Supplementary Figures

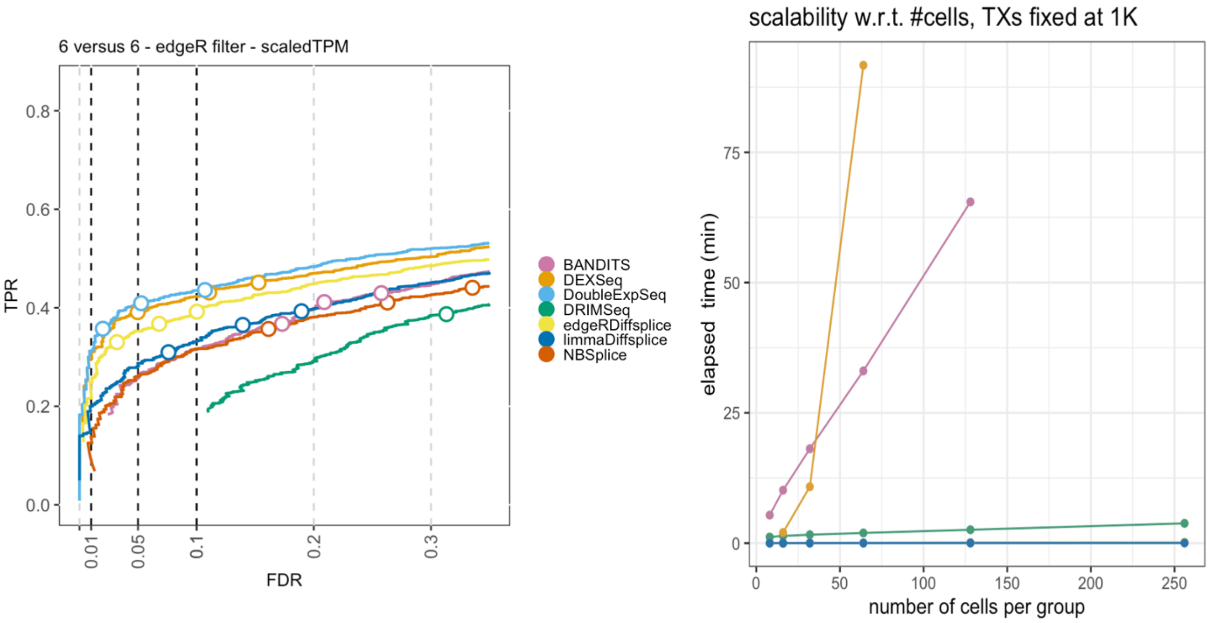


Figure S1: Performance and scalability evaluation on a subset of the Love *et al.* dataset. To allow for a performance and scalability evaluation of BANDITS, which does not scale to datasets with many transcripts, we here perform a DTU analysis for the *6 versus 6* samples dataset of Love *et al.* with only 1000 transcripts. **Left panel: performance evaluation.** The results are in line with those of Figure 1A. The performance of BANDITS is indicated in pink. **Right panel: Scalability evaluation.** BANDITS scales linearly with respect to the number of cells (or samples) in the dataset. The slope of the linear trend, however, is considerably larger than those of the other DTU methods that scale linearly. Note that the profiles of limma diffsplice, edgeR diffsplice and DoubleExpSeq overlap in this figure.

