



Using Our Discipline to Enhance Human Welfare August 7 - 11, 2005

Lab 5: Advanced Linear Modeling and Time-Series Analysis

Ben Bolstad. August 6-7, 2005.

1. Software required for this lab.

You will need R 2.1.0 or R 2.1.1 (<http://www.R-Project.org>) for this lab. This section lists all of the R packages you need to have installed in order to complete the lab exercises.

1.1 Required R packages

It is highly desirable to have these R packages in a directory in which you have write permission, especially for the *estrogen* package. You can use `.libPaths("C:/Custom/R/library/directory")` or `.libPaths("C:\\Custom\\R\\library\\directory")` before you run `install.packages()` (or equivalent) to install the package(s) in a customized directory location.

Note that most of the R packages can be installed from the Bioconductor site automatically using:

```
source("http://www.bioconductor.org/getBioC.R")
getBioC()
```

However, Bioconductor releases only occur once every six months, whereas the authors of these labs would typically use a more up-to-date version of some packages e.g. *limma*, which is why it is advisable to check the R package version numbers below and install them from the links provided if your installation is not up-to-date.

Package	Windows	MacOS X	Source
Biobase_1.5.12	Biobase 1.5.12.zip	Biobase 1.5.12.tgz	Biobase 1.5.12.tar.gz
drosEmbryo_1.0	drosEmbryo 1.0.zip	drosEmbryo 1.0.tar.gz	drosEmbryo 1.0.tar.gz
limma_2.0.2	limma 2.0.2.zip	limma 2.0.2.tar.gz	limma 2.0.2.tar.gz
statmod_1.1.1 or statmod_1.2.0	statmod 1.2.0.zip	statmod 1.2.0.tar.gz	statmod 1.2.0.tar.gz

2. Introduction

This lab is intended to give additional practice at analyzing a microarray experiment using linear models. It extends the principles introduced in Lab 4. As in Lab 4, the `limma` will serve as the primary analysis tool. The primary dataset we will use is a time-series dataset. Microarray time course datasets typically involve measurements of thousands of genes over a small number of time points, 4-10 time points are typical. Very few datasets have more than 20 timepoints.

3. Drosophila Embryogenesis dataset

This lab focuses on the dataset from Tomancak et al (2002). This dataset contains 36 DrosGenome1 microarrays. The goal was to investigate Drosophila embryogenesis using an Affymetrix microarray time course experiment. Wild type CantonS flies were expanded to large quantities and the entire population was split into 12 population cages that were subsequently treated equally. For three consecutive days fresh apple juice plates were introduced to each cage in the morning to allow two hours clearing of the retained embryos. Subsequently females in each cage were allowed to lay eggs for one hour before all twelve plates were removed simultaneously and transferred to 25-degree incubator and aged for 30 minutes. From then on at the end of each hour for the next 12 hours embryos from one plate were washed of the plate dechorionated and frozen in liquid nitrogen. Overall three independent replicates of twelve one-hour embryogenesis windows were collected over the period of 3 days.

4. Analyzing the Data

4.1 Loading the Data

First, load the packages and the dataset. The `drosEmbryoRMA` is an `exprSet` object containing RMA expression values for this dataset.

```
library(Biobase)
library(limma)
library(drosEmbryo)
data(drosEmbryoRMA)
```

As mentioned in Lab 3 and Lab 4, it is important to check the quality of your data before proceeding to further analysis. For brevity this quality check is skipped in this lab. However, a set of quality plots for this dataset, available at <http://www.stat.berkeley.edu/~bolstad/PLMImageGallery>, show few significant quality problems. One thing to note is that the arrays for the second replicate Day have slightly, though not significantly elevated NUSE values.

Examining the phenotypic data for this dataset:

```
pData(drosEmbryoRMA)
```

shows the sample names for this lab, the original filenames and time points. For the sample names the nomenclature is `XhrY` where `X` is the time-point, with value 1 through 12, and `Y` is the replication day, with value 1,2 or 3.

4.2 Setting up the model and contrasts

To begin our analysis the first thing we need to do is set up the model that will be fitted to each probeset. It seems sensible to include an effect both for time-point and replicate day. To prevent confusion between time point and replicate day we will label the days using letters. This may be accomplished using:

```
times <- pData(drosEmbryoRMA)$Time
rep.day <- rep(c("A", "B", "C"), 12)
design <- model.matrix(~factor(times) + factor(rep.day))
colnames(design) <- c(as.character(1:12), "B", "C")
```

Next we should set up the contrast matrix for all the possible comparisons we are interested in. In this case we are interested in looking for any changes in expression so we use the following contrast matrix:

```
cont.matrix <- rbind(rep(0, 11),
                    diag(11),
                    rep(0, 11),
                    rep(0, 11))
```

