



Deeper differential expression analysis with shrinkage correction

Jared Brown

Postdoctoral research fellow

Lab of Rafael Irizarry · DFCI Data Science

21 May 2024

Follow-along preliminaries



Software:

- R: <https://cran.r-project.org/>
- Rstudio: <https://posit.co/download/rstudio-desktop/>
- R packages:

```
install.packages("BiocManager")  
BiocManager::install(c("DESeq2", "pasilla",...))  
DESeq2, pasilla, ggplot2, matrixStats, apeglm, ashR, TENxPBMCDATA, Matrix,  
    irlba, scran, scater
```

Analysis scripts:

- https://github.com/JBrownBiostat/DifferentialExpressionTraining_May2024

From the original developers, Michael Love, Simon Anders, Wolfgang Huber:
<https://bioconductor.org/packages/release/bioc/vignettes/DESeq2/inst/doc/DESeq2.html>

General (single-cell RNA) pipelines from some greats in the field:
<https://bioconductor.org/books/3.17/OSCA/>

Carpentry workshop from Bioconductor:
<https://carpentries-incubator.github.io/bioc-rnaseq/>

DE definitions: data structure

A $G \times J$ matrix of abundance measures across:

- G many features: **genes**, transcripts, exons, protein binding peaks, methylation sites, etc.
- J many samples: **experimental libraries**, single cells, spatial spots, binned pixels, etc.
- First few lines from the *pasilla* dataset of pasilla gene knock-down in drosophila melanogaster

	Samp1	Samp2	Samp3	Samp4	Samp5	Samp6	Samp7
FBgn0000003	0	0	0	0	0	0	1
FBgn0000008	92	161	76	70	140	88	70
FBgn0000014	5	1	0	0	4	0	0
FBgn0000015	0	2	1	2	1	0	0
FBgn0000017	4664	8714	3564	3150	6205	3072	3334
FBgn0000018	583	761	245	310	722	299	308
FBgn0000022	0	1	0	0	0	0	0
FBgn0000024	10	11	3	3	10	7	5
FBgn0000028	0	1	0	0	0	1	1
FBgn0000032	1446	1713	615	672	1698	696	767



Switch over to `bulkRNA_AnalysisScript.R`

DE definitions: what are we testing

Most generally:

- Is the effect size of a given combination of covariates significantly non-zero

In practice:

- Is the abundance of gene g in condition A significantly different than in condition B controlling for appropriate nuisance variation

Implied comparison of interest:

- Ratios measured as log (2) fold-changes

DE definitions: typical model spec.

DESeq2 (and other models) assume observed expression is effectively modeled as a *Negative Binomial GLM*, i.e.:

$$y_{gj} \sim NB(\mu_{gj}, \phi_g)$$
$$\log(\mu_{gj}) := [X\vec{\beta}]_{gj} + \log(s_j)$$

For observed counts y_{gj} , design matrix X , coefficient vector $\vec{\beta}$, normalization factor/offset s_j , and dispersion parameter ϕ_g .

NB GLM: Comp. to linear regression



Traditional “simple linear regression” can be written as a Normal GLM:

$$y_{gj} = [X\vec{\beta}]_{gj} + \epsilon_{gj} \text{ s.t. } \epsilon_{gj} \sim N(0, \sigma_g^2)$$

is equivalent to:

$$\begin{aligned} y_{gj} &\sim N(\mu_{gj}, \sigma_g^2) \\ \mu_{gj} &:= [X\vec{\beta}]_{gj} \end{aligned}$$

Identity link (Normal):

$$\mu_{gj} = [X\vec{\beta}]_{gj} = x_{g0}\beta_0 + x_{g1}\beta_1 \cdots + x_{gK}\beta_K$$

Log link (Negative Binomial):

$$\begin{aligned}\log(\mu_{gj}) &= [X\vec{\beta}]_{gj} + \log(s_j) \\ \Rightarrow \mu_{gj} &= (e^{x_{g0}\beta_0} e^{x_{g1}\beta_1} \cdots e^{x_{gK}\beta_K}) s_j\end{aligned}$$

Action of log link indicates data Y should be raw expression and not normalized expression

