RNA-seq Differential Expression (DE) Analysis Using baySeq

Jessica Randall

Briefly, baySeq uses empirical Bayesian inference to determine the likelihood that genes in compared samples are indeed DE genes. baySeq improve seeks to improve accuracy in DE estimation over other popular packages by using the underlying structure of the data itself. baySeq does show improved performance in the case of more complex study designs (i.e. multiple group comparisons) and in studies with large numbers of libraries compared to other popular packages. Here we illustrate a simplified example of a comparison between a control and a treatment group.

Linked is the original paper from Hardcastle introducing the concepts implemented in baySeq.

We will be using the pasilla package for our example data.

baySeq has wide range of applications to genomic analyses. We strongly encourage you to reach out to EICC with questions regarding options available to you with baySeq. Check out some of the graphs from previous projects here.

Definition of terms

0.0.0.1 * Bayesian Statistics:

Learning from experience to make inferences about the relationship between our variables of interest. While the null hypothesis in the frequentist approach (i.e. the use of p values) says that "there is no relationship", the Bayesian approach allows us to incorporate previous knowledge we may have about a relationship into future research we want to do on it. The associated credible interval (as compared to the confidence interval) tells us how likely it is that a particular value of interest lies between the prior and posterior estimates of that value. In our example we would interpret this as, given what we know about genetic expression between our control and treatment groups, how likely it is that a certain percentage of genes are DE genes?

0.0.0.1.1 Prior probability

The information we have about our data before we see it. This could come from what we know about similar experiments or we can estimate it from our data and these options vary in their utility. In our example, this could refer to how likely we think it is that a certain percentage of genes are DE between our control and treatment groups.

0.0.0.1.2 Posterior probability

The results of incorproating our prior knowledge and our observed data. In our case that means, given that our control and treatment samples have particular genes with particular amounts of expression, how likely it is that a certain percentage of genes between them are DE genes?

For more information and the source for these definitions please see "A Gentle Introduction to Bayesian Analysis" from Schoot, Kaplan, Denissen, Asendorpf, Neyer, and Aken (Developmental Method, 2013) here

0.0.0.2 * Unadjusted p values vs adjusted p-values/(FDR):

In DE analysis, a single p-value tells you how likely it is that a single gene is differential expressed between at least two groups (ex: a control and a treatment group) due to some actual difference between the groups as opposed to random chance. False Discovery Rate (FDR) tells you how likely it is that all genes identified as DE are false positives. A FDR of 5% means that among all genes called DE, an average of 5% of those are truly not DE. DE genes are only considered significantly so if they meet the adjusted p value, not only the unadjusted p-value. FDRs for each individual gene are called q-values or local FDRs.

Loading data

Our very first step is to load the libraries we'll need to assess the functions required for analysis and graphing. Please see Bioconductor for information about initial installation and use of Bioconductor and its packages. We also set the minimal theme in gglot2 for all graphs to have the same aesthetic features by default.

The pasilla experiment studied RNAi knockdown of Pasilla, the Drosophila melanogaster ortholog of mammalian NOVA1 and NOVA2, on the transcriptome. Data are provided by NCBI Gene Expression Omnibus under accession numbers GSM461176 to GSM461181.

Here we will demonstrate importing the count matrix and sample data from the pasilla package since we're using it as an example. Typically we will use the here package to specify the path for the counts and sample data files in a list of files to import and export from the task.

We're also going to specify that we'd like the row names of our sample data to come from the first column, called "file" since this is where we've stored which sample is which and finally we remove extra columns from our sample data which we won't be using in our analysis.

Please reach out to EICC if you would like to compare 3 or more groups as this is a simplified example. It may also be the case you will need more than 6 samples per experimental group or that you may need to remove genes with average counts greater than 5, 10, 15, or even 20 for sufficient statistical power. Please see our PROPER walk-through for an example of our of power and sample size analysis.

Preparing for Analysis

In order to perform our pairwise comparison we need to specify some informaton about our data. Replicates and Groups represent the labels for our replicates (Control vs. Treatment) and the null (NDE) and alternative (DE) hypotheses.

