A comparison of normalization methods for `fbseq`

June 20, 2016

Will Landau

Department of Statistics
Iowa State University

1 Introduction

In the case study [2], there is a question of how best to compute the normalization factors h_n. Currently, we compute them using the method in Section 3.2 (“default”). However, there's also the TMM method (“tmm”) [3], which is widely-used, and a third option of setting $h_n = 0$ for all n (“zero”). In order to compare these methods in terms of effectiveness, we use several performance metrics described in the case study: mean squared error, receiver operating characteristic (ROC) curves, areas under the ROC curves, calibration curves, and calibration errors. For data, we simulated one dataset with $N = 16$ columns and $G = 30000$ genes for each of the edgeR, model, and simple scenarios described in Section 4.3 of the case study. We analyzed the data with our fully Bayesian approach from Section 3 using the Monte Carlo runtime parameters in Section 3.2. For hardware, we used the Slurm system described at `http://it.las.iastate.edu/slurm-simple-linux-utility-resource-management`. The README file at `https://github.com/wlandau/normalization/blob/master/README.md` has instructions for reproducing the results below.

2 Results

We assessed convergence with Gelman-Rubin potential scale reduction factors on each parameter except for the ε_{gn}'s [1]. Five of the nine analyses had all Gelman factors below the common tolerance threshold of 1.1. The others had few high Gelman factors, all corresponding to gene-specific parameters. These specific Gelman factors are displayed in Table 1. Any evidence of lack of convergence is weak and unconvincing.

Table 1: For Section 2, Gelman-Rubin potential scale reduction factors above 1.1 for the non-ε_{gn} parameters.

Dataset	Analysis	Gelman factors	Respective parameters
edgeR	default	5.94, 2.33, 1.53, 1.20	$\beta_{13806,2}$, γ_{13806}, $\beta_{13806,1}$, $\beta_{13806,3}$
model	default	1.50, 1.13	$\beta_{838,1}$, $\beta_{15685,1}$
model	tmm	1.26	$\beta_{838,1}$
model	zero	1.35	$\beta_{838,1}$

Figure 1 shows mean squared errors (MSE) of the $\beta_{g\ell}$'s for each ℓ, dataset, and normalization method. MSE appears slightly lower for the TMM method, the edgeR scenario, and β_{g5}. Otherwise, MSE appear to depend little on the normalization method used.

Figure 2 shows receiver operating characteristic (ROC) curves for each dataset, analysis, and mode of heterosis. From this figure, differences in performance among the normalization methods appear extremely small and possibly insignificant. The biggest differences are seen for the “high H21” and “high mean” modes of heterosis, which favor the default method the best and the “zero” method the worst.

Figure 3 shows the areas under the ROC curves (AUC) in Figure 2. Subtle differences in performance are magnified here. Apparently, the default normalization method outperforms TMM and “zero” for the “high H21” and “high mean” modes of heterosis in the edgeR dataset. Everywhere else in the edgeR dataset, the TMM method does at least as well as the default and “zero” methods, outperforming them except for the “low H12” mode of heterosis. Differences in AUC for the model and simple datasets appear extremely small to insignificant.

Figure 4 shows calibration curves. Differences in calibration are extremely small for the model and simple scenarios. For the edgeR scenario, differences are more pronounced. The default method appears to perform uniformly worse than the TMM method for low-parent heterosis, but better than TMM for high-parent heterosis when probabilities exceed 0.5.

