The babel vignette

Adam B. Olshen, Richard A. Olshen, and Barry S. Taylor

February 10, 2026

1 Introduction

This document presents an overview of the `babel` package. This package is for analyzing ribosome profiling data. Ribosome profiling data arises from paired next generation sequencing experiments [1, 2]. One experiment is standard RNA-seq, which we abbreviate as mRNA. The other is RNA-seq of ribosome-protected fragments, which we abbreviate as RPF. It implements our methodology for finding genes that have unusual RPF counts given their mRNA counts [3].

Our model is that the mRNA counts follow a negative binomial distribtion, as is standard [4, 5]. Because sample sizes are always low in this context, we estimate a single over-dispersion parameter across genes using the Robinson and Smyth methodology found in edgeR [6]. Given the mRNA counts, we model the RPF counts also as negative binomial with mean according to a trimmed least sequares regression fit of RPF counts on mRNA counts and over-dispersion estimated in an iterative fashion [3].

As we detail below, our package does three main things. First, we identify genes with unusual RPF counts given their mRNA counts *within* conditions. Second, we combine p-values across multiple experiments within conditions. Third, we use the within-condition p-values to identify genes where the RPF/mRNA relationship has changed *between* conditions.

2 Data

We selected a ribosome profiling data set consisting of 1000 genes from two conditions (GroupA and GroupB) with each condition having two replicates (Sample1 and Sample2). We are calling this data set `ribo.prof`.

3 An Example

Here we perform an analysis on the `ribo.prof` data described above. First we load the library and data.

```
> library(babel)
> data(ribo.prof)
```

Next we print the first five lines of the mRNA (`test.rna`) and RPF (`test.rp`) elements of `ribo.prof`. Note that these are raw counts that are not normalized in any way. Raw counts must be used with `babel`.

```
> test.rna <- ribo.prof$test.rna
> print(test.rna[1:5, ])
      Sample1_groupA Sample1_groupB Sample2_groupA Sample2_groupB
gene1            812            812           1008           1176
gene2           1138           1238           1542           1358
gene3            160            207            274            437
gene4            993           1343            800           1171
gene5            259            307            279            347
```

```
> test.rp <- ribo.prof$test.rp
> print(test.rp[1:5, ])
      Sample1_groupA Sample1_groupB Sample2_groupA Sample2_groupB
gene1            282            207            848            748
```

```
gene2           224        61       392       269
gene3           109        65       409       359
gene4           154       119       188       247
gene5            95        39       210       182
```

We see paired count data for the first five genes. We plot the Sample1 and GroupA data, with both mRNA and RPF counts on the log scale. We expect and actually do see an increasing function between the mRNA counts and RPF counts.

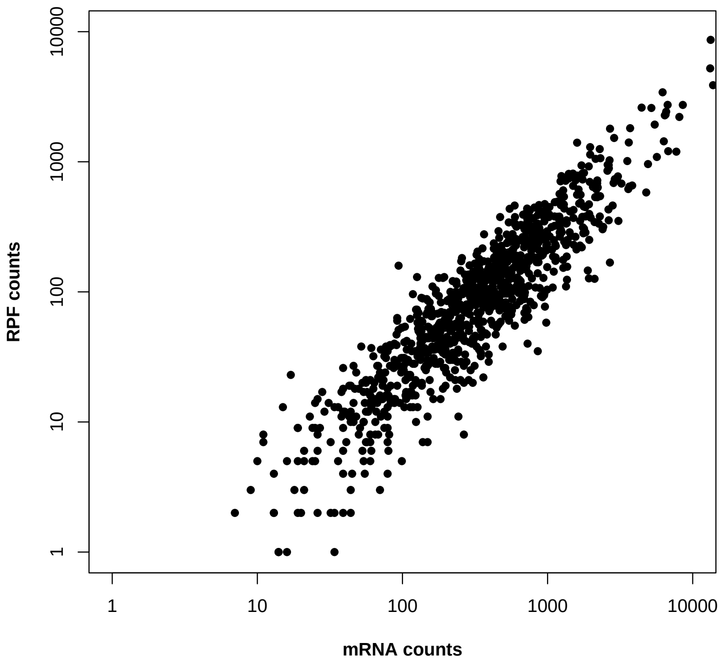

Figure 1: Scatterplot of Sample1 and GroupA data.

