Elements of R for Genetics & Bioinformatics
Session 9: Bioconductor #2

Note: all these exercises follow the class examples closely. Try one which is close to your interests in genetics.

snpStats
Install the `snpStats` package from Bioconductor, and load it into your current session

Download the file `AMDchrom1snpStats.Rdata` from the course site, and load it into your current session with

```
> load("AMDchrom1snpStats.Rdata")
```

- this will create an object called `amd1`. This is a `snpStats` representation of the chromosome 1 data from a (small) genome-wide association study, with 96 AMD cases and 50 controls. Make an R representation of the case-control status with e.g.

```
> cc.status <- rep(1:0, times=c(96,50) )
```

Using the session 9 slides as a guide, and the `snpStats` vignette to help you, gives summaries of SNP-by-SNP and subject-by-subject quality-control measures for this data. Using the whole dataset, compute association tests of case/control status with all SNPs. How long does this take, compared to your code for session 5?

ShortRead

Install the `yeastRNASeq` and the `ShortRead` packages from Bioconductor, and load them into your current session.

Get a list of the short-read files in the yeastRNASeq package with

```
files<- list.files(file.path(system.file(package =
"yeastRNASeq"), "reads"), pattern = "bowtie", full.names =
TRUE)
```

and read the first one in with

```
reads1 <- readAligned(files[1], type="Bowtie")
```

The data are available more directly with

```
data(yeastAligned)
```

which returns a list of four data sets. Check that the one you read in is the same as the first one in the list.

1) extract the genomic positions from each of the four data sets (using the `position()` function) and draw histograms of these.

2) extract the read quality and turn it into a matrix, with e.g.

```
a <- as(quality(aligned[["mut_1_f"]]),"matrix")
```

then plot mean quality for each of the 26 read positions (column means of the matrix) and plot mean quality for each read (row means of the matrix) against genomic position

edgeR

Install `edgeR` packages and load them into your session. Load exon counts from an experiment comparing normal and tumour tissue:

```
rawdata <-
read.delim("http://bioinf.wehi.edu.au/edgeR/TableS1.txt",
check.names=FALSE, stringsAsFactors=FALSE)
```

The counts are in columns 4-9, with gene information in the first three columnsn. Put them together with

```
y <- DGEList(counts=rawdata[,4:9], genes=rawdata[,1:3],
group= factor(c("N","T","N","T","N","T")))
```

and then follow the same steps as in the slides.

Actually, these are paired samples, so the analysis treating them as six separate samples is conservative. You can start up the manual for edgeR with `edgeRUsersGuide()`
Starting at page 41 there is a more detailed analysis of this dataset using regression models to account for the paired samples.