Feature Selection and Limma

Aedín Culhane

December 15, 2011

Contents

CCCB course on R and Bioconductor, Dec 2011,

1 Introduction to the dataset for this tutorial

For the first part of this tutorial we will use a subset of the primate fibroblast gene expression from Karaman et al., Genome Research 2003. This study examines 3 groups, human, bonobo and gorilla expression profiles on Affymetrix HG_U95Av2 chips (1). This dataset contains 46 chips and is available in the Bioconducor library fibroEset (MAS5.0 data), and the web site `http://hacialab.usc.edu/supplement/karaman_etal_2003/index.html` (raw cel files).

In this tutorial we will look at 9 chips which have been normalised using *vsn*. For information I have included details of how I normalised these, at the end of the tutorial.

Download the normalized gene expression profiles from the web site (or Course wiki). The data are stored as a comma separated file, which is readable by MSExcel.

Install the following packages

```
source("http://www.bioconductor.org/biocLite.R")
biocLite("siggenes")
biocLite("RankProd")
biocLite("limma")
biocLite("fibroEset")
```

2 Load Dataset

As we will be examining Affymetrix data, load the package *affy*.

```
> require(affy)
> require(annaffy)
> require(hgu95av2.db)
> require(made4)
```

In this case the *vsn* normalised data are provided as a comma separated file. The sample annotations are in the file annt.txt, which is on the course webpage/wiki. To load in R:

```
> data.vsn<- read.csv("data.vsn.csv", as.is=TRUE, row.names=1)
> dim(data.vsn)

[1] 12625     9

> annt<-read.table("annt.txt", header=TRUE)
> annt[1:2,]

             Cels short.names Donor Age Gender  DT
1 AG_05414_AS.cel    AG_05414   Hsa  73      M 2.3
2 AG_11745_AS.cel    AG_11745   Hsa  43      F 1.8
  estb.same
1         D
2         D
```

This file contains the cel filenames (Cels), shorter names for the arrays (short.names), information about the Donor (Gorilla, Bonobo, Human), Age (years), Gender (male/female), doubling time (DT) of the cell lines, and information about whether cells where established from the same cell lines (estb.same). To view the data in a column in the `data.frame`, use the $ symbol and the column label. `table` can also be used to tabulate a summary of a categorical vector.

```
> annt$Donor

[1] Hsa Hsa Hsa Ggo Ppa Ppa Ggo Ppa Ggo
Levels: Ggo Hsa Ppa

> table(annt$Donor)

Ggo Hsa Ppa
  3   3   3

> table(annt$Gender)

F M
5 4
```

Lets convert this into an expressionSet as it be will easier to use in Bioconductor First we need to check that the column names of the data set match the rownames of the annotation

```
> names(data.vsn)

[1] "AG_05414_AS.cel"       "AG_11745_AS.cel"
[3] "AG_13927_AS.cel"       "KB_5047_2070_2_AS.CEL"
[5] "KB_5275_2_AS.CEL"      "KB_5828_AS.cel"
[7] "KB_6268_2_AS.cel"      "KB_8025_AS.cel"
[9] "KB_8840_AS.cel"

> annt

                   Cels short.names Donor Age Gender  DT
1       AG_05414_AS.cel    AG_05414   Hsa  73      M 2.3
2       AG_11745_AS.cel    AG_11745   Hsa  43      F 1.8
3       AG_13927_AS.cel    AG_13927   Hsa  45      F 2.8
4 KB_5047_2070_2_AS.CEL     KB_5047   Ggo  19      F 2.0
5      KB_5275_2_AS.CEL     KB_5275   Ppa   2      M 2.4
6        KB_5828_AS.cel     KB_5828   Ppa  12      M 2.7
7      KB_6268_2_AS.cel     KB_6268   Ggo  19      F 2.0
8        KB_8025_AS.cel     KB_8025   Ppa  19      M 2.0
9        KB_8840_AS.cel     KB_8840   Ggo   2      F 2.5
  estb.same
1         D
2         D
3         -
4         -
```

```
5         -
6         -
7         -
8         -
9         -
```

```
> rownames(annt) <-annt$Cels
```

```
> makeEset<-function(eSet, annt){
+     #Creating an ExpressionSet from eSet, a normalized gene expression matrix
+     # and annt, a data.frame containing annotation
+     metadata <- data.frame(labelDescription = colnames(annt), row.names=colnames(an
+     phenoData<-new("AnnotatedDataFrame", data=annt, varMetadata=metadata)
+     if (inherits(eSet, "data.frame")) eSet= as.matrix(eSet)
+     if (inherits(eSet, "ExpressionSet")) eSet=exprs(eSet)
+     data.eSet<-new("ExpressionSet", exprs=eSet, phenoData=phenoData)
+     print(varLabels(data.eSet))
+     return(data.eSet)
+ }
> eSet<-makeEset(data.vsn, annt)
```

```
[1] "Cels"        "short.names" "Donor"       "Age"
[5] "Gender"      "DT"          "estb.same"
```