Similar patterns are repeated across the three other experiments. Next we estimate p-values for the RPF counts given the mRNA counts within every experiment. The tests are one-sided; higher than expected RPF counts lead to low p-values, while lower than expected RPF counts lead to high p-values.

We set group labels corresponding to the columns of the mRNA and RPF inputs.

```
> test.group <- c("A", "B", "A", "B")
```

Then, for the purpose of reproducibility, we fix the number of cores at 1. Two cores are used by default, unless the machine is running Windows, where one is the default because Windows cannot use the fork command in the parallel library. Users are encouraged to use multiple cores if they are available. Future versions of the software will allow computation over multiple nodes of a computational cluster.

```
> options(mc.cores = 1)
```

We set the seed for reproducibility purposes.

```
> set.seed(12345)
```

We run the main function, which is called `babel()`. The argument `nreps` is set to 100000. This is the number of permutations, so the minimum p-value is `1/nreps`. Therefore, when correcting for multiple comparisons, We would prefer a million, or, even better, ten million reps, but here we are just demonstrating the procedure on one core. The argument `min.rna` is set to 10. This is the minimum number of mRNA counts acrossing all experiments for a gene to be included. This cutoff leads to the removal of 9 genes so that 991 are analyzed.

```
> test.babel <- babel(test.rna, test.rp, group = test.group,
+     nreps = 1e+05, min.rna = 10)
[1] "Running Sample1_groupA"
[1] "Running Sample1_groupB"
[1] "Running Sample2_groupA"
[1] "Running Sample2_groupB"
```

Now we examine the first five lines of the within element of the list created by the `babel()` run.

```
> within.babel <- test.babel$within
> print(within.babel[[1]][1:5, ])
   Gene Direction P-value (one-sided) P-value (two-sided)       FDR
1 gene1         1          0.28498430          0.56996860 0.9869836
2 gene2        -1          0.68115638          0.63768725 0.9869836
3 gene3         1          0.02611948          0.05223896 0.9869836
4 gene4        -1          0.80469391          0.39061219 0.9869836
5 gene5         1          0.24797504          0.49595008 0.9869836
```

The element `Direction` is whether the RPF count is greater (1) or less (-1) than expected given the mRNA count. The `P-value (one-sided)` tells us how unusual the RPF count is given the mRNA count, with low or high being interesting. The `P-value (two-sided)` correspond to how unusual the RPF count is relative to the mRNA count, with only low being interesting. The `FDR` is the estimated false discovery rate corresponding to the gene for the two-sided p-value.

We plot the interesting genes from the within analysis on this sample. Genes in red have (one-sided) p-values < 0.025, while genes in green have p-values > 0.975.

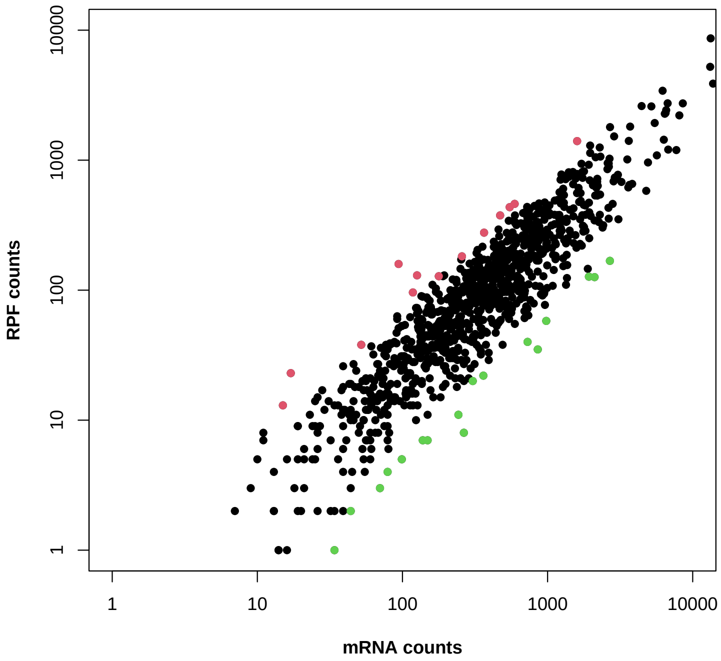

Figure 2: Scatterplot of Sample1 and GroupA data with unusual genes highlighted.

