure does not find any genes for this particular dataset.

When comparing these results with t-tests with Benjamini & Hochberg p-value adjustment, we find that this approach does not find any genes either. We obtain the same results when using the Significance Analysis of Microarrays as implemented in the library `siggenes`. SAM estimates the proportion of differentially expressed genes to be 87.8%. The fact that we find few or not genes with all these methods should not be surprising, considering that we only have two observations per group and that the amount of differential expression is relatively small.

```
> library(genefilter)
> z.ttest <- rowttests(cbind(x, y), as.factor(groups))
```

```
> rej.ttest <- (p.adjust(z.ttest$p.value, method = "BH") < 0.05)
> table(rej.ttest, truede)

         truede
rej.ttest FALSE TRUE
    FALSE  9465  535

> library(siggenes)
> sam.x <- sam(cbind(x, y), groups, q.version = 2)

We're doing 6 complete permutations

> rej.sam <- -(sam.x@q.value < 0.05) * sign(sam.x@d)
> table(abs(rej.sam), truede)

   truede
    FALSE TRUE
  0  9465  535

> sam.x@p0

[1] 0.878399
```

Finally, we assess the normality assumption in our partial mixture fit by obtaining a qq-normal plot of the normalized values `m.norm`.

```
> qqnorm.pme(m.norm)
```

The resulting plot is shown in Figure 2. The horizontal lines indicate the interval where 98.8% of the data falls, that is the partial mixture does not assume normality for the observations outside of this interval.

3 Partial mixture estimation fixing the null distribution

We now analyze the data from Section 2 by computing a moderated t-test statistic and assuming that its null distribution follows a Student's t distribution. That is, only the proportion of differentially expressed genes will be estimated.

First, we compute the moderated t-test statistic using the functions `lmFit` and `eBayes` from library `limma` (3; 4). We then find that the estimated degrees of freedom for the moderated t-test statistic are 18.39 (note that the classical t-test that assumes equal variances has 2 degrees of freedom).

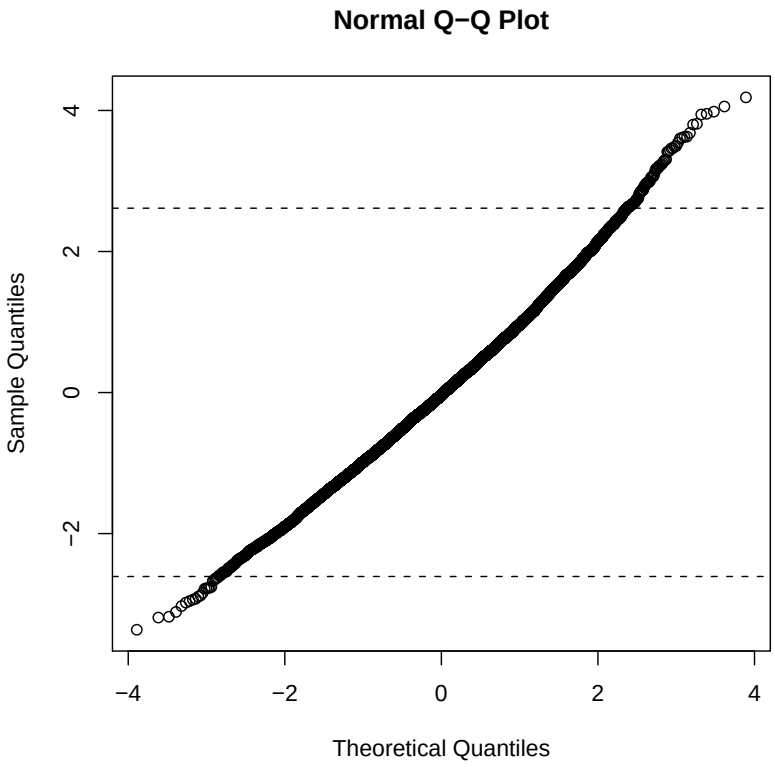

Figure 2: Assessing normality assumption

```
> library(limma)
> design <- cbind(Grp1 = 1, Grp2vs1 = groups)
> fit <- lmFit(cbind(x, y), design)
> eb <- eBayes(fit)
> nu <- mean(eb$df.residual + eb$df.prior)
> nu

[1] 18.39283
```

Second, we assess if it is reasonable to assume that the tests statistics are equally distributed for all genes, and also whether they follow a t distribution with `nu` degrees of freedom.

```
> a <- rowMeans(x) + rowMeans(y)
> acat <- cut(a, breaks = quantile(a, probs = seq(0, 1, length = 50)))
> boxplot(eb$t[, 2] ~ acat)
> abline(h = 0, lty = 2)
```

From Figure 3(a) we see that the equally distribution does not hold, so we use the function `ma.boxplot` to obtain Figure 3(b). This is still not perfectly identically distributed, since the variance of the test statistic seems to be larger for smaller values of `a`, but we will carry on the analysis nevertheless. Note that it would not be appropriate to scale the data (*i.e.* setting `scalex=TRUE`), since that would force the overall distribution of the test statistic to have unit standard deviation, and so we would expect to estimate that the proportion of equally expressed genes is 1. Intuitively, the key to our approach is that it assumes that part of the genes have test statistics with variance 1, which are interpreted to be equally expressed genes, but for the rest of the genes the variance should be greater than 1.

```
> tstat <- ma.boxplot(m = eb$t[, 2], a = a, centerx = TRUE, scalex = FALSE,
+     plot = TRUE)
> abline(h = 0, lty = 2)
```