We will look at a simple 2 class comparison, human v non-human (other primate). So lets add that factor to the eSet

```
> human<- eSet$Donor=="Hsa"
> table(human)
```

```
human
FALSE  TRUE
    6     3
```

```
> eSet$Human<-human
> eSet
```

```
ExpressionSet (storageMode: lockedEnvironment)
assayData: 12625 features, 9 samples
  element names: exprs
protocolData: none
phenoData
  sampleNames: AG_05414_AS.cel AG_11745_AS.cel ...
    KB_8840_AS.cel (9 total)
  varLabels: Cels short.names ... Human (8 total)
  varMetadata: labelDescription
featureData: none
experimentData: use 'experimentData(object)'
Annotation:
```

It will also be useful to have a set of gene annotation. So get the gene symbols for the hgu95av2 chip

```
> affy.id = featureNames(eSet)
> affy.symbols<-aafSymbol(affy.id, "hgu95av2.db")
> affy.symbols <-getText(affy.symbols)
> names(affy.symbols)<-featureNames(eSet)
```

It is a good idea to ALWAYS perform an exploratory analysis of the data PRIOR to feature selection. This will enable one to get a feel for bias in the data, and may indicate that further normalization or replicates are required. See the ordination tutorial for examples of exploratory analysis approaches.

3 Limma

The package *limma* (6), (7) has a very comprehensive user manual which is available from `http://bioinf.wehi.edu.au/limma/`. Please review this.

Although *limma* is a large package, with normalization and many other functions, the core of *limma* is the fitting of gene-wise linear models to microarray data.

We will apply this very simple example using, limma, however much more complex analysis can be applied. These include the case where multiple factors (eg Dose Response and Time 0,24,48 hours) are considered and one what to obtain the interaction between co-variates in this factorial design.

```
> require(limma)
```

Use the `vignette("limma")` or `limmaUsersGuider()` to find help on limma.

Please have a look at the limma userguide `http://www.bioconductor.org/packages/release/bioc/html/limma.html`. This is very extensive, its 100 pages!

To fit a very simple design, you can create a design matrix.

```
> design= model.matrix(~eSet$Human)
> fit <- lmFit(eSet,design)
> fit <- eBayes(fit)
> topTable(fit,coef=2)
```