Next we combine the p-values from the two experiments from Group A using the technique described in our manuscript [3], and show the first five lines from the results. Note there is now a single (two-sided) combined p-value and a corresponding FDR. The Direction let us know whether RPF counts are higher or lower than expected.

```
> combined.babel <- test.babel$combined
> print(combined.babel[[1]][1:5, ])
   Gene Direction     P-value       FDR
1 gene1         1 0.299404920 0.5701312
2 gene2        -1 0.242812560 0.5190445
3 gene3         1 0.006557498 0.1291361
4 gene4        -1 0.121210537 0.3800734
5 gene5         1 0.320230001 0.5871779
```

We plot the genes at the same p-value cutoffs as before. Note the increase in power from combining experiments and that all highlighted genes are unusual in both samples.

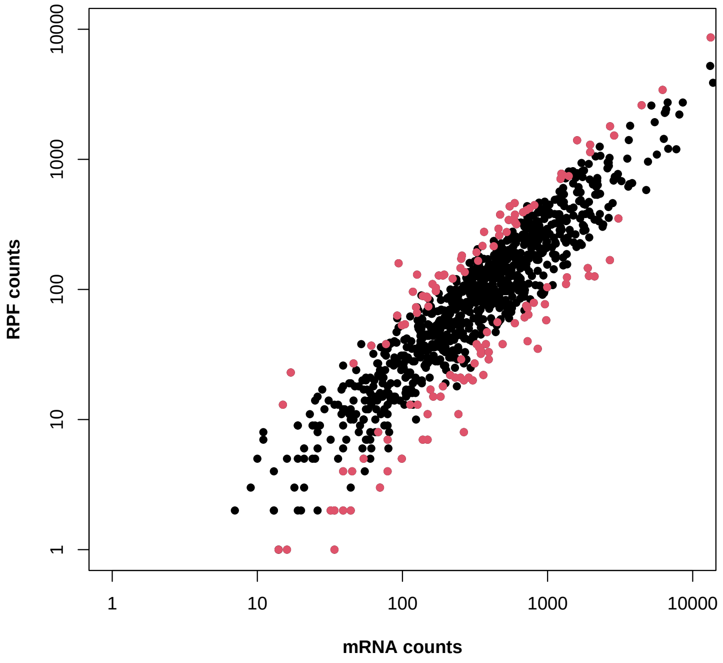

Figure 3: Scatterplot of the Sample1 and GroupA data with unusual genes highlighted from combined analysis.

Next we look for genes whose RPF to mRNA counts vary between group A and group B. Here there are only two groups, but babel automatically tests all pairwise combinations. We again print the first five lines of the output.

```
> between.babel <- test.babel$between
> print(between.babel[[1]][1:5, ])
   Gene  mRNA_logFC  mRNA_FDR       Change_type   P-value       FDR Direction
1 gene1  0.01894523 1.0000000 translational_only 0.8111135 0.9964579        -1
2 gene2  0.17853400 1.0000000 translational_only 0.2691062 0.9964579         1
3 gene3 -0.42625694 0.9973087 translational_only 0.1315799 0.9964579         1
4 gene4 -0.33674012 0.9973087 translational_only 0.7978599 0.9964579        -1
5 gene5 -0.13679618 1.0000000 translational_only 0.2334080 0.9964579         1
```

The elements mRNA_logFC and mRNA_FDR are based on tests for differential expression just on the mRNA data [6]. Genes with mRNA_FDR < 0.05 are labeled as ”both,” meaning change is in expression and translation; other genes are labeled ”translation_only”. There is a single (two-sided) p-value for the difference in RPF relative to mRNA and the corresponding FDR. The element Direction determines the type of change (1 for translation higher in the first group label, -1 for lower in the first group label). Note that an FDR of 25% there are 11 significant genes.