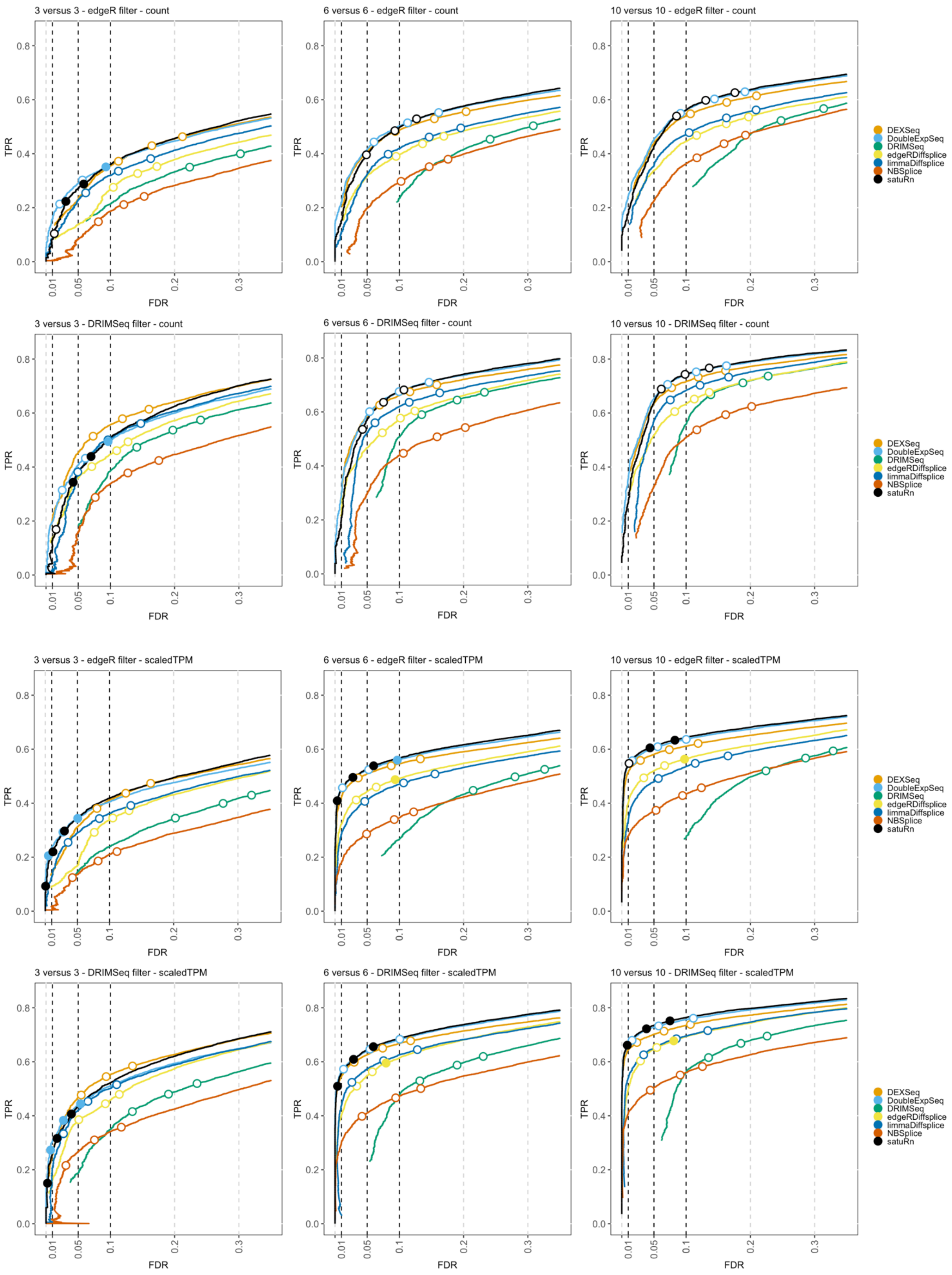


Figure S2: Performance evaluation of satuRn on different subsamples of the simulated bulk RNA-seq dataset by Love *et al.* FDR-TPR curves visualize the performance of each method by displaying the sensitivity of the method (TPR) with respect to the false discovery rate (FDR). The three circles on each curve represent working points when the FDR level is set at nominal levels of 1%, 5% and 10%, respectively. The circles are filled if the

empirical FDR is equal or below the imposed FDR threshold. We subsampled two-group comparisons according to three different samples sizes; a *3 versus 3*, *6 versus 6* and *10 versus 10* comparison, as denoted in the panel titles. The benchmark was performed both on the raw counts **(rows 1 and 2)** or on scaled transcripts-per-million (TPM) **(rows 3 and 4)** as imported with the Bioconductor R package tximport[1]. We additionally adopted two different filtering strategies: an edgeR-based filtering **(rows 1 and 3)** and a DRIMSeq-based filtering **(rows 2 and 4)**. Overall, the performance of satuRn is on par with those of the best tools in the literature, DEXSeq and DoubleExpSeq. In addition, satuRn achieves a better control of the FDR on all datasets. For extremely small sample size, i.e. the *3 versus 3* comparison, the performance is slightly below that of DEXSeq, and inference does become slightly too conservative. Note that, as expected, the performances increase with increasing sample size, and a higher performance is achieved with the more stringent DRIMSeq filtering criterion (see Methods), which goes at the cost of retaining fewer transcripts for DTU analysis. Finally, we note that the performances and FDR control are consistently better for the scaled TPM data as compared to the raw counts. Note that this was only observed for this particular dataset.