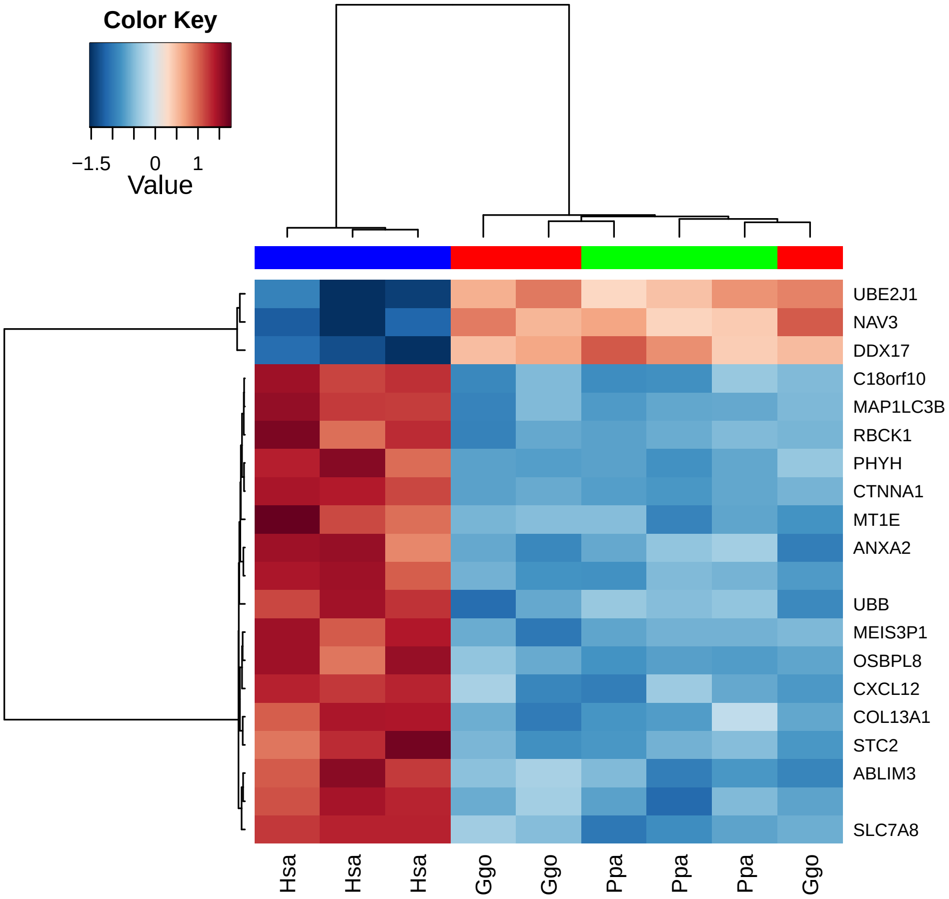
```
            ID      logFC   AveExpr         t      P.Value
11262 41155_at  2.8190437 10.279341  30.01078 7.837652e-11
6043  35985_at  2.5377652 10.064852  18.93091 6.158343e-09
9460  39370_at  1.6676149 10.298392  18.65525 7.067721e-09
2750  32724_at  1.1662335  9.135296  14.43195 7.723514e-08
11547 41438_at  1.2001230  9.388458  13.94891 1.057180e-07
2691  32666_at  2.5258429  9.864402  13.70948 1.239757e-07
11367 41260_at -1.9185845 10.615935 -13.44403 1.483680e-07
11378 41271_at  0.8715227 10.482055  13.27435 1.667010e-07
2062  32043_at  1.5862742  9.763682  12.86454 2.221256e-07
348    1323_at  2.2758296 12.635851  12.61591 2.654496e-07
         adj.P.Val         B
11262 9.895036e-07 12.699581
6043  2.974332e-05 10.241602
9460  2.974332e-05 10.146763
2750  2.437734e-04  8.347922
11547 2.608656e-04  8.091996
2691  2.608656e-04  7.960518
11367 2.630749e-04  7.811027
11378 2.630749e-04  7.713357
2062  3.062217e-04  7.470421
348   3.062217e-04  7.318010
```

```
> limmaRes = topTable(fit,coef=2,  p.value=0.001, number=500)
> print(nrow(limmaRes))
```

```
[1] 20
```