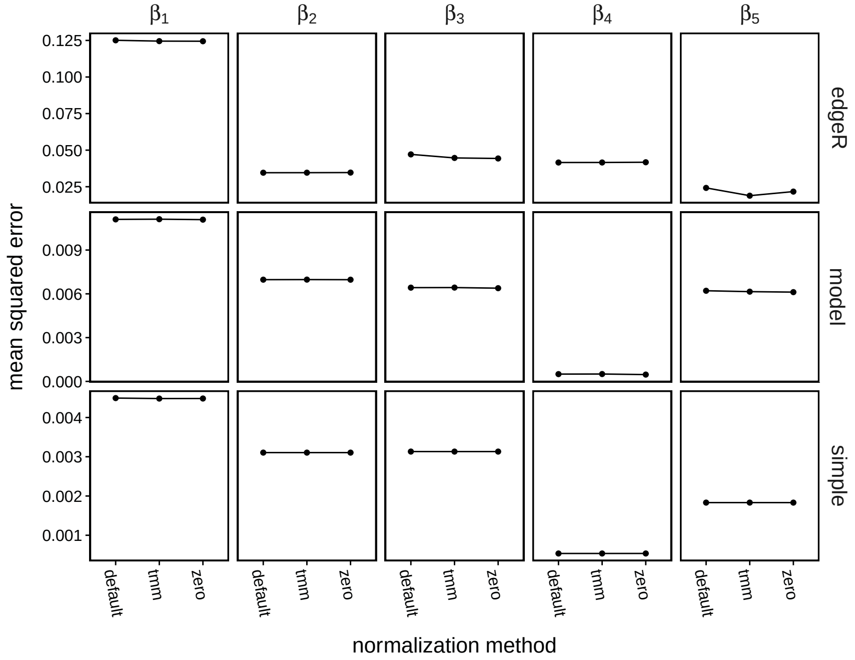

Figure 1: Mean squared errors of the $\beta_{g\ell}$ parameters, shown by simulation (right) and analysis/normalization (bottom) methods.

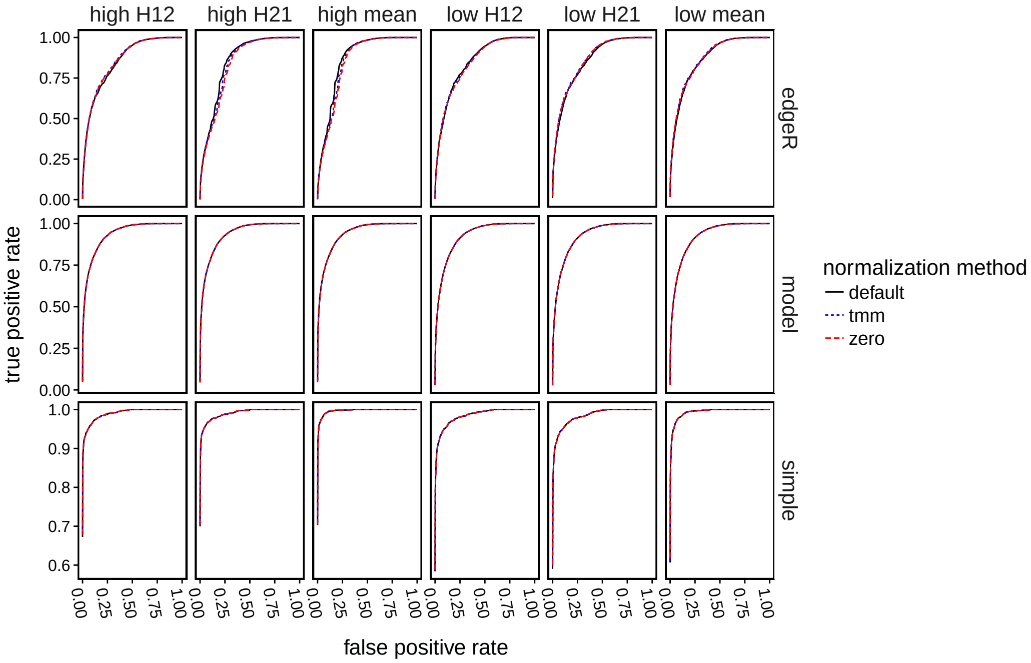

Figure 2: Receiver operating characteristic (ROC) of heterosis gene detection. The type of heterosis, as explained in the case study [2], is indicated at the top, and the dataset is indicated on the right.