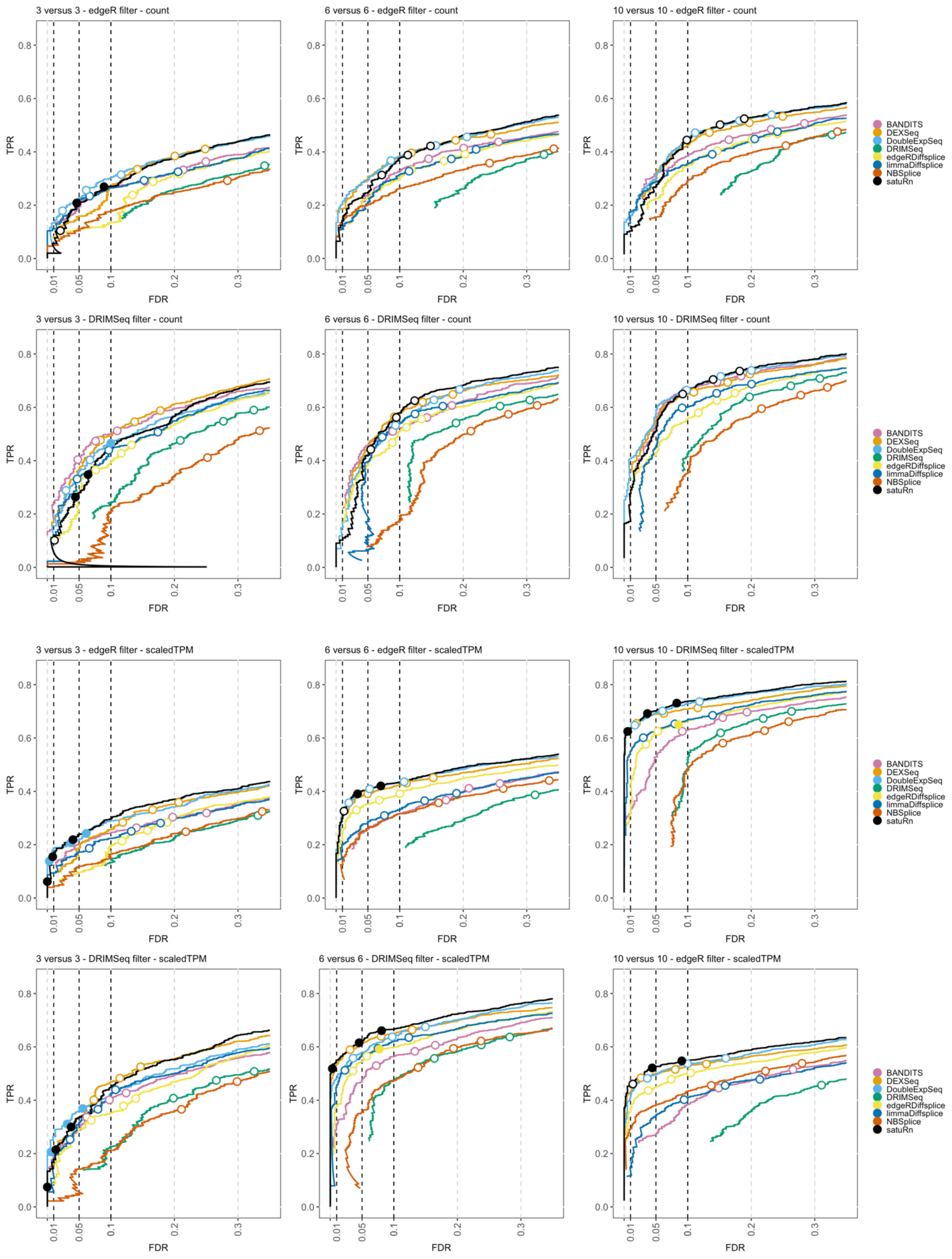


Figure S3: Performance evaluation on different subsamples of the simulated bulk RNA-seq dataset by Love *et al.* with a reduced number of transcripts to allow for a comparison with BANDITS. FDR-TPR curves visualize the performance of each method by displaying the sensitivity of the method (TPR) with respect to the