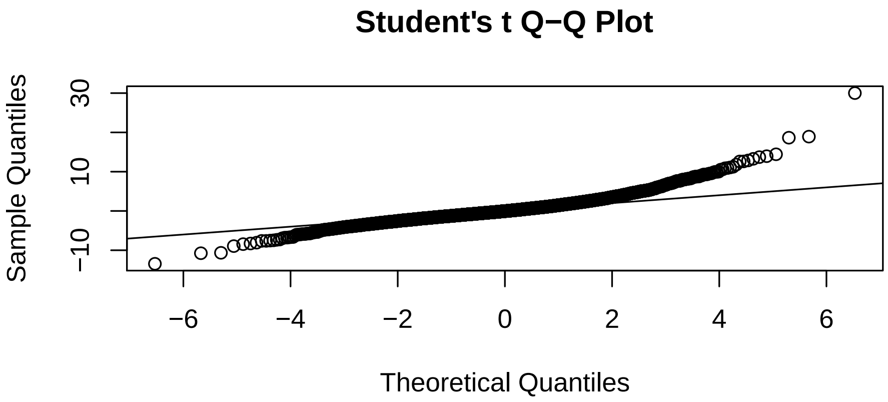
```
> heatplot(eSet[limmaRes$ID,], classvec=eSet$Donor, labRow=affy.symbols[limmaRes$ID],
```

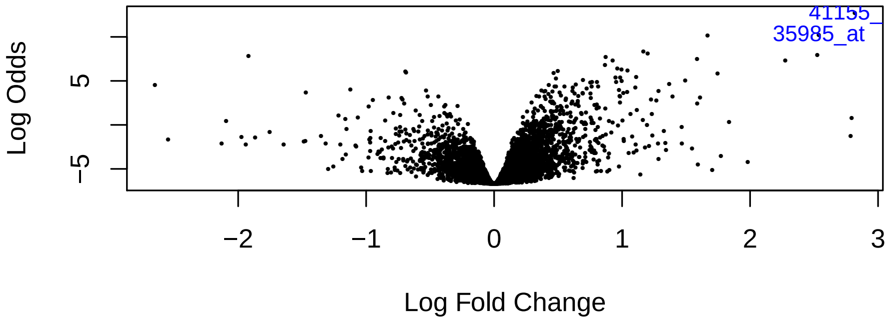
```
[1] "Data (original) range:  8.62 14.37"
[1] "Data (scale) range:  -1.53 1.77"
[1] "Data scaled to range:  -1.53 1.77"
     Class Color
[1,] "Ggo" "red"
[2,] "Hsa" "blue"
[3,] "Ppa" "green"
```



Therefore there were 20 genes with a p-value less than 0.0001.

```
> par(mfrow=c(2,1))
> qqt(fit$t[,2],df=fit$df.residual+fit$df.prior)
> abline(0,1)
> volcanoplot(fit,coef=2,highlight=2)
```





4 Rank Products Analysis

Rank Products was described by Rainer Breitling and in available in the Bioconductor package RankProd (5), (4). To run Rank Products Analysis:

```
> require(RankProd)
> RP.out <- RP(eSet, eSet$Human, rand=123)

Rank Product analysis for two-class case

Starting 100 permutations...
Computing pfp ..
Outputing the results ..

> plotRP(RP.out, cutoff=0.05)
> RP.res = topGene(RP.out,cutoff=0.05,method="pfp",logged=TRUE,logbase=2,gene.names=a
Table1: Genes called significant under class1 < class2

Table2: Genes called significant under class1 > class2
```

```
> names(RP.res)

[1] "Table1" "Table2"

> RP.res$Table1[1:10,]

         gene.index RP/Rsum FC:(class1/class2)   pfp
CTNNA1        11262  4.3046             0.1417 0.000
TGFBI           416  4.8969             0.1442 0.000
CXCL12         2691  6.9596             0.1736 0.000
               6043  7.2590             0.1722 0.000
MMP3          12004  8.0134             0.1450 0.000
UBB             348  8.9049             0.2065 0.000
ANXA2         12321 21.4063             0.2981 0.000
PODXL         10534 22.6307             0.2801 0.000
MAP1LC3B       9460 23.7955             0.3148 0.000
STC2           2062 29.0234             0.3330 0.001
         P.value
CTNNA1         0
TGFBI          0
CXCL12         0
               0
MMP3           0
UBB            0
ANXA2          0
PODXL          0
MAP1LC3B       0
STC2           0

> RP.res$Table2[1:10,]

         gene.index RP/Rsum FC:(class1/class2) pfp P.value
MFGE8          4446  3.6607             6.2787   0       0
IGFBP5         8733  5.4255             5.8460   0       0
CRIP1          3263 10.6694             4.2692   0       0
DDX17         11367 11.5299             3.7805   0       0
IGFBP2        10522 12.5817             4.3746   0       0
IGFBP5          428 14.2647             3.9278   0       0
COL11A1        7968 14.6925             3.6477   0       0
```

```
IGFBP2          831 18.1408              3.8370   0       0
SERPINB2       7254 20.3639              3.3726   0       0
CDH13         12049 27.2069              2.7716   0       0
```