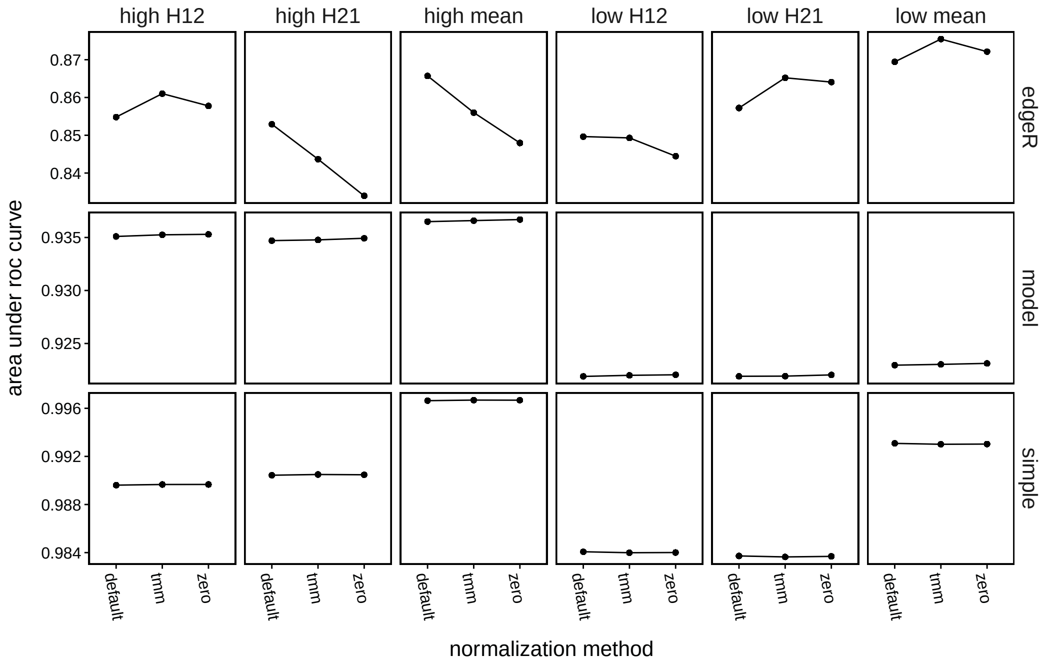

Figure 3: Areas under the ROC curves in Figure 2.

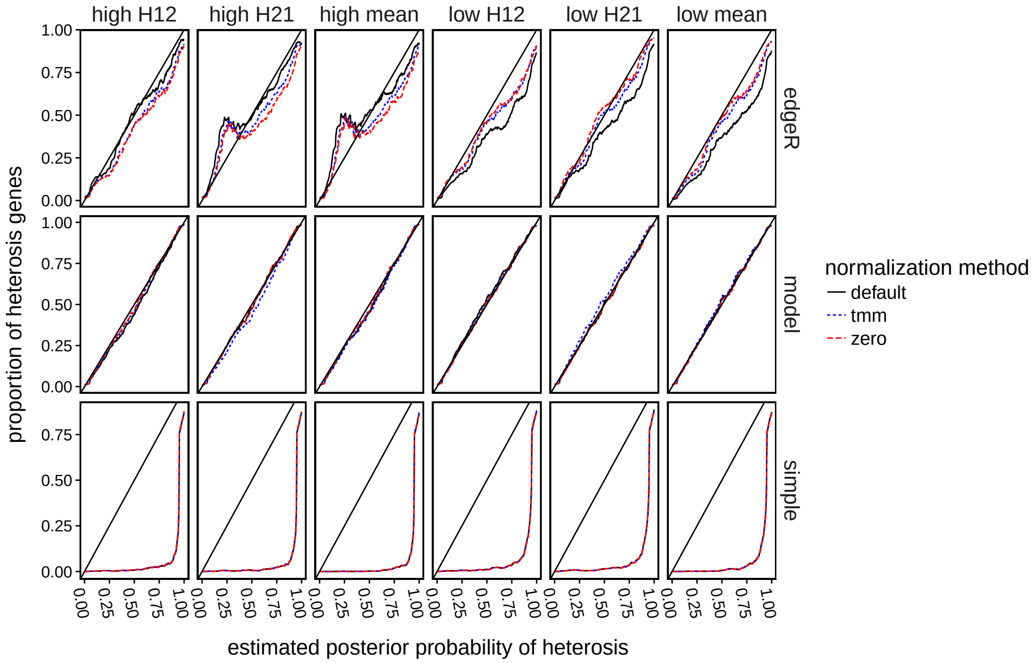

Figure 4: Calibration curves for each analysis of each dataset. The type of heterosis, as explained in the case study [2], is indicated at the top, and the dataset is indicated on the right.

