Pathway Analysis for RNA-Seq

Cavan Reilly

December 6, 2019

Table of contents

Overview

We've seen that there are tools for testing for overenrichment of certain biological pathways, such as the methods based on the gene ontology.

It would seem that the same approaches can be used for the analysis of RNA-Seq experiments.

While one can't rely on the annotation packages specific to certain microarrays, one can still obtain data from biomart and take the same approach.

goseq

Some have maintained that these approaches are inadequate for RNA-Seq data.

The main argument advanced for this case has to do with claims that RNA-Seq has lower power for testing for differences across groups for genes that are shorter.

If one assumes

1. one is testing for differences with counts for each gene
2. these counts are distributed according to the Poisson distribution
3. one is using a 2 sample t-test to test for differences

goseq

Then it follows that one has lower power for shorter genes.

This is because the power of a 2 sample t-test depends on the ratio of the difference in the means across 2 groups relative to the within group standard deviation.

If the data are distributed according to a Poisson distribution in each group, and L is the length of the gene, then

mean of the counts in group $1 = L\mu_1$

and

mean of the counts in group $2 = L\mu_2$

goseq

and since the mean and variance are the same for a Poisson random variable:

variance of the counts in group 1$=L\mu_1$

and

variance of the counts in group 2$=L\mu_2$

so the mean variance is $\frac{L(\mu_1+\mu_2)}{2}$.

Thus the ratio of the difference in the means to the standard deviation is

$$\frac{L\mu_1 - L\mu_2}{\sqrt{L(\mu_1+\mu_2)/2}}$$

which is $\sqrt{L}(\mu_1-\mu_2)/(0.5(\mu_1+\mu_2))^{0.5}$

therefore the power is greater for longer genes.

goseq

However:

1. no one is simply testing for differences with counts for each gene currently (e.g. edgeR)
2. no one claims that RNA-Seq data are well modeled as Poisson random variables
3. no one is using a 2 sample t-test to test for differences between groups

Nonetheless there are methods that have been designed to correct for differences in gene length when conducting geneset enrichment analyses: the goseq package.

goseq

To use this approach, first we need to find the appropriate gtf file for our organism, then make a transcriptome database from that file.

```
> library(GenomicFeatures)
> btdb <- makeTxDbFromGFF("/export/home/courses/ph7445/
+   data/Bos_taurus.UMD3.1.83.gtf",dataSource=
+   "ftp.ensembl.org/pub/release-83/gtf/
+   bos_taurus/",organism="Bos taurus")
```

Then we get a gene based dataset from this database, and from that get summaries of each gene's length.

```
> txsByGene=transcriptsBy(btdb,"gene")
> lengthData=median(width(txsByGene))
```

goseq

```
> summary(lengthData)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
     26     963    8396   30910   31200 1851000
```

After the length data is set up then we need to define an indicator variable for which genes differ and give that indicator gene IDs.

```
> library(goseq)
> bovDif=as.integer(padj<0.05)
> names(bovDif)=bovIDs[apply(bovCnts,1,min)>4]
> bovLen=lengthData[names(lengthData) %in% names(bovDif)]
```

goseq

Then we see there are genes without length data, so we need to exclude those from further consideration.

```
> table(names(bovDif) %in% names(bovLen))

FALSE  TRUE
  374 10616
> table(names(bovLen) %in% names(bovDif))

 TRUE
10616
```