The groups object specifies our hypothesis. In the null hypothesis of no differential expression (NDE), if all of our samples are 1's, this means that all of the samples belonging to the same group, there is no difference between the treated and untreated samples. In the alternative hypothesis of differential expression, (DE) 1's represent membership in a single group and 2's represent membership in a second group. This set of 4 1's and 3 2's says that there exist two distinct patterns of expression in the control and treatment groups.

We also annotate all 13064 genes with a number to cross-reference with the list of gene names and counts after analysis. Our final preparation step is to get our library size. Sometimes you may already know this but we have inferred it from the data.

```
groups<- list(
  NDE = c("untreated","untreated","untreated","untreated","untreated",
  "untreated", "untreated"),
  DE = c("untreated","untreated","untreated","untreated","treated",
       "treated", "treated"))

CD <- new("countData",
          data = countdata,
```

```
          replicates = sampledata$condition,
          groups = groups)

libsizes(CD) <- getLibsizes(CD)

CD@annotation <-data.frame(name = rownames(countdata))
```

Determine Prior Probabilities

We determine the prior probabilities of DE from the count data. Recall that this means we are incorporating some structure inherent in our data to inform the likeliehood that there are DE genes between our two groups.

For our simplified example we have chosen to do 10000 bootstrap samples using quasi-likelihood estimation.These parameters will vary by experiment and we would tailor these to your specific project.The length of time that determining the priors and posterior likelihoods will take depends on your computational power, your sample size, the method of estimation, the number of genes, number of samples, and the number of groups you are comparing.

```
if(require("parallel")) cl <- makeCluster(8) else cl <- NULL

start_time <- Sys.time()

CD <- getPriors.NB(CD, samplesize = 10000, estimation = "QL", cl = cl)

end_time <- Sys.time()

end_time - start_time
```

```
## Time difference of 1.440237 mins
```

Determine Posterior Likelihoods

Next, we determine the posterior likelihood, that is, we are using our prior knowledge and looking at our existing data to estimate how likely is it that there are DE genes between our two groups?

We chose to use just 3 bootstrap estimates here in the interest of the accuracy vs. time trade-off. We would assess this parameter based on your specific experiment.

```
start_time <- Sys.time()

CD <- getLikelihoods(CD, bootStraps = 3, verbose = FALSE, cl = cl)
```

```
## ...
```

```
end_time <- Sys.time()

end_time - start_time
```

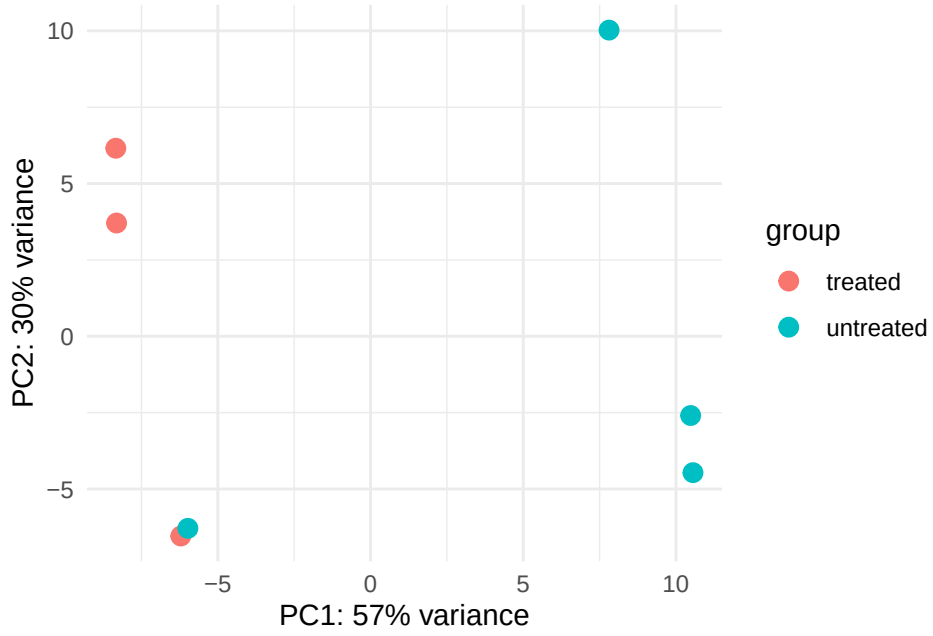
```
## Time difference of 3.708142 mins
```

If we had any additional data to add about the samples that we wanted to include in our analysis we would add it next but since this is a simplified example, we are only comparing treated and control samples without taking into account any additional information about them.