We perform the partial mixture analysis with the function `ebayes.l2e`. This automatically loads the library `quantreg`, which is used to obtain a Cauchy kernel density estimate of the overall distribution of the test statistic.

```
> ebayes.fit <- ebayes.l2e(tstat, group, wl2e = T, fdr = 0.05,
+     nufix = nu)

Package SparseM (0.73) loaded.  To cite, see citation("SparseM")
```

(a) (b)

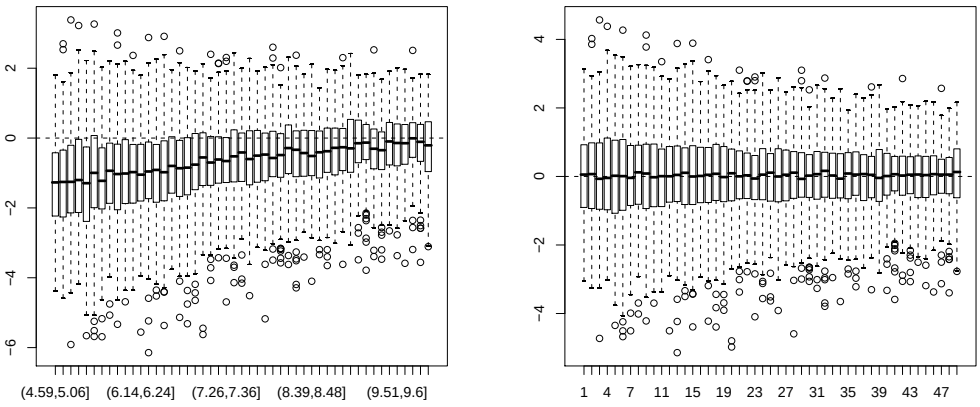

Figure 3: Identically distributed assumption for moderated t-test statistic

```
> ebayes.fit$w0.wl2e

[1] 0.9609105

> table(abs(ebayes.fit$rej.wl2e), truede)

   truede
    FALSE TRUE
  0  9462  527
  1     3    8
```

We estimate the proportion of equally expressed genes to be 96.1%, which is much closer to the true value 95% than the estimate we obtained in Section 2. We then check whether the genes declared to be differentially expressed are actually so. Out of the 11 genes found, 8 were actually differentially expressed and the other 3 were false positives. This is not surprising, since in Figure 3(b) we saw that the partial mixture assumptions are violated. Still, in practical terms it's probably not a bad performance, considering that the competing methods did not find any genes and that only two samples per group were available.

Finally, we check the assumption that the test statistic is t-distributed by means of a qq-plot (Figure 4(a)). The assumption seems reasonable. We obtain a plot of the estimated posterior probabilities as a function of the test statistic (Figure 4(b)).

(a) (b)

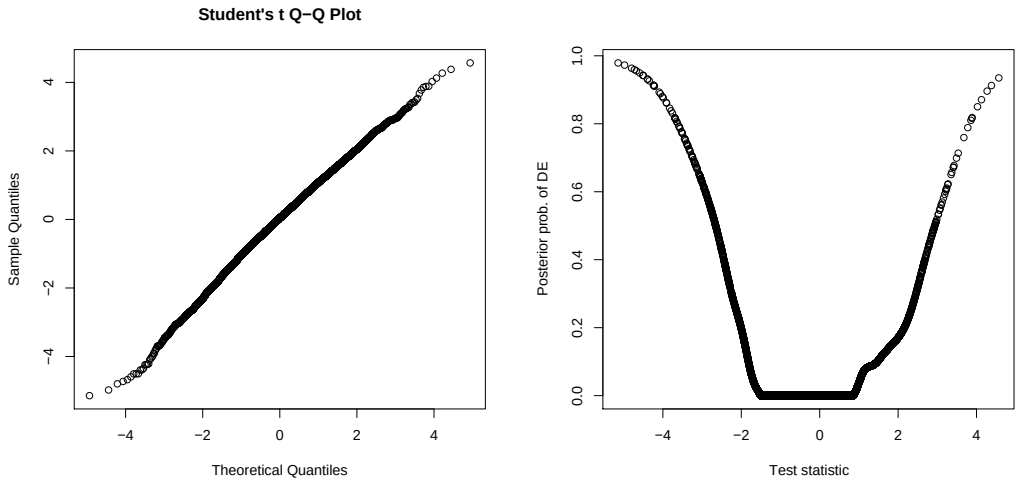

Figure 4: (a): assessing assumption of t distributed; (b): posterior probabilities vs. value of the test statistic

```
> qqt(tstat, df = nu)
```

```
> plot(ebayes.fit$tstat, ebayes.fit$wl2e.pde, xlab = "Test statistic",
+     ylab = "Posterior prob. of DE")
```

References

[1] C. Genovese and L. Wasserman. Operating characteristics and extensions of the false discovery rate procedure. *Journal of the Royal Statistical Society B*, 64:499–518, 2002.

[2] P. Müller, G. Parmigiani, C. Robert, and J. Rousseau. Optimal sample size for multiple testing: the case of gene expression microarrays. *Journal of the American Statistical Association*, 99:990–1001, 2004.

[3] G.K. Smyth. Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, 3, 2004.

[4] G.K. Smyth. Limma: linear models for microarray data. In R. Gentleman, V. Carey, S. Dudoit, R. Irizarry, and W. Huber, editors, *Bioinformatics*

and Computational Biology Solutions using R and Bioconductor, pages 397–420. Springer, New York, 2005.