RankProd also has an advanced rank product method to identify differentially expressed genes but combining data from different studies, e.g. data sets generated at different laboratories. See the function `RPadvance`.

5 Which did best?

Load the complete dataset.

```
> require(fibroEset)
> data(fibroEset)
> phenoData(fibroEset)
```

Examine each of the above genesets from Rank Products and Limma in the complete dataset.

- What overlap in there is the genelists?
- Draw a heat map and perform a cluster analysis on each.
- Re-examine the Correpondence Analysis and Principal Component (ord) of the all genes (complete datset). Where these genes present at the ends of the axes?

6 More on Factorial Designs and Limma

See the limma user guide, follow the example in the CASE Studies section entitled ”11.4 Estrogen Data: A 2x2 Factorial Experiment with Affymetrix Arrays”

TASK: Download the annotation for all of the celfiles. Fit a design which includes >1 covariate, a factorial design. Download the phenotype data for the complete dataset (Gender and Species).

7 Creating Annotation tables (HTML)

There are several further annotation tools in annAffy
To obtain a browsable html table of gene annotation:

```
> anncols<-aaf.handler()
> anncols
> anntable <- aafTableAnn(limmaRes$ID, "hgu95av2.db", anncols)
> saveHTML(anntable, "example1.html", title = "Example")
```

8 Annotating using biomaRt

BiomaRt connects to the Biomart resource at `www.biomart.org` to pull data from marts including the Ensembl genome browser, Uniprot and HapMap.

```
> require(biomaRt)
> mart <- useMart("ensembl")
> mart<-useDataset("hsapiens_gene_ensembl",mart)
> res<-getBM(attributes=c("affy_hg_u95av2","hgnc_symbol", "chromosome_name","band"),f
> res[1:5,]
```

```
  affy_hg_u95av2 hgnc_symbol chromosome_name  band
1     37486_f_at     MEIS3P1              17   p12
2        1323_at                          17 p11.2
3     37486_f_at     MEIS3P2              17 p11.2
4        1323_at         UBB              17 p11.2
5       39040_at      UBE2J1               6   q15
```

to see more Datasets, filters and attributes see

```
> listDatasets(mart)[1:10,]
```

```
                   dataset
1   oanatinus_gene_ensembl
2    tguttata_gene_ensembl
3  cporcellus_gene_ensembl
4  gaculeatus_gene_ensembl
5   lafricana_gene_ensembl
6  mlucifugus_gene_ensembl
7    hsapiens_gene_ensembl
8  choffmanni_gene_ensembl
9   csavignyi_gene_ensembl
10     fcatus_gene_ensembl
                               description     version
1   Ornithorhynchus anatinus genes (OANA5)       OANA5
2  Taeniopygia guttata genes (taeGut3.2.4) taeGut3.2.4
3          Cavia porcellus genes (cavPor3)     cavPor3
4   Gasterosteus aculeatus genes (BROADS1)     BROADS1
5       Loxodonta africana genes (loxAfr3)     loxAfr3
6         Myotis lucifugus genes (myoLuc2)     myoLuc2
7           Homo sapiens genes (GRCh37.p5)   GRCh37.p5
8      Choloepus hoffmanni genes (choHof1)     choHof1
9           Ciona savignyi genes (CSAV2.0)     CSAV2.0
10                 Felis catus genes (CAT)         CAT
```

```
> listFilters(mart)[1:10,]

             name     description
1  chromosome_name Chromosome name
2            start Gene Start (bp)
3              end   Gene End (bp)
4       band_start      Band Start
5         band_end        Band End
6     marker_start    Marker Start
7       marker_end      Marker End
8             type            Type
9    encode_region   Encode region
10          strand          Strand

> listAttributes(mart)[1:10,]

                             name
1                 ensembl_gene_id
2           ensembl_transcript_id
3              ensembl_peptide_id
4  canonical_transcript_stable_id
5                     description
6                 chromosome_name
7                  start_position
8                    end_position
9                          strand
10                           band
                         description
1                    Ensembl Gene ID
2              Ensembl Transcript ID
3                 Ensembl Protein ID
4  Canonical transcript stable ID(s)
5                        Description
6                    Chromosome Name
7                    Gene Start (bp)
8                      Gene End (bp)
9                             Strand
10                              Band
```