Figure 5 shows the calibration errors taken from the calibration curves in Figure 4, magnifying the overall differences in the calibration of posterior probabilities of heterosis. Differences among normalization methods within the simple dataset appear negligibly small. For the “model” dataset, the TMM method appears to perform mostly worst, but from a glance at the uniformly excellent calibration curves in Figure 4 tells us that this performance loss is extremely small to insignificant. For the edgeR scenario, where differences in calibration are more significant, the rankings are clear. For high-parent heterosis, the “zero” method is the best and the default method is the second-best. For low-parent heterosis, this ranking completely reverses.

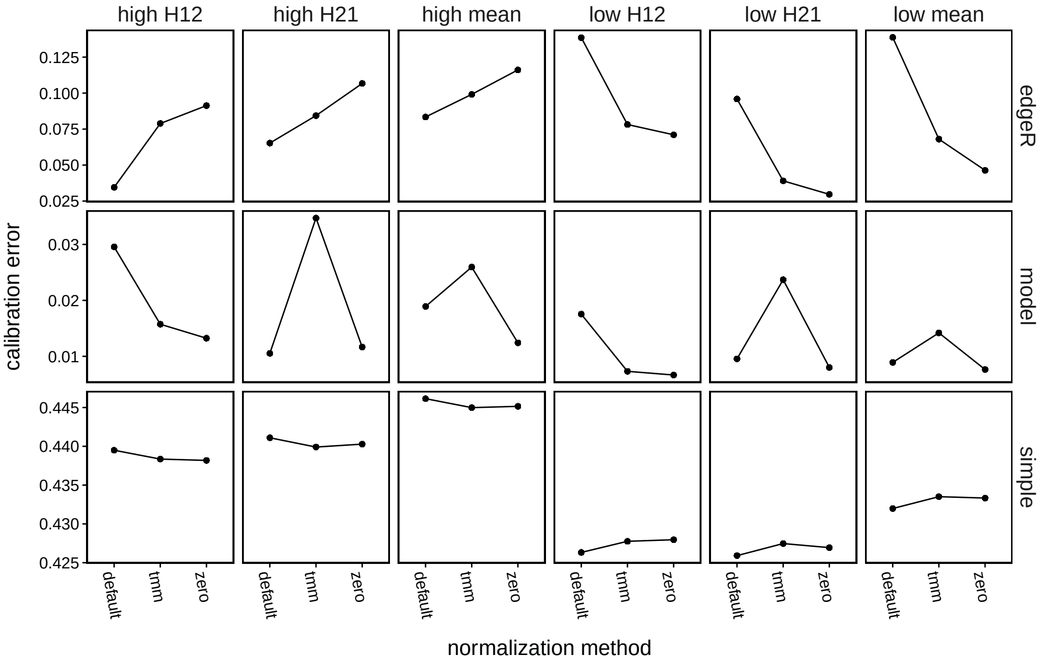

Figure 5: Calibration errors taken from the calibration curves in Figure 4.

3 Conclusion

For our three simulated datasets, with respect to MSE and ROC curves, the choice of normalization method for the h_n's appears inconsequential. Normalization does start to matter once we consider calibration, but there, the performance ranking for high-parent heterosis is the reverse of that of low-parent heterosis. So far, unless we exclude high-parent heterosis, there is no reason to prefer TMM over our default normalization method.

References

[1] Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis*. CRC Press, 3rd edition, 2013.

[2] William Landau, Jarad Niemi, and Dan Nettleton. Fully bayesian analysis of RNA-seq data for the detection of heterosis with respect to gene expression. Submission pending to the *Journal of the American Statistical Association*, 2016.

[3] M.D. Robinson and A. Oshlack. A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biology*, 11(3):R25, 2010.