false discovery rate (FDR). The three circles on each curve represent working points when the FDR level is set at nominal levels of 1%, 5% and 10%, respectively. The circles are filled if the empirical FDR is equal or below the imposed FDR threshold. We subsampled two-group comparisons according to three different samples sizes; a *3 versus 3*, *6 versus 6* and *10 versus 10* comparison, as denoted on top of the panels. The benchmark was performed both on the raw counts **(rows 1 and 2)** or on scaled transcripts-per-million (TPM) **(rows 3 and 4)** as imported with the Bioconductor R package tximport[1]. We additionally adopted two different filtering strategies: an edgeR-based filtering **(rows 1 and 3)** and a DRIMSeq-based filtering **(rows 2 and 4)**. Note that, in contrast to Figure S2, we additionally randomly subsampled 1000 genes (~3000-5000 transcripts) after filtering, in order to reduce the number of transcripts in the data and thereby allowing for a DTU analysis with BANDITS. In concordance with Figure S2, the performance of satuRn is on par with the best tools of the literature with a better control of the FDR in general. While the performance of BANDITS is good for the settings for which it was originally developed, (i.e., small datasets with a stringent filtering criterium), its performance is reduced in larger, more leniently filtered datasets and inference is also overly liberal in these settings. In addition, while all other methods perform much better on the scaledTPM data (rows 3 and 4) than on the raw count data (rows 1 and 2), BANDITS has a similar performance on both input data types. This can be explained by the fact that BANDITS inherently corrects for differences in transcript length, even when raw counts are used as an input.