(Pre) Calculation of normalization offset



In DESeq2 (and other methods) normalization factors s_j are offsets:

- s_j are calculated prior to model estimation (fixed)
- s_j have an implicit fixed coefficient $\beta_s = 1$

Popular norm. factor calculation methods include:

1. Reads-per-kilobase-million (RPKM)
2. Transcripts/Counts-per-million (TPM/CPM)
3. Library-size normalization
4. Trimmed-mean-of-m-values (TMM, default in edgeR)

DESeq2 normalization default



Method of *Median of Ratios* and *Scran*:

- Median of Ratios (MR) from Anders and Huber 2010
- Scran based on Lun, Bach, and Marioni 2016

Both methods based on the median of ratios relationship:

$$\hat{s}_j := \text{median}_g \left\{ \frac{\hat{y}_{gj}}{\left(\prod_g^G \hat{y}_{gj} \right)^{\frac{1}{G}}} \right\} = \text{median}_g \left\{ \frac{\hat{y}_{gj}}{\exp \left[\frac{1}{G} \sum_g^G \log(\hat{y}_{gj}) \right]} \right\}$$

For MR, $\hat{y}_{gj} = y_{gj}$ and $s_j = \hat{s}_j$

For Scran, \hat{y}_{gj} is based on a pooling of cells and s_j is derived from a deconvolution of (multiple) estimates \hat{s}_j that are based on sample j



Switch over to `bulkRNA_AnalysisScript.R`

NB GLM: Dispersion estimates



Dispersion is analogous to variance, specifically:

$$y \sim NB(\mu, \phi) \Rightarrow \mathbb{E}[y] = \mu; \mathbb{V}[y] = \mu + \phi\mu^2$$

In typical bulk experiments, replicate counts are low:

- High standard error on fitted ϕ
- Reduced power to identify significant β_{gk}

So pool information across genes with similar expression levels...

Note: MLE of ϕ only exists when sample variance is greater than sample mean...



Switch over to `bulkRNA_AnalysisScript.R`

Standard multiple-regression is encoded in the design matrix X as:

$$\log(\mu_{gj}) = [X\vec{\beta}]_{gj} + \log(s_j)$$

- Factors encoded in usual “dummy variable” format
- Default model specification chooses first factor as intercept
- For three group model, encoded as:

$$\log(\mu_{gj}) = x_{g1}\beta_1 + x_{g2}\beta_2 + x_{g3}\beta_3 = \beta_1 \mid j \in \mathcal{A}$$

$$\log(\mu_{gj}) = x_{g1}\beta_1 + x_{g2}\beta_2 + x_{g3}\beta_3 = \beta_1 + \beta_2 \mid j \in \mathcal{B}$$

$$\log(\mu_{gj}) = x_{g1}\beta_1 + x_{g2}\beta_2 + x_{g3}\beta_3 = \beta_1 + \beta_3 \mid j \in \mathcal{C}$$

Contrasts and complex comparisons

We can use “contrasts” or *linear combinations of coefficients* to test non-default or complex hypotheses

- Typical DESeq2 reduces general *contrasts* to the difference between the sums of two groups of coefficients (general linear combination possible):

$$c = \sum_{k \in \mathcal{C}_1} \beta_k - \sum_{k \in \mathcal{C}_2} \beta_k$$
$$H_0: c = 0 \quad H_1: c \neq 0$$

- Equivalent to testing whether fold-change (ratio) between groups is different from 1; using groups from before:

$$\frac{\mu_{gB}}{\mu_{gC}} = \frac{e^{\beta_1 + \beta_2}}{e^{\beta_1 + \beta_3}} = e^{(\beta_1 + \beta_2) - (\beta_1 + \beta_3)} = e^{\beta_2 - \beta_3} \Rightarrow c = \beta_2 - \beta_3$$



Switch over to `bulkRNA_AnalysisScript.R`

Shrinkage and calling sig. features



Goals:

- Maximize power
- Maintain control on FDR (or something like it...)
- Improve interpretability/reliability

