Very Brief Normalization

Aedín Culhane

May 16, 2012

Contents

We already mentioned how to read data into R, but once its read in it will need to be normalized. There are numerous normalization methods in Bioconductor, of which quantile and lowess normalization are probably the most widely used.

1 normalization Affymetrix data

Download a .zip of 9 cel file cels.zip from the coures website. Clicking on this will extract the files to a folder cels.

A summary of the main function are:

```
> require(affy)
> require(made4)
> eset1<-justRMA(celfile.path="cels")
> overview(eset1)
```

```
> annt<-read.table("annt.txt", header=TRUE)
> overview(eset1, labels=annt$Donor, classvec=annt$Donor)
```

Other approaches

- ReadAffy() Reads all *.CEL (*.cel) files in your current working directory
- rma(abatch) RMA normalize. Data are stored as ExpressionSet class. For large data sets use justRMA()
- mas5(abatch) MAS 5.0 normalize module instead of RMA
- gcrma(abatch) gcrma normalize. Load 'library(gcrma)' first.
- justPlier(abatch) plier normalize. load 'library(plier)' first.
- mas5calls(abatch) Generates MAS 5.0 P/M/A calls.
- expresso(mydata, normalize.method="invariantset", bg.correct=FALSE, pmcorrect.method="pmon summary.method="liwong") Generates expression calls similar to dChip (MBEI) invariant set method from Li and Wong.

2 affycoretools

affycoretools will normalize all cel files in current working directory, perform qc plots and export normalized data to file. Works for mas5, rma and gcrma.

```
> library(affycoretools)
> affystart(plot=T, express="mas5")
```

3 normalization Affymetrix data -custom cdf

Several groups, including the University of Michigan BrainArray project provide custom cdf files will allow one to summarize Affymetrix probes to the gene, transcript rather than probeset. `http://brainarray.mbni.med.umich.edu/brainarray/default.asp`

For R version 2.14 the custom cdf packages are available on `http://brainarray.mbni.med.umich.edu/Brainarray/Database/CustomCDF/14.1.0/ensg.asp`. Download the EnsEMBL gene summaries for hgu95av2.

Whilst the following should work

```
> source("http://www.bioconductor.org/biocLite.R")
> biocLite("HGU95Av2_Hs_ENSG",respos="http://brainarray.mbni.med.umich.edu/bioc")
```

It may not in which case you need to download 2 package and do a local install of the .zip files (or source, depending on your Operating system)

The are 2 files: the cdf file `http://brainarray.mbni.med.umich.edu/Brainarray/Database/CustomCDF/14.1.0/ensg.download/hgu95av2hsensgcdf_14.1.0.zip` and the probe file `http://brainarray.mbni.med.umich.edu/Brainarray/Database/CustomCDF/14.1.0/ensg.download/hgu95av2hsensgprobe_14.1.0.zip`

Download these, install and now run the normalization

```
> abatch<-ReadAffy(celfile.path="cels",cdfname = "HGU95Av2_Hs_ENSG" )
> eset2<-rma(abatch)
```

4 ArrayQualityMetrics - checking for data quality

ArrayQuality metrics will produce a QC report for one colour, two colour, Affymetrix, or Illumina data.

```
> require(arrayQualityMetrics)
> arrayQualityMetrics(expressionset = eset1, outdir = "Report_for_Cels", force = TRUE
```

For Affymetrix, the library AffyPLM contains useful scores NUSE, RLE. I use these with the percent P calls (mas5calls) to detect outliers.

5 fRMA

fRMA is frozen RMA and is useful when trying to collate many studies from U133a or U133plus2 Affymetrix chips. Just use frma instead of rma. Only works on Affymetrix u133a and u133plus2.

6 vsn normalization of data

vsn can be applied to Affmetrix and 2 channell data. To produce the vsn normalized data:

```
> getwd()
> dir()
> library(affy)
> library(vsn)
> cels <- list.celfiles()
> data <- ReadAffy(filenames= cels)
> normalize.AffyBatch.methods <- c(normalize.AffyBatch.methods, "vsn")
> data.vsn <- es1 = expresso(data, bg.correct = FALSE,
```

```
+ normalize.method = "vsn", pmcorrect.method = "pmonly",
+ summary.method   = "medianpolish")
> exprs2excel(data.vsn, file="data.vsn.csv")
```

A tab delimited text file could also be saved using

```
> write.exprs(data.vsn, file="data.rma.txt")
```

7 limma normalization of data

Limma has an extensive manual (Worth reading) and will read 2 channell data, and process these, Affymetrix and RNA-seq data.

Limma expects a file 'Targets.txt' which contains the sample names and information. It uses this to read the images to an RGList.

The default within array normalization is print-tip loess normalization. This is following by background correction and quantile between array normalization.

```
> targets <- readTargets("Targets.txt")
> RG <- read.maimages(targets$FileName, source="spot", sep="\t", path="./")
> MA <- normalizeWithinArrays(RG)
> MA.pAq <- normalizeBetweenArrays(MA, method="quantile")
```

In some cases this is not appropriate. For example, Agilent arrays do not have print-tip groups, so one should use global loess normalization or robust spline

```
> MA <- normalizeWithinArrays(RG, method="loess")
> MA <- normalizeWithinArrays(RG, method="robustspline")
```

normalizeBetweenArrays with method="vsn" will perform vsn normalization. See the limma manual for an indepth discussion of each of these

8 Summarize Probeset to Genes

The most widely used approaches are either probes with greatest CV (coefficient of variation) or sd. Simply order the probeset by CV (sd/mean) or the sd, then remove duplicates. An alternative is to use a custom chip definition (describes above)