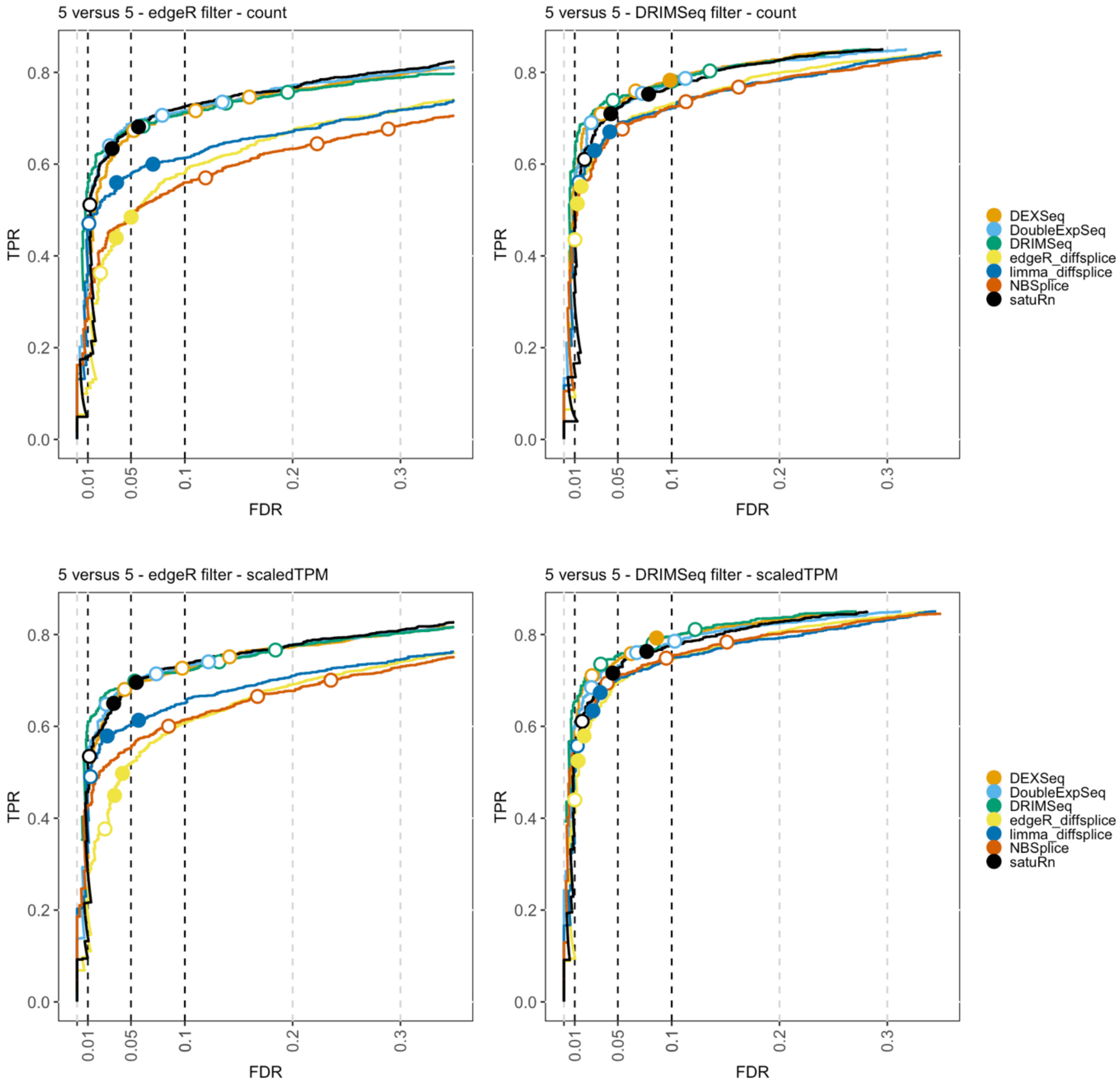


Figure S4: Performance evaluation of DTU methods on the "Dmelanogaster" simulated bulk RNA-seq dataset by Van den Berge *et al.* FDR-TPR curves visualize the performance of each method by displaying the sensitivity of the method (TPR) with respect to the false discovery rate (FDR). The three circles on each curve represent working points when the FDR level is set at nominal levels of 1%, 5% and 10%, respectively. The circles are filled if the empirical FDR is equal or below the imposed FDR threshold. The benchmark was performed both on the raw counts **(row 1)** and on scaled TPM **(row 2)** as obtained with the Bioconductor R package tximport^{1}. We additionally adopted two different filtering strategies; an edgeR-based filtering **(column 1)** and a DRIMSeq-based filtering **(column 2)**. Overall, the performance of satuRn is on par with those of the best tools in the literature, DEXSeq and DoubleExpSeq. In contrast to the performance evaluation on the dataset by Love *et al.* (Figures 1A and S2), there is a limited difference in performances based on the data input type (i.e., counts versus scaled TPM), and DRIMSeq also performs well on these datasets.

