HOMER

Software for motif discovery and next-gen sequencing analysis

Processing Affymetrix Gene Expression Arrays

Analyzing Affy microarrays with Bioconductor is "*relatively*" easy, particularly if all you want is to get the gene expression matrix. Once you have the gene expression values, much of the analysis techniques that can be used for RNA-Seq analysis can also be used for microarrays. Below is some basic information to get you started.

*This mini-tutorial assumes you are trying to analyze affymetrix arrays from CEL files. CEL files the 'raw' data files produced at the end of the array scan, and are normally deposited in gene expression databases like GEO. If you do not have the CEL files, make every attempt to find them (if it's your data)!

R/Bioconductor

The Bioconductor community has been one of the primary driving forces behind microarray analysis in the past decade. There are many packages, tutorials, and countless additional resources available on their site.

One of the best resources for R/Bioconductor is maintained at UC Riverside by IIGB: Manual

Installing Bioconductor and packages in R

To install R, go to the R homepage and install the appropriate version for your computer (CRAN download page).

Go here to get a full description about how what bioconductor is and how to install it (below is the cheat sheet):
After you start R, type:

```
# load the script from the internet that is used in install bioconductor
source("http://bioconductor.org/biocLite.R")
```

```
# Each of these commands tells Bioconductor to download and install each package
biocLite("affy")
biocLite("oligo")
biocLite("limma")
```

You may have to install additional packages if needed (just substitute the package name in the biocLite command).

Couple of R basics

Main R tutorial

This one isn't too bad either.

Also, each bioconductor package has it's own tutorial/documentation (usually they offer a lot of explanation)

Also, to change your current working directory in R, got to the top menu "Misc -> Change Working Directory"

Normalizing Affy Data from CEL files

>90% of all studies using Affy chips will use RMA for normalization. RMA (Robust Multi-array Average) was developed in the Speed Lab at UC Berkeley (Irizarry et al.). Variants such as gcRMA are also available.

Three prime Affymetrix Arrays (older ones)

Normalizing older Affymetrix arrays is EASY!!! This methods and technology are very mature.

First, save all of the CEL files you want to analyze in a single directory. They may be gzipped (*.gz) - you do not need to gunzip them. Once in R, use the top menu to change your current directory to the same directory you saved the CEL files in ("Misc -> Change Working Directory"). Then run the following commands:

```
# load the affy library
library(affy)

# Read in the CEL files in the directory, then normalize the data
data <- ReadAffy()
eset <- rma(data)

# Finally, save the data to an output file to be used by other programs, etc (Data will be log2 transformed and normalized)
write.exprs(eset,file="data.txt")
```

Newer Affymetrix Arrays (Gene ST arrays, etc.)

The old affy package doesn't work with these - instead you'll have to use the oligo package:

```
# load the oligo library
library(oligo)

# Read in the CEL files in the directory
celFiles <- list.celfiles()
affyRaw <- read.celfiles(celFiles)

# You might need to install and load a package for the specific array you are using (this example is mouse gene 2.0 ST)
# It may try to load it automatically, but may fail.  Install & load the library manually if this happens.
library(pd.mogene.2.0.st)
eset <- rma(affyRaw)

# Finally, save the data to an output file to be used by other programs, etc (Data will be log2 transformed and normalized)
write.exprs(eset,file="data.txt")
```

Adding Gene Annotation to Normalized Expression Output

From what I can tell, there is no super-easy way to do this. Huge oversight by the Bioconductor guys - I may have just missed it (if so, please let me know !!!)

Below is how I get it to work. The key is to find the annotation package for your array. For example, the mouse gene 2.0 ST array has a package named 'mogene20sttranscriptcluster.db'. You may have to do some google searching to find the correct package for your project. Install it with biocLite("mogene20sttranscriptcluster.db").

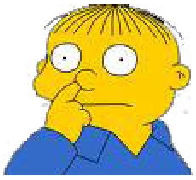
```
# This assumes you already normalized the data, and the object "eset" has the data
```

in it (from above)

```
# Load annotation library
library(mogene20sttranscriptcluster.db)

# Strategy is to create data frame objects and merge them together - put expression
info into a data frame
my_frame <- data.frame(exprs(eset))

# Put annotation information in a data frame.  To get specific fields, use
packageNameSYMBOL, where the caps part names the type of data you're after
# To get a list of available annotation information, run the packagename with () at
the end, i.e. mogene20sttranscriptcluster()
Annot <-
data.frame(ACCNUM=sapply(contents(mogene20sttranscriptclusterACCNUM),
paste, collapse=", "),
SYMBOL=sapply(contents(mogene20sttranscriptclusterSYMBOL), paste,
collapse=", "),
DESC=sapply(contents(mogene20sttranscriptclusterGENENAME), paste,
collapse=", "))

# Merge data frames together (like a database table join)
all <- merge(Annot, my_frame, by.x=0, by.y=0, all=T)

# Write out to a file:
write.table(all,file="data.ann.txt",sep="\t")
```

Advanced Affymetrix with R/Bioconductor

Check out this link for more ideas about how to analyze your data:
http://manuals.bioinformatics.ucr.edu/home/R_BioCondManual



Can't figure something out? Questions, comments, concerns, or other feedback:
cbenner@salk.edu