At this point we generate our first exploratory visualization, the principal components analysis plot. This will show us how your data cluster or how similar each sample is to others of the same group. There are percentages along the axes and the percentage on the x-axis tells us how much the differences between the samples is explained by them being treated or untreated.

Since baySeq does not have a built-in function for PCA plots, we can import the data into the file structure preferred by DESEq2 and use their built-in function. Please note that we are only using this data structure to visualize data from baySeq, we are not using DESeq's tests for differential expression. For that, please see our DESeq2 walktrhough here. As we do in that tutorial's PCA plot, we first transform our data using the variance stabilizing transformation available from the vsn library (Tibshirani 1988; Huber et al. 2003; Anders and Huber 2010). This is similar to using a log2 transformation with normally distributed data with many very small values. VST adjust the data such that if the means of the rows are small, as they often are in gene counts, the variance will remain relatively constant across all all counts. Doing this allows the user to cluster the samples into experimentally interesting groups in graphs rather than seeing groups clustered by their variance. We then typically save this as a data frame to export to clients.

We can use the same plotPCA function to obtain the coordinates for each sample on the plot. This is helpful in identifying samples we would consider outliers since we haven't labelled each sample on the graph.



We would interpret this as the samples being pretty clearly by group and interpret the percentage on the x-axis as 57% of the variability between these samples is due to them being treated or untreated. The y-axis

tells us how much variability between these samples is due to other factors in our model or if we have none, sources of variability we may not have accounted for like sex or ethnicity which are often leading contributors of variability between samples and should be accounted for in experimental design if you wish to control for their effects.

In the lower left corner of the graph we see that two samples from the treated and untreated groups are clustering. This may suggest that these two samples are too similar to one another for us to distinguish between-group difference.

Results

Here we have the list of top 10 genes which are most likely to be differentially expressed at FDR controlled at 0.05. This cut-off may vary anywhere from 0.01-0.2 and will depend on the individual experiment but should be specified a priori. We can see that the very first gene, FBgn002511 is significantly up regulated in the treated as compared to the untreated samples whereas FBgn0039155 is significantly up regulated in the untreated samples as compared to the treated samples.

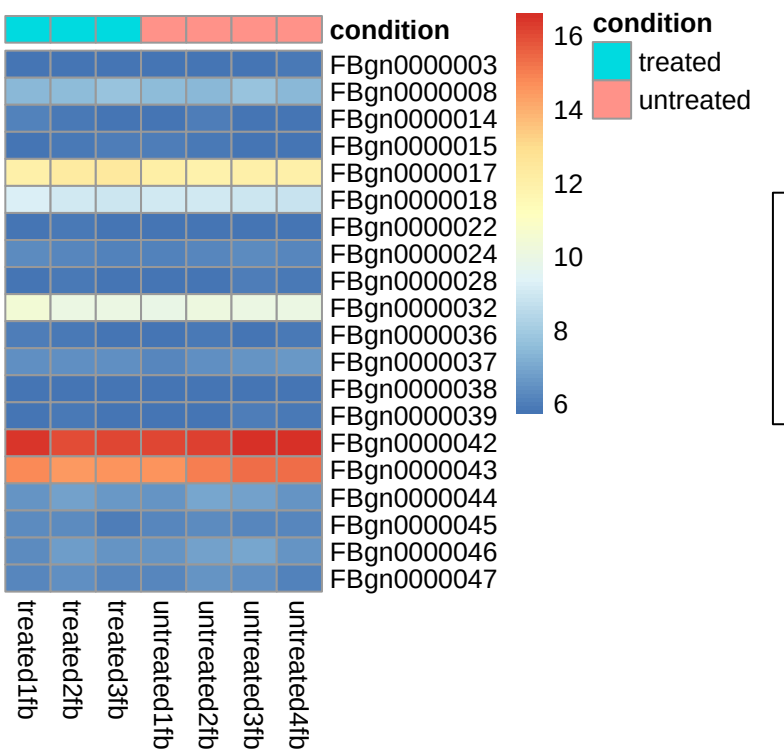
FDR thresholds can also range from 0.05-0.2 and it is much better to have the option to lower your threshold during analysis rather than have to increase it because your study was not sufficiently powered to find anything but the most extremely highly or lowly differentially expressed genes. Please reach out to us at EICC if you would like assistance in planning your experimental design for your RNAseq project and in setting appropriate FDR thresholds.