goseq

```
> bovDif1=bovDif[names(bovDif) %in% names(bovLen)]
```

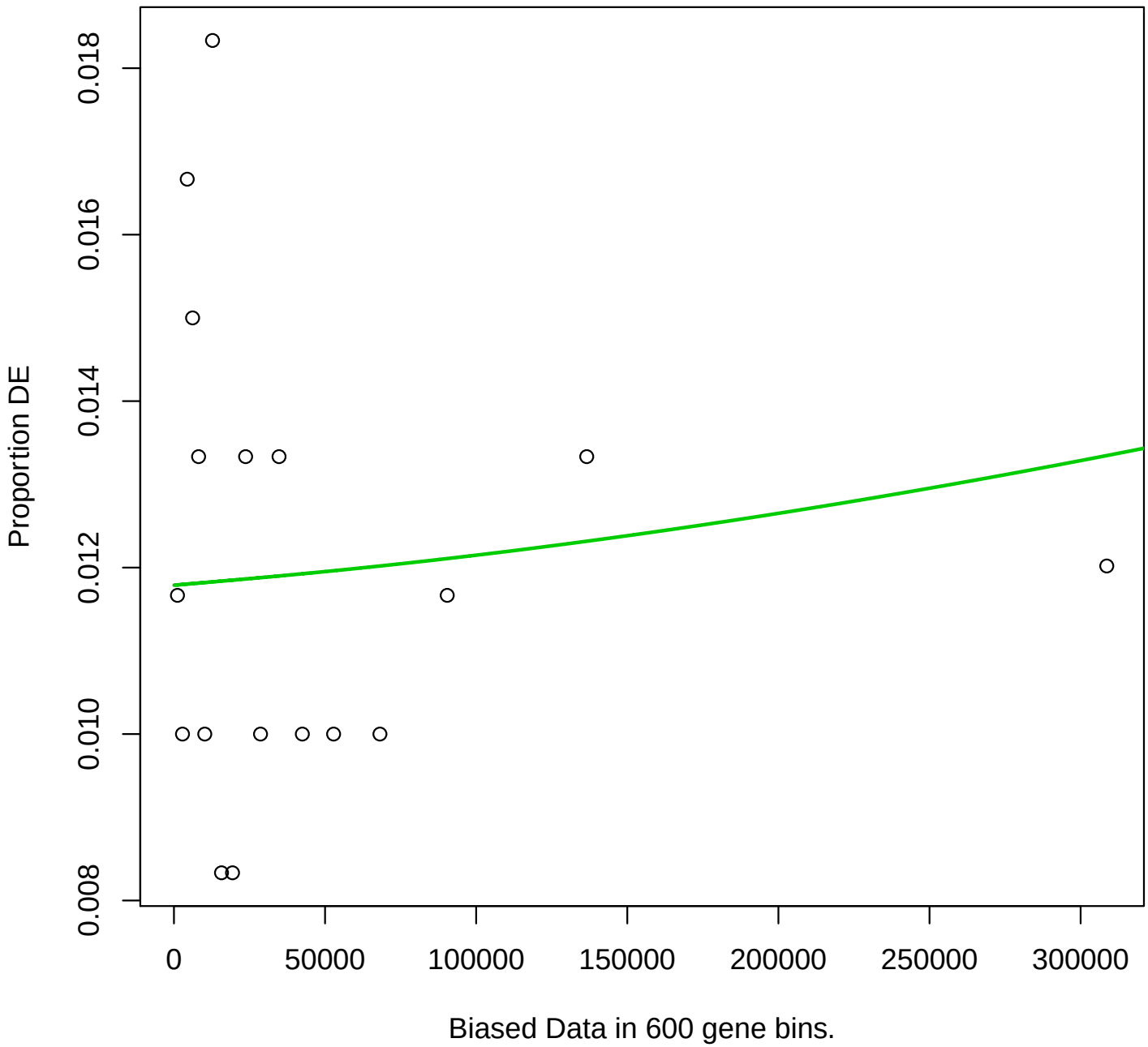
but we lose a few genes associated with a low FDR in the process

```
> table(bovDif1)
bovDif1
    0     1
10489   127
```

goseq

Next we need to estimate the change in power as a function of gene length. Here is a graphical method to visualize this:

```
> pdf("null_p_bov.pdf")
> pwf=nullp(bovDif1,bias.data=bovLen)
Warning message:
In pcls(G) : initial point very close to some inequality
constraints
> dev.off()
```


Proportion DE
0.008
0.010
0.012
0.014
0.016
0.018
0
50000
100000
150000
200000
250000
300000
Biased Data in 600 gene bins.

