Ballgown
flexible RNA-seq differential expression analysis

Alyssa Frazee
Johns Hopkins Biostatistics
@acfrazee
RNA-seq data

Reads
(50-100 bases)

Transcripts
(RNA)

Genome
(DNA)
RNA-seq data analysis

align
RNA-seq data analysis

align

(or not: de novo assembly)
RNA-seq data analysis

align → assemble
RNA-seq data analysis

align → assemble → estimate abundances
RNA-seq data analysis

align → assemble → estimate abundances

differential expression analysis
RNA-seq data analysis

align → assemble → estimate abundances

differential expression analysis

Is transcript X expressed differently in population A vs. population B?
RNA-seq data analysis

- align
- assemble
- estimate abundances

Differential expression analysis

Is transcript X expressed differently in population A vs. population B vs. population C?
RNA-seq data analysis

align → assemble → estimate abundances

Differential expression analysis

Is transcript X expressed constantly over the course of time in the study?
align \rightarrow \text{assemble} \rightarrow \text{estimate abundances}

\text{differential expression analysis}

\text{visualization postprocessing}

organize output
organize output: Tablemaker

top_dir/
sample01/
  e2t.ctab
  e_data.ctab
  i2t.ctab
  i_data.ctab
  t_data.ctab

sample02/
  e2t.ctab
  e_data.ctab
  i2t.ctab
  i_data.ctab
  t_data.ctab

sample03/
  e2t.ctab
  e_data.ctab
  i2t.ctab
  i_data.ctab
  t_data.ctab
S4 class for transcript assemblies

```r
library(ballgown)

my_assembly = ballgown(dataDir='top_dir', samplePattern='sample')
my_assembly
## ballgown instance with 124892 transcripts and 3 samples

my_rsem_assembly = ballgownrsem(dir='top_dir', samples=c('sample01', 'sample02', 'sample03'),
                                 gtf='hg19_genes.gtf')
my_rsem_assembly
## ballgown instance with 45914 transcripts and 3 samples
```
S4 class for transcript assemblies
convenient visualization

plotTranscripts(gene='XLOC_000454', gown=my_assembly, samples='sample12', meas='FPKM', colorby='transcript',
main='transcripts from gene XLOC_000454: sample 12, FPKM')
fast, flexible statistical analysis

```r
stat_results = stattest(my_assembly, feature='transcript', meas='FPKM', covariate='group')
head(stat_results)
## feature     id   pval       qval
## transcript  10 0.0138158  0.10521233
## transcript  25 0.2677362  0.79114975
## transcript  35 0.0108507  0.08951825
## transcript  41 0.4710802  0.90253748
## transcript  45 0.0840295  0.48934814
## transcript  67 0.2731739  0.79114975
```
drop-in replacement for Cuffdiff
**fast, flexible statistical analysis**

```r
stat_results = stattest(my_assembly, feature='transcript', meas='FPKM', covariate='group')
head(stat_results)

## feature     id     pval       qval
## transcript 10 0.01381576 0.105212332
## transcript 25 0.26773622 0.791149753
## transcript 35 0.01085070 0.089518254
## transcript 41 0.47108019 0.902537475
## transcript 45 0.08402948 0.489348136
## transcript 67 0.27317385 0.79114975
```

drop-in replacement for Cuffdiff

bonus: handles more experimental designs!
models expression over time as smooth curve; compares to model where expression is assumed constant over time
custom models

\begin{verbatim}
stat_results = stattest(my_assembly, feature='transcript',
meas='FPKM', covariate='group')

timecourse_results = stattest(my_assembly,
feature='transcript', meas='FPKM', covariate='time',
timecourse=TRUE)

mod = model.matrix(~ time + library_size)
mod0 = model.matrix(~ library_size)
my_timecourse_results = stattest(my_assembly,
feature='transcript', meas='FPKM', mod=mod, mod0=mod0)
\end{verbatim}
easy integration with other packages

```r
stat_results = stattest(my_assembly, feature='transcript', meas='FPKM', covariate='group')

timecourse_results = stattest(my_assembly, feature='transcript', meas='FPKM', covariate='time', timecourse=TRUE)

mod = model.matrix(~ time + library_size)
mod0 = model.matrix(~ library_size)
my_timecourse_results = stattest(my_assembly, feature='transcript', meas='FPKM', mod=mod, mod0=mod0)

library(limma)
expression_mat = log2(texpr(my_assembly, 'FPKM') + 1)
fit = lmFit(y, mod)
fit = eBayes(fit, trend=TRUE)
```
wrangling large datasets
reanalysis of GEUVADIS data: eQTL

(a) Chr1:64193406–64195282

Genomic position

(b) Copies of MA for SNP Chr1:Pos 64122505

FPKM
wrangling large datasets

reanalysis / processing of GEUVADIS data

Ballgown objects: figshare

TopHat read alignments: ArrayExpress (E-GEUV-6)

analysis code: GitHub,
https://github.com/alyssafrazee/ballgown_code
thanks!

Collaborators: Jeff Leek (advisor), Geo Pertea, Andrew Jaffe, Ben Langmead, Steven Salzberg

References:
manuscript: http://biorxiv.org/content/early/2014/03/30/003665
Cufflinks: PMID 20436464 (Trapnell et al 2010)
RSEM: PMID 21816040 (Li & Dewey 2011)

Software: Ballgown is under Bioconductor review; currently available and fully documented at https://github.com/alyssafrazee/ballgown
differential expression model

for each transcript, compare the fits of the following models using an F-test. Null hypothesis is that the fits of model (a) and model (b) are equally good; alternative is that (a) fits better.

(a) expression\(_i\) = \(\alpha + \beta_0\) \text{group}_i + \sum_{p=1}^{P} \gamma_p \text{confounder}_{ip} + \text{noise}_{ip}

(b) expression\(_i\) = \(\alpha^* + \sum_{p=1}^{P} \gamma^*_p \text{confounder}_{ip} + \text{noise}^*_ip\)