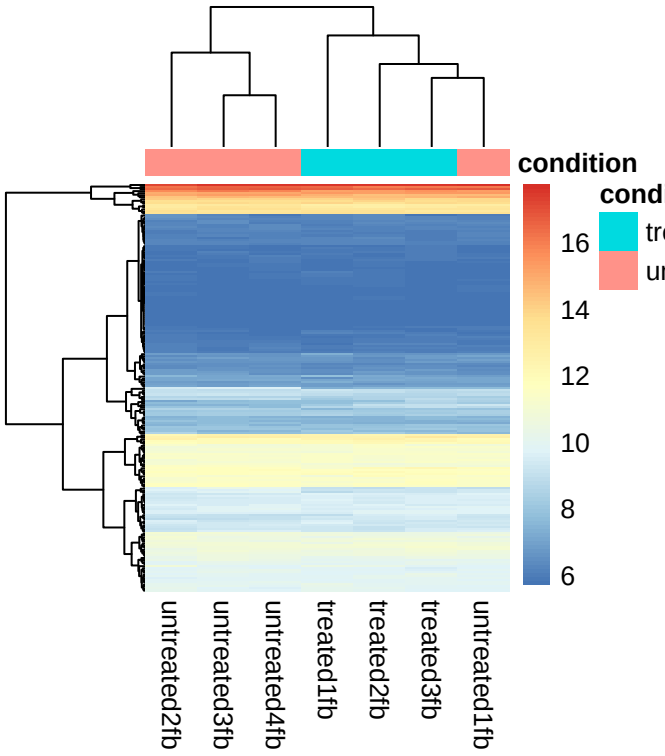
Finally, we create a data frame sorted by adjusted FDR of the DE group. We check that the data frame was created successfully by using the informal unit test of dimension with the expected number of rows and columns and telling the program to stop if the file does not have these dimensions. After this runs successfully we typically export them as a .csv file for you.

Visualizing

We now perform additional data visualizations. Typically we provide a PCA plot, heat maps, and volcano plots. We would be happy to work with you to customize these for publication. Please see our Data Visualization menu for more options and examples from previous projects.

Before we generate our heatmap in baySeq, we need to transform the count. In this example we have decided to sort the DE genes by smallest adjusted p value and only examine the top 20 DE genes. The number of DE genes you may want to visualize is customizable based on your project with EICC.