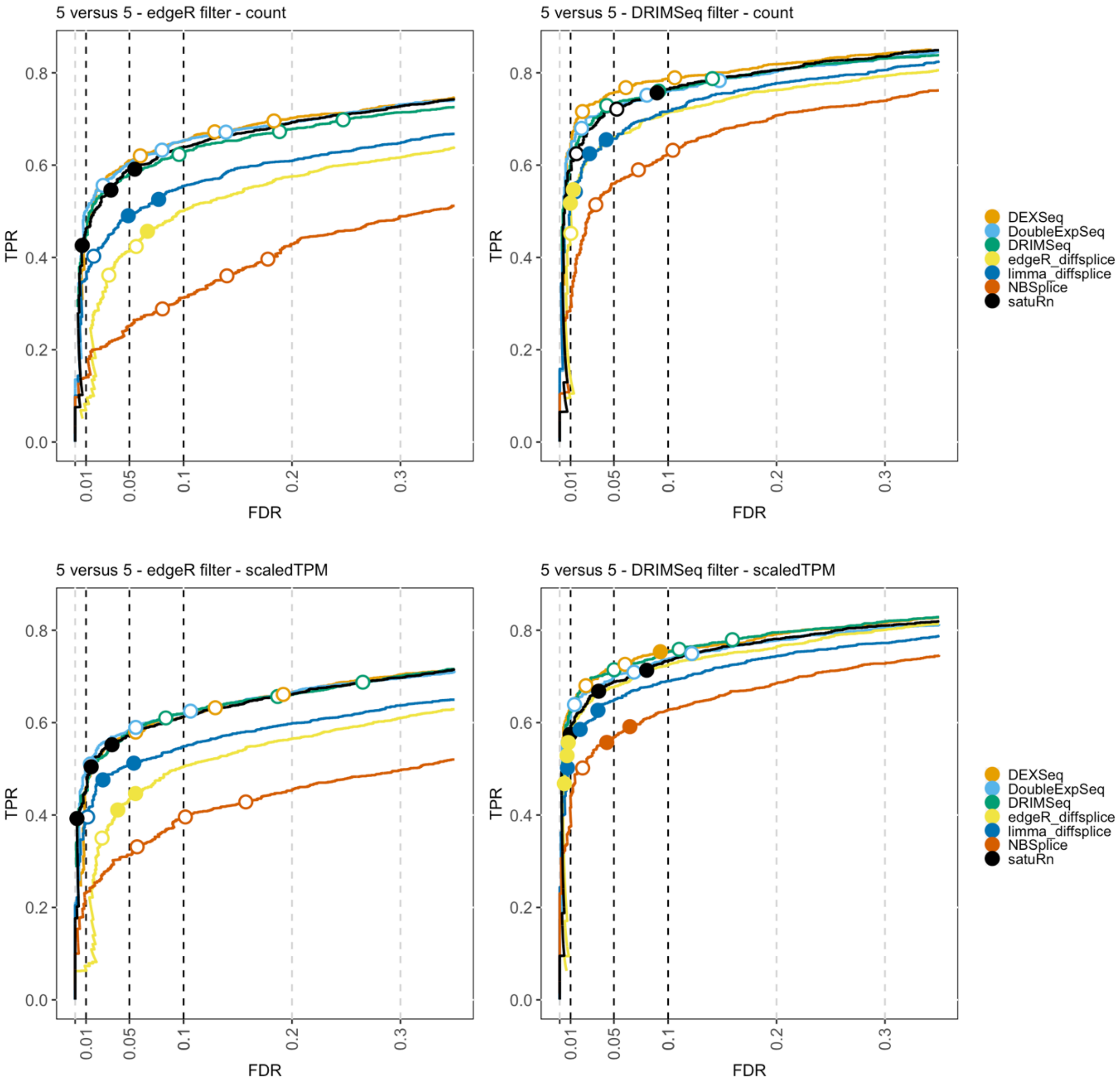
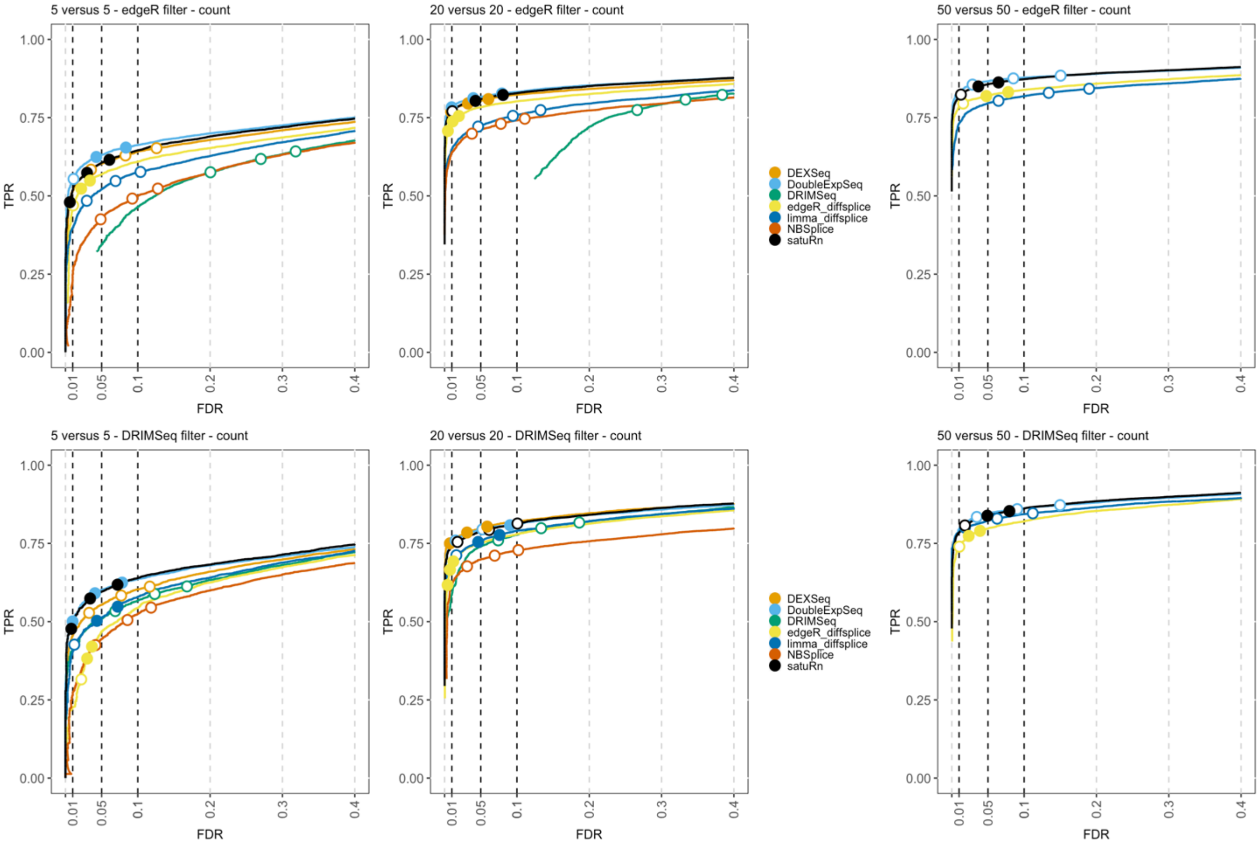