Note 1: traditional methods for controlling FDR tend to be under-powered

Note 2: leveraging extra information can boost power

q-value and local false discovery rate



q-value (Corollary 2, J. Storey 2003) for statistic t and rejection region Γ_α :

$$q(t) := \inf_{\{\Gamma_\alpha: t \in \Gamma_\alpha\}} \mathbb{P}(H = 0 | T \in \Gamma_\alpha)$$

lfdr for a typical null hypothesis ($\beta_j = 0$):

$$lfdr_j := \mathbb{P}(\beta_j = 0 | \hat{\beta}_j, \hat{s}_j, \hat{\pi})$$

For a particular set of observed effect sizes, ordered by lfdr,

$$q(\hat{\beta}_{(j)}) = \frac{1}{j} \sum_{i=1}^j lfdr_{(i)}$$

Local false sign rate and s-value



lfsr (Eqn. 2.7 from M. Stephens 2017):

$$lfsr_j := \min[\mathbb{P}(\beta_j \geq 0 | \hat{\pi}, \hat{\beta}, s), \mathbb{P}(\beta_j \leq 0 | \hat{\pi}, \hat{\beta}, s)]$$

Tukey (1991): “All we know about the world teaches us that the effects of A and B are always different – in some decimal place – for any A and B”

For a particular set of observed effect sizes, ordered by lfsr,

$$s(\hat{\beta}_{(j)}) = \frac{1}{j} \sum_{i=1}^j lfsr_{(i)}$$

In practice lfsr can be more powerful than lfdr in the sense that calculated lfsr is closer to the true lfsr while still being conservative

Adaptive SHrinkage (ASH) model (1)



Model from *M. Stephens 2017*

Consider *true* effects (model coefficients) $\beta = (\beta_1, \dots, \beta_J)$ and observed effect sizes $\hat{\beta} = (\hat{\beta}_1, \dots, \hat{\beta}_J)$ with corresponding estimated standard errors $\hat{s} = (\hat{s}_1, \dots, \hat{s}_J)$

We can conduct hypothesis testing based on the model of the true effects:

$$\mathbb{P}(\beta | \hat{\beta}, \hat{s}) \propto \mathbb{P}(\beta | \hat{s}) \mathbb{P}(\hat{\beta} | \beta, \hat{s})$$

Model the conditional distribution on true effects and observed effects as:

$$\begin{aligned}\mathbb{P}(\beta | \hat{s}, \pi) &= \prod_j g(\beta_j | \pi) \\ g(\cdot | \pi) &= \pi_0 \delta_0(\cdot) + \sum_{k=1}^K \pi_k N(\cdot | 0, \sigma_k^2) \\ \mathbb{P}(\hat{\beta} | \beta, \hat{s}) &= \prod_j N(\hat{\beta}_j | \beta_j, \hat{s}_j^2)\end{aligned}$$

Adaptive SHrinkage (ASH) model (2)



Integrating out β reveals a convolution of normal such that:

$$\mathbb{P}(\hat{\beta}|\hat{s}, \pi) = \prod_j \left[\pi_0 N(\hat{\beta}_j | 0, \hat{s}_j^2) + \sum_{k=1}^K \pi_k N(\hat{\beta}_j | 0, \sigma_k^2 + \hat{s}_j^2) \right]$$

Note 1: π_0 is a direct estimate of the proportion of null effects

Note 2: in practice optimization penalizes to prefer large π_0

Adaptive Shrinkage (ASH) model (3)



Estimates of π give a closed form distribution for the posterior on β :

$$\mathbb{P}(\beta | \hat{\beta}, \hat{s}, \pi) \propto \prod_j \left[\pi_0 \delta_0(\beta_j) + \sum_{k=1}^K \pi_k N(\beta_j | 0, \sigma_k^2) \right] N(\hat{\beta}_j | \beta_j, \hat{s}_j^2)$$

- Posterior *mean* yields shrunk estimate of effect size
- Posterior tails and point mass at 0 give lfsr/s-values



Switch over to `bulkRNA_AnalysisScript.R`

Some considerations for single cell

Data sparsity causes normalization problems

- Use `scrn`

Data sparsity also affects lower bound on fitted dispersion parameters

- Set `minmu = 1e-6`

High sample (cell) counts trigger outlier correction

- Set `minReplicatesForReplace = Inf`