BiomaRt is highly versatile,see its vignette on its Bioconductor homepage `http://www.bioconductor.org/packages/release/bioc/html/biomaRt.html`

9 Session Info

Information about this session:

```
> sessionInfo()

R version 2.14.0 (2011-10-31)
Platform: i386-pc-mingw32/i386 (32-bit)

locale:
[1] LC_COLLATE=English_United States.1252
[2] LC_CTYPE=English_United States.1252
[3] LC_MONETARY=English_United States.1252
[4] LC_NUMERIC=C
[5] LC_TIME=English_United States.1252

attached base packages:
[1] grid      stats     graphics  grDevices utils
[6] datasets  methods   base

other attached packages:
 [1] biomaRt_2.10.0        RankProd_2.26.0
 [3] limma_3.10.0          made4_1.28.0
 [5] scatterplot3d_0.3-33  gplots_2.10.1
 [7] KernSmooth_2.23-7     caTools_1.12
 [9] bitops_1.0-4.1        gdata_2.8.2
[11] gtools_2.6.2          RColorBrewer_1.0-5
[13] ade4_1.4-17           hgu95av2.db_2.6.3
[15] org.Hs.eg.db_2.6.4    annaffy_1.26.0
[17] KEGG.db_2.6.1         GO.db_2.6.1
[19] RSQLite_0.11.1        DBI_0.2-5
[21] AnnotationDbi_1.16.10 affy_1.32.0
[23] Biobase_2.14.0

loaded via a namespace (and not attached):
[1] affyio_1.22.0         BiocInstaller_1.2.1
[3] IRanges_1.12.5        preprocessCore_1.16.0
[5] RCurl_1.8-0.1         tools_2.14.0
[7] XML_3.6-2.1           zlibbioc_1.0.0
```

References

[1] Karaman MW, Houck ML, Chemnick LG, Nagpal S, Chawannakul D, Sudano D, Pike BL, Ho VV, Ryder OA, Hacia JG Comparative analysis of gene-expression patterns in human and African great ape cultured fibroblasts. *Genome Res.* **13(7)**:1619-30.2003.

[2] Jeffery IB, Higgins DG, Culhane AC. (2006) Comparison and evaluation of microarray feature selection methods. *BMC Bioinformatics* **7**:359. 2006.

[3] Tusher VG, Tibshirani R, Chu G. Significance analysis of microarrays applied to the ionizing radiation response. *Proc Natl Acad Sci U S A*.**98(9)**:5116-21. 2001.

[4] Hong F, Breitling R, McEntee CW, Wittner BS, Nemhauser JL, Chory J. RankProd: a bioconductor package for detecting differentially expressed genes in meta-analysis. *Bioinformatics* **22(22)**:2825-7. 2006

[5] Breitling R, Armengaud P, Amtmann A, Herzyk P. Rank products: a simple, yet powerful, new method to detect differentially regulated genes in replicated microarray experiments. *FEBS Lett* **573(1-3)**:83-92. 2004

[6] Smyth, G. K. Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology* **3(1)** Article 3. 2004. `http://www.bepress.com/sagmb/vol3/iss1/art3`

[7] Smyth, G. K., Michaud, J., and Scott, H. The use of within-array replicate spots for assessing differential expression in microarray experiments. *Bioinformatics* **21(9)**: 2067-2075. 2005.