Figure S5: Performance evaluation of DTU methods on the "Hsapiens" simulated bulk RNA-seq dataset by Van den Berge *et al.* FDR-TPR curves visualize the performance of each method by displaying the sensitivity of the method (TPR) with respect to the false discovery rate (FDR). The three circles on each curve represent working points when the FDR level is set at nominal levels of 1%, 5% and 10%, respectively. The circles are filled if the empirical FDR is equal or below the imposed FDR threshold. The benchmark was performed both on the raw counts **(row 1)** and on scaled TPM **(row 2)** as obtained with the Bioconductor R package tximport[1]. We additionally adopted two different filtering strategies; an edgeR-based filtering **(column 1)** and a DRIMSeq-based filtering **(column 2)**. Overall, the performance of satuRn is on par with those of the best tools in the literature, DEXSeq and DoubleExpSeq. In contrast to the performance evaluation on the dataset by Love *et al.* (Figures 1A and S2),), there is a limited difference in performances based on the data input type (i.e., counts versus scaled TPM), and DRIMSeq also performs well on these datasets.


5 versus 5 - edgeR filter - count
20 versus 20 - edgeR filter - count
50 versus 50 - edgeR filter - count
5 versus 5 - DRIMSeq filter - count
20 versus 20 - DRIMSeq filter - count
50 versus 50 - DRIMSeq filter - count
TPR
FDR
0.00
0.25
0.50
0.75
1.00
0.01
0.05
0.1
0.2
0.3
0.4
DEXSeq
DoubleExpSeq
DRIMSeq
edgeR_diffsplice
limma_diffsplice
NBSplice
satuRn