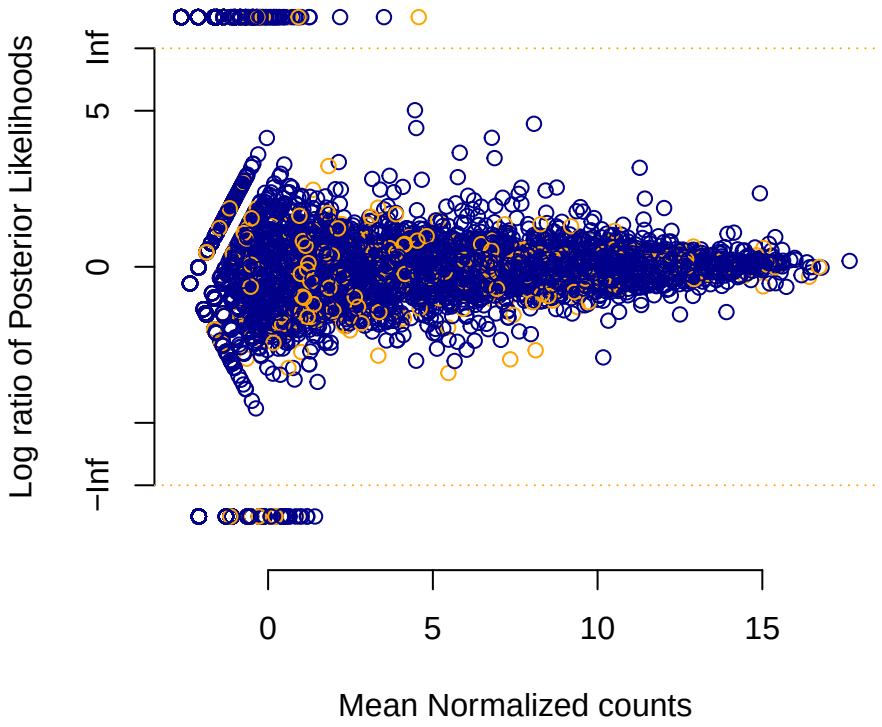




Please note that these are sorted for convenience but the gene at the top of the list is no more significant than the gene at the bottom of the list. As is the case with nominal p-values, a smaller adjusted p-value does not make a gene more statistically significant than one with a larger adjusted p-value. If the genes are below the threshold, they are all equally statistically significantly differential expressed. These are sorted for convenience but the gene at the top of the list is no more significant than the gene at the bottom of the list.

When using edgeR, DESeq2, or other frequentist based packages, we also typically provide clients with an initial volcano plot of the log(2) fold changes by the adjusted p-values created with the EnhancedVolcano R library. Similar to the PCA plot and heat map, volcano plots are highly customizable graph and we would like to work with you to design graphs which best tell the story of your results.

bayseq provides a built-in plotMA function to graph the mean normalized counts and the log ratio of the Posterior Likelihoods of differential expression. This shows us the distribution of the likelihoods given the counts of each gene. Blue circles indicate genes not determined to be differentially expressed while orange genes indicate those which are differentially expressed at the FDR threshold of 0.05. From this graph we see that as expected, most genes are not differentially expressed and those genes which are, have all levels of counts. This graph is more of a quality control check on the analysis than something we would suggest for a publication.



There are many more functions and many more specifications to functions than are used here in order to show a simplified example of one of the tools we use for differential expression analysis. Obtaining specific, actionable, and publication quality results from analysis requires a deeper understanding of your specific data set and we would love the opportunity to discuss these options with you.

While we encourage clients to reach out prior to sequencing so that we can collaborate to design the experiment to answer your specific questions, we look forward to hearing from you at any stage of your RNA-seq project. Please find our contact information available on our website and check out some of the graphs we've made for previous clients here.