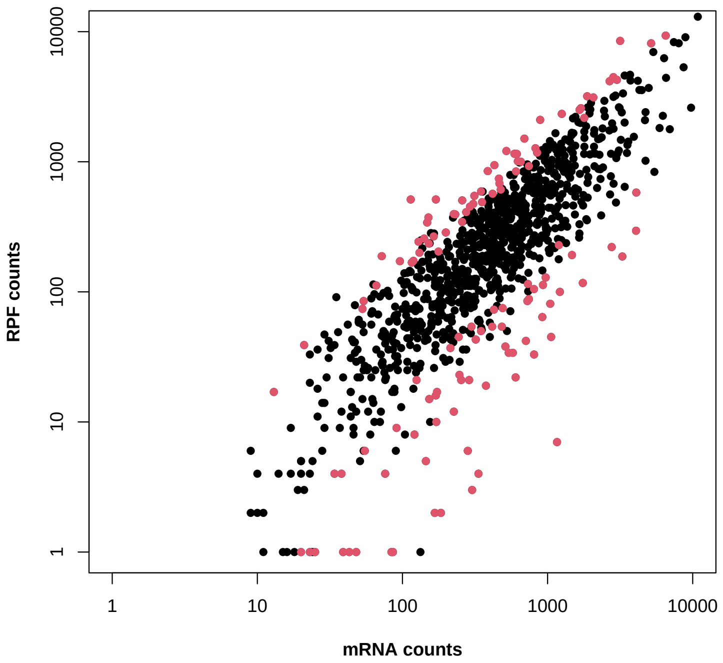

Figure 4: Scatterplot of the Sample2 and GroupA data with unusual genes highlighted from combined analysis.

References

[1] N. T. Ingolia, S. Ghaemmaghami, J. Newman, and J. S. Weissman, "Assembly of microarrays for genome-wide measurement of dna copy number," *Science*, vol. 324, pp. 218–23, Genome-wide analysis in vivo of translation with nucleotide resolution using ribosome profiling.

[2] A. C. Hsieh, Y. Liu, M. P. Edlind, N. T. Ingolia, M. R. Janes, A. Sher, E. Y. Shi, C. R. Stumpf, C. Christensen, M. J. Bonham, S. Wang, P. Ren, M. Martin, K. Jessen, J. S. W. M. E. Feldman, K. M. Shokat, C. Rommel, and D. Ruggero, "The translational landscape of mtor signalling steers cancer initiation and metastasis.," *Nature*, vol. 485, pp. 55–61, 2012.

[3] A. B. Olshen, A. C. Hsieh, C. R. Stumpf, R. A. Olshen, D. Ruggero, and B. S. Taylor, "Assessing gene-level translational control from ribosome profiling.," *Bioinformatics*, 2013.

[4] M. D. Robinson and G. K. Smyth, "Small-sample estimation of negative binomial dispersion, with applications to sage data.," *Biostatistics*, vol. 9, pp. 321–32, 2008.

[5] S. Anders and W. Huber, "Differential expression analysis for sequence count data.," *Genome Biology*, vol. 11, p. R106, 2010.

[6] M. D. Robinson, D. J. McCarthy, and G. K. Smyth, "edger: a bioconductor package for differential expression analysis of digital gene expression data.," *Bioinformatics*, vol. 26, pp. 139–40, 2010.

Appendix

Session information

- R version 4.5.2 (2025-10-31), x86_64-pc-linux-gnu
- Locale: LC_CTYPE=en_US.UTF-8, LC_NUMERIC=C, LC_TIME=en_US.UTF-8, LC_COLLATE=C, LC_MONETARY=en_US.UTF-8, LC_MESSAGES=en_US.UTF-8, LC_PAPER=en_US.UTF-8, LC_NAME=C, LC_ADDRESS=C, LC_TELEPHONE=C, LC_MEASUREMENT=en_US.UTF-8, LC_IDENTIFICATION=C
- Time zone: Etc/UTC
- TZcode source: system (glibc)
- Running under: Ubuntu 24.04.3 LTS
- Matrix products: default
- BLAS: /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
- LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblasp-r0.3.26.so; LAPACK version3.12.0
- Base packages: base, datasets, grDevices, graphics, methods, stats, utils
- Other packages: R.devices 2.17.3, R.methodsS3 1.8.2, R.oo 1.27.1, R.utils 2.13.0, babel 0.3-0, edgeR 4.9.2, limma 3.67.0
- Loaded via a namespace (and not attached): R.cache 0.17.0, R.rsp 0.46.0, base64enc 0.1-6, buildtools 1.0.0, cli 3.6.5, compiler 4.5.2, digest 0.6.39, evaluate 1.0.5, grid 4.5.2, knitr 1.51, lattice 0.22-7, locfit 1.5-9.12, maketools 1.3.2, parallel 4.5.2, rlang 1.1.7, statmod 1.5.1, sys 3.4.3, tools 4.5.2, xfun 0.56

This report was generated using the R.rsp package.