4.3 Fitting the model and finding the moderated F values

The next stage is to fit our model for each probeset, find the values of the contrasts we wish to examine, then compute the moderated F statistic. Type:

```
fit <- lmFit(drosEmbryoRMA, design)
fit <- contrasts.fit(fit, cont.matrix)
eb <- eBayes(fit)
modF <- eb$F
```

4.4 Examining expression patterns for top genes

We have just computed moderated F statistic values, looking at all possible changes between time points, for each probeset. It would now be useful to examine the expression values of the most significant genes. We will do this graphically. First we need to set up a few things:

```
par(cex=0.7, mfrow=c(2, 2), ask=T)
which.A <- seq(1, 34, 3)
which.B <- seq(2, 35, 3)
which.C <- seq(3, 36, 3)
```

Note that `par(ask=T)` means that R will ask you to press enter to move to the next plot.

In this lab we choose to look at the 500 probesets with most extreme values of the moderated F statistic. For each probeset we will plot each of the three replicate time series with a different symbol and color. Each plot will have the probeset identifier, moderated test statistic value and overall ranking displayed in the title. This can all be accomplished using the following code:

```
for(i in 0:499)
{
  indx <- rank(modF) == nrow(exprs(drosEmbryoRMA)) - i
  plot(1:12, exprs(drosEmbryoRMA)[indx, which.A], type="b", pch="A",
       lwd=2, col="red", xlab="Time", ylab="Expression",
       main=paste(rownames(exprs(drosEmbryoRMA))[indx], "modF=",
round(modF[indx], 2),
       "Ranking=", i+1), ylim=range(exprs(drosEmbryoRMA)[indx, ]))
  points(1:12, exprs(drosEmbryoRMA)[indx, which.B], pch="B",
        col="green", lwd=2, type="b")
  points(1:12, exprs(drosEmbryoRMA)[indx, which.C], pch="C",
        col="blue", lwd=2, type="b")
}
```

Press enter after you have finished examining each set of four probesets. As you progress through the plots you will see a variety of different expression profiles. Some probesets have initially high expression which drops after a few time points. Others have low initial expression but increase in expression near the end of the time course. There are a few probesets which begin low in expression, increase in the middle of the time course and decrease in expression at the end of the time period examined. Other temporal patterns are also visible. In general we notice that the A,B and C replicate time courses seem to match up very well to each other. If we were looking only for specific temporal patterns, for instance only early responding genes, we would repeat the procedure with a different contrast matrix.

4.5 Checking for differences between the time courses.

As we noted earlier, quality assessment showed that the arrays from the second replication of the time course had slightly elevated NUSE values. We can check if this has resulted in any differences in the temporal patterns for the three replicates. Do this using:

```
cont.matrix <- rbind(matrix(0,nrow=12,ncol=2),
                    c(1,0),
                    c(0,1))
fit <- lmFit(drosEmbryoRMA,design)
fit <- contrasts.fit(fit, cont.matrix)
eb <- eBayes(fit)
modF <- eb$F
for(i in 0:99)
{
  indx <- rank(modF)==nrow(exprs(drosEmbryoRMA))-i
  plot(1:12, exprs(drosEmbryoRMA)[indx,which.A], type="b",pch="A",
       lwd=2,col="red",xlab="Time", ylab="Expression",
       main=paste(rownames(exprs(drosEmbryoRMA))[indx], "modF=",
round(modF[indx],2),
       "Ranking=", i+1),ylim=range(exprs(drosEmbryoRMA)[indx,]))
  points(1:12, exprs(drosEmbryoRMA)[indx,which.B],pch="B",
col="green", lwd=2,type="b")
  points(1:12, exprs(drosEmbryoRMA)[indx,which.C],pch="C",
col="blue", lwd=2,type="b")
}
```

You will notice that in almost all cases the temporal patterns for A and C track each other closely, while the profile for B seems to show more difference.

5. Where To Go For More Information

1. Smyth, G. et al. Limma Users Guide
<http://bioinf.wehi.edu.au/limma/usersguide.pdf>
2. Tomancak P, Beaton A, Weiszmam R, Kwan E, Shu S, Lewis SE, Richards S, Ashburner M, Hartenstein V, Celniker SE, Rubin GM. *Systematic determination of patterns of gene expression during Drosophila embryogenesis*. Genome Biol. 2002;3(12):RESEARCH0088. Epub 2002 Dec 23.
<http://genomebiology.com/2002/3/12/research/0088.1>
3. Berkeley Drosophila Genome Project <http://www.fruitfly.org/cgi-bin/ex/insitu.pl>

6. Acknowledgments

Thank you to Yu Chuan Tai <http://www.stat.berkeley.edu/~yuchuan/> for suggestions about a suitable analysis. She will be releasing a `timecourse` software package later in 2005 with a more sophisticated test statistic (called MB) for time series analysis.

The authors of Tomancak et al (2002) should be commended for making their dataset available for public download.