Session information and References

```
## [1] "Wed Mar 25 20:03:08 2020"

## R version 3.6.3 (2020-02-29)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Debian GNU/Linux 9 (stretch)
##
## Matrix products: default
## BLAS:   /usr/lib/openblas-base/libblas.so.3
## LAPACK: /usr/lib/libopenblasp-r0.2.19.so
##
## locale:
```

```
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] parallel  stats4    stats     graphics  grDevices utils     datasets
## [8] methods   base
##
## other attached packages:
##  [1] tinytex_0.20                random_0.2.6
##  [3] pheatmap_1.0.12             EnhancedVolcano_1.4.0
##  [5] ggrepel_0.8.2               ggplot2_3.3.0
##  [7] vsn_3.54.0                  DESeq2_1.26.0
##  [9] SummarizedExperiment_1.16.1 DelayedArray_0.12.2
## [11] BiocParallel_1.20.1         matrixStats_0.56.0
## [13] Biobase_2.46.0              baySeq_2.20.0
## [15] abind_1.4-5                 GenomicRanges_1.38.0
## [17] GenomeInfoDb_1.22.0         IRanges_2.20.2
## [19] S4Vectors_0.24.3            BiocGenerics_0.32.0
## [21] tidyr_1.0.2                 knitr_1.28
## [23] dplyr_0.8.5                 here_0.1
## [25] readr_1.3.1                 rmarkdown_2.1
##
## loaded via a namespace (and not attached):
##  [1] bitops_1.0-6           bit64_0.9-7            RColorBrewer_1.1-2
##  [4] rprojroot_1.3-2        tools_3.6.3            backports_1.1.5
##  [7] R6_2.4.1               affyio_1.56.0          rpart_4.1-15
## [10] Hmisc_4.3-1            DBI_1.1.0              colorspace_1.4-1
## [13] nnet_7.3-13            withr_2.1.2            tidyselect_1.0.0
## [16] gridExtra_2.3          curl_4.3               preprocessCore_1.48.0
## [19] bit_1.1-15.2           compiler_3.6.3         pacman_0.5.1
## [22] htmlTable_1.13.3       labeling_0.3           bookdown_0.18
## [25] scales_1.1.0           checkmate_2.0.0        genefilter_1.68.0
## [28] affy_1.64.0            stringr_1.4.0          digest_0.6.25
## [31] foreign_0.8-76         XVector_0.26.0         base64enc_0.1-3
## [34] jpeg_0.1-8.1           pkgconfig_2.0.3        htmltools_0.4.0
## [37] limma_3.42.2           htmlwidgets_1.5.1      rlang_0.4.5
## [40] rstudioapi_0.11        RSQLite_2.2.0          farver_2.0.3
## [43] acepack_1.4.1          RCurl_1.98-1.1         magrittr_1.5
## [46] GenomeInfoDbData_1.2.2 Formula_1.2-3          Matrix_1.2-18
## [49] Rcpp_1.0.3             munsell_0.5.0          lifecycle_0.2.0
## [52] stringi_1.4.6          yaml_2.2.1             edgeR_3.28.1
## [55] zlibbioc_1.32.0        grid_3.6.3             blob_1.2.1
## [58] crayon_1.3.4           lattice_0.20-40        splines_3.6.3
## [61] annotate_1.64.0        hms_0.5.3              locfit_1.5-9.1
## [64] pillar_1.4.3           geneplotter_1.64.0     XML_3.99-0.3
## [67] glue_1.3.2             evaluate_0.14          latticeExtra_0.6-29
## [70] BiocManager_1.30.10    data.table_1.12.8      png_0.1-7
## [73] vctrs_0.2.4            gtable_0.3.0           purrr_0.3.3
## [76] assertthat_0.2.1       xfun_0.12              xtable_1.8-4
## [79] survival_3.1-11        tibble_2.1.3           AnnotationDbi_1.48.0
```

```
## [82] memoise_1.1.0          cluster_2.1.0

##
## To cite the 'bookdown' package in publications use:
##
##   Yihui Xie (2020). bookdown: Authoring Books and Technical Documents
##   with R Markdown. R package version 0.18.
##
##   Yihui Xie (2016). bookdown: Authoring Books and Technical Documents
##   with R Markdown. Chapman and Hall/CRC. ISBN 978-1138700109
##
## To see these entries in BibTeX format, use 'print(<citation>,
## bibtex=TRUE)', 'toBibtex(.)', or set
## 'options(citation.bibtex.max=999)'.

##
## To cite package 'readr' in publications use:
##
##   Hadley Wickham, Jim Hester and Romain Francois (2018). readr: Read
##   Rectangular Text Data. R package version 1.3.1.
##   https://CRAN.R-project.org/package=readr
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {readr: Read Rectangular Text Data},
##     author = {Hadley Wickham and Jim Hester and Romain Francois},
##     year = {2018},
##     note = {R package version 1.3.1},
##     url = {https://CRAN.R-project.org/package=readr},
##   }

##
## To cite package 'abind' in publications use:
##
##   Tony Plate and Richard Heiberger (2016). abind: Combine
##   Multidimensional Arrays. R package version 1.4-5.
##   https://CRAN.R-project.org/package=abind
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {abind: Combine Multidimensional Arrays},