goseq

Then we use this estimated change in power to test for differences in the frequency of the occurrence of genes that display differences across groups between functional categories.

```
> gs1=goseq(pwf,gene2cat=bov_bm[,c(1,3)])
Using manually entered categories.
For 153 genes, we could not find any categories. These
genes will be excluded. To force their use, please
run with use_genes_without_cat=TRUE (see documentation).
This was the default behavior for version 1.15.1 and
earlier.
Calculating the p-values...
'select()' returned 1:1 mapping between keys and columns
```

goseq

Then the following code can be used to examine the output.

```
> cat=gs1$category[which(p.adjust(
+  gs1$under_represented_pvalue,
+  method="BH")<.1)]
> cat=cat[cat!=""]
> cat_name=rep(NA,length(cat))
> for(i in 1:length(cat))
+  cat_name[i]=unique(bov_bm[
+  bov_bm[,3]==cat[i],4])
> bov_go_under=cat_name
[1] NA
```

goseq

```
> cat=gs1$category[which(p.adjust(
+  gs1$over_represented_pvalue,
+  method="BH")<.1)]
> cat=cat[cat!=""]
> cat_name=rep(NA,length(cat))
> for(i in 1:length(cat))
+  cat_name[i]=unique(bov_bm[
+  bov_bm[,3]==cat[i],4])
> bov_go_over=cat_name
[1] NA
```

goseq

Let's try the approach the authors of Bioconductor Case Studies recommend: don't correct for multiple hypothesis testing (we get 348 categories if we use $\alpha = 0.05$ so we use 0.01 here).

```
> cat=gs1$category[which(gs1$over_represented_pvalue<
+  0.01)]
> cat=cat[cat!=""]
> cat_name=rep(NA,length(cat))
> for(i in 1:length(cat))
+  cat_name[i]=unique(bov_bm[
+  bov_bm[,3]==cat[i],4])
> bov_go_over=cat_name
```

goseq

```
> bov_go_over
 [1] "enzyme inhibitor activity"
 [2] "negative regulation of intestinal phytosterol
    absorption"
 [3] "ATP-binding cassette (ABC) transporter complex"
 [4] "negative regulation of intestinal cholesterol
    absorption"
 [5] "acetyl-CoA carboxylase activity"
 [6] "facial nerve morphogenesis"
 [7] "glycerol metabolic process"
 [8] "acetyl-CoA metabolic process"
...
 42 total
```

goseq

We can also restrict this to a particular component of the ontology with the following:

```
> cat=gs1$category[which(gs1$over_represented_pvalue<
+  0.01)]
> cat=cat[cat!=""]
> cat_name=rep(NA,length(cat))
> ontol=rep(NA,length(cat))
> for(i in 1:length(cat)){
+   cat_name[i]=unique(bov_bm[bov_bm[,3]==cat[i],4])
+   ontol[i]=unique(bov_bm[bov_bm[,3]==cat[i],5])
+ }
> bov_go_over=cat_name
```

goseq

```
> table(ontol)
ontol
biological_process cellular_component molecular_function
                23                   7                  12

> bov_go_over[ontol=="biological_process"]
 [1] "negative regulation of intestinal phytosterol
    absorption"
 [2] "negative regulation of intestinal cholesterol
    absorption"
 [3] "facial nerve morphogenesis"
 [4] "glycerol metabolic process"
 [5] "acetyl-CoA metabolic process"
 [6] "negative regulation of lipoprotein lipase activity"
...
```

goseq

```
> bov_go_over[ontol=="cellular_component"]
[1] "ATP-binding cassette (ABC) transporter complex"
[2] "insulin-like growth factor ternary complex"
[3] "extracellular space"
[4] "extracellular region"
[5] "apical plasma membrane"
[6] "integral component of plasma membrane"
...
```

goseq

```
> bov_go_over[ontol=="molecular_function"]
 [1] "enzyme inhibitor activity"
 [2] "acetyl-CoA carboxylase activity"
 [3] "growth factor activity"
 [4] "biotin carboxylase activity"
 [5] "integrin binding"
 [6] "oxidoreductase activity, acting on paired donors,
  with incorporation or reduction of molecular oxygen"
...
```

Then many of the genes associated with biological processes are sensible given that this is liver tissue from cows in negative energy balance.

Conclusions

While the motivation for the method behind the goseq package may not hold in contemporary applications of RNA-Seq, these results can still be useful.

As an alternative, one can always make a matrix of gene expression data into an ExpressionSet object and use tools for microarrays.

This is an attractive feature in that there are many tools available for microarray analysis.

The primary difference is that one must download information from biomart to link gene identities to functional classes.