##     author = {Tony Plate and Richard Heiberger},
##     year = {2016},
##     note = {R package version 1.4-5},
##     url = {https://CRAN.R-project.org/package=abind},
##   }
##
## ATTENTION: This citation information has been auto-generated from the
## package DESCRIPTION file and may need manual editing, see
## 'help("citation")'.

##
```

```
## To cite package 'tidyr' in publications use:
##
##   Hadley Wickham and Lionel Henry (2020). tidyr: Tidy Messy Data. R
##   package version 1.0.2. https://CRAN.R-project.org/package=tidyr
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {tidyr: Tidy Messy Data},
##     author = {Hadley Wickham and Lionel Henry},
##     year = {2020},
##     note = {R package version 1.0.2},
##     url = {https://CRAN.R-project.org/package=tidyr},
##   }

##
## To cite package 'dplyr' in publications use:
##
##   Hadley Wickham, Romain François, Lionel Henry and Kirill Müller
##   (2020). dplyr: A Grammar of Data Manipulation. R package version
##   0.8.5. https://CRAN.R-project.org/package=dplyr
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {dplyr: A Grammar of Data Manipulation},
##     author = {Hadley Wickham and Romain François and Lionel Henry and Kirill Müller},
##     year = {2020},
##     note = {R package version 0.8.5},
##     url = {https://CRAN.R-project.org/package=dplyr},
##   }

##
## To cite the 'knitr' package in publications use:
##
##   Yihui Xie (2020). knitr: A General-Purpose Package for Dynamic Report
##   Generation in R. R package version 1.28.
##
##   Yihui Xie (2015) Dynamic Documents with R and knitr. 2nd edition.
##   Chapman and Hall/CRC. ISBN 978-1498716963
##
##   Yihui Xie (2014) knitr: A Comprehensive Tool for Reproducible
##   Research in R. In Victoria Stodden, Friedrich Leisch and Roger D.
##   Peng, editors, Implementing Reproducible Computational Research.
##   Chapman and Hall/CRC. ISBN 978-1466561595
##
## To see these entries in BibTeX format, use 'print(<citation>,
## bibtex=TRUE)', 'toBibtex(.)', or set
## 'options(citation.bibtex.max=999)'.

##
## To cite package 'tidyr' in publications use:
##
```

```
##   Hadley Wickham and Lionel Henry (2020). tidyr: Tidy Messy Data. R
##   package version 1.0.2. https://CRAN.R-project.org/package=tidyr
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {tidyr: Tidy Messy Data},
##     author = {Hadley Wickham and Lionel Henry},
##     year = {2020},
##     note = {R package version 1.0.2},
##     url = {https://CRAN.R-project.org/package=tidyr},
##   }

##
## To cite ggplot2 in publications, please use:
##
##   H. Wickham. ggplot2: Elegant Graphics for Data Analysis.
##   Springer-Verlag New York, 2016.
##
## A BibTeX entry for LaTeX users is
##
##   @Book{,
##     author = {Hadley Wickham},
##     title = {ggplot2: Elegant Graphics for Data Analysis},
##     publisher = {Springer-Verlag New York},
##     year = {2016},
##     isbn = {978-3-319-24277-4},
##     url = {https://ggplot2.tidyverse.org},
##   }

##
## To cite package 'baySeq' in publications use:
##
##   Thomas J. Hardcastle (2019). baySeq: Empirical Bayesian analysis of
##   patterns of differential expression in count data. R package version
##   2.20.0.
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {baySeq: Empirical Bayesian analysis of patterns of differential
## expression in count data},
##     author = {Thomas J. Hardcastle},
##     year = {2019},
##     note = {R package version 2.20.0},
##   }
##
## ATTENTION: This citation information has been auto-generated from the
## package DESCRIPTION file and may need manual editing, see
## 'help("citation")'.

##
## To cite package 'EnhancedVolcano' in publications use:
```

```
##
##   Kevin Blighe, Sharmila Rana and Myles Lewis (2019). EnhancedVolcano:
##   Publication-ready volcano plots with enhanced colouring and labeling.
##   R package version 1.4.0.
##   https://github.com/kevinblighe/EnhancedVolcano
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {EnhancedVolcano: Publication-ready volcano plots with enhanced colouring and
## labeling},
##     author = {Kevin Blighe and Sharmila Rana and Myles Lewis},
##     year = {2019},
##     note = {R package version 1.4.0},
##     url = {https://github.com/kevinblighe/EnhancedVolcano},
##   }
##
## ATTENTION: This citation information has been auto-generated from the
## package DESCRIPTION file and may need manual editing, see
## 'help("citation")'.
```