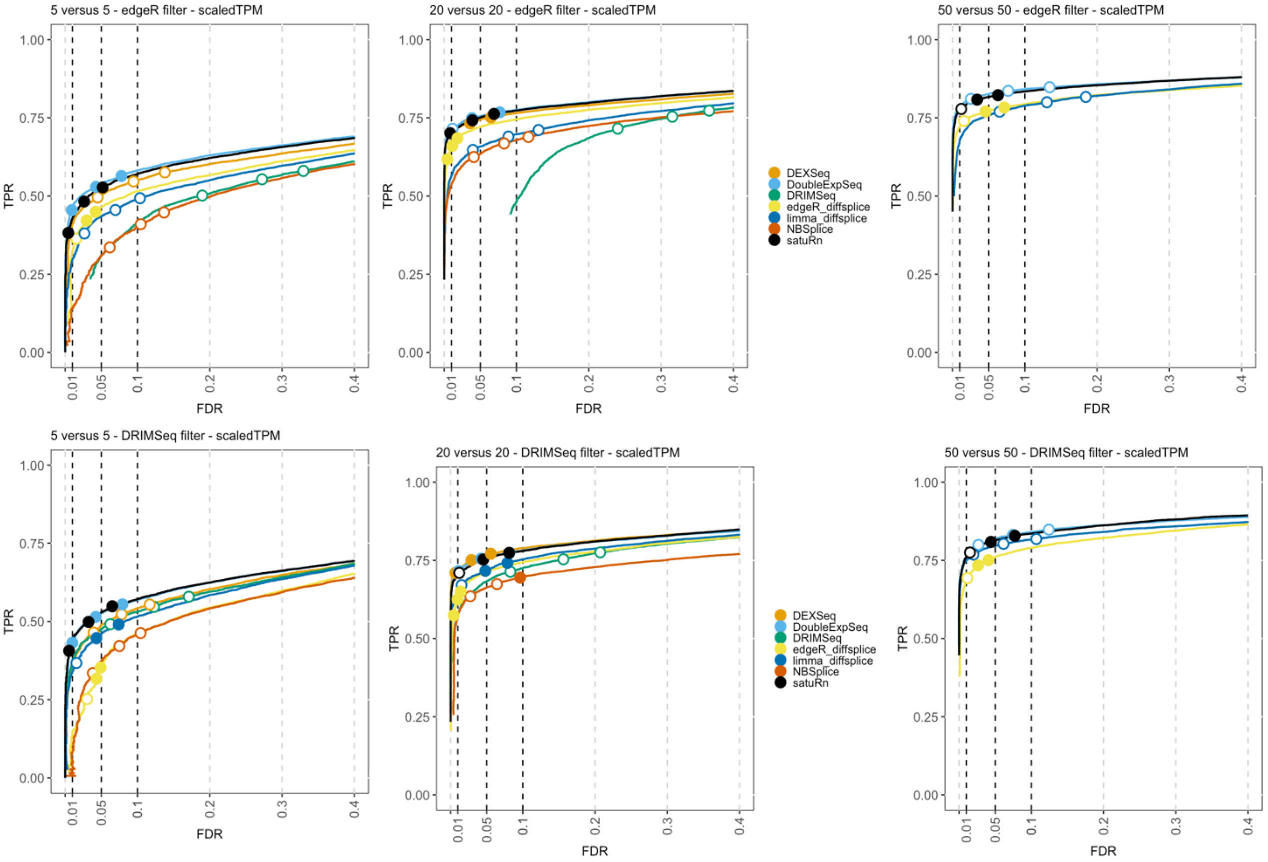

5 versus 5 - edgeR filter - scaledTPM
20 versus 20 - edgeR filter - scaledTPM
50 versus 50 - edgeR filter - scaledTPM
5 versus 5 - DRIMSeq filter - scaledTPM
20 versus 20 - DRIMSeq filter - scaledTPM
50 versus 50 - DRIMSeq filter - scaledTPM
TPR
FDR
0.00
0.25
0.50
0.75
1.00
0.01
0.05
0.1
0.2
0.3
0.4
DEXSeq
DoubleExpSeq
DRIMSeq
edgeR_diffsplice
limma_diffsplice
NBSplice
satuRn

Figure S6: Performance evaluation of DTU methods on the GTEx bulk RNA-seq dataset. FDR-TPR curves visualize the performance of each method by displaying the sensitivity (TPR) with respect to the false discovery rate (FDR). The three circles on each curve represent working points when the FDR level is set at nominal levels of 1%, 5% and 10%, respectively. The circles are filled if the empirical FDR is equal or below the imposed FDR threshold. The benchmark was performed both on the raw counts **(rows 1 and 2)** or on scaled transcripts-per-million (TPM) **(rows 3 and 4)** as obtained with the Bioconductor R package tximport[1]. We additionally adopted two different filtering strategies; an edgeR-based filtering **(rows 1 and 3)** and a DRIMSeq-based filtering **(rows 2 and 4)**. The performance of satuRn is on par with the best tools from the literature, DEXSeq and DoubleExpSeq. In addition, satuRn consistently provides a stringent control of the FDR, while DoubleExpSeq becomes more liberal with increasing sample sizes. Note that DEXSeq, DRIMSeq and NBSplice were omitted from the largest comparison, as these methods do not scale to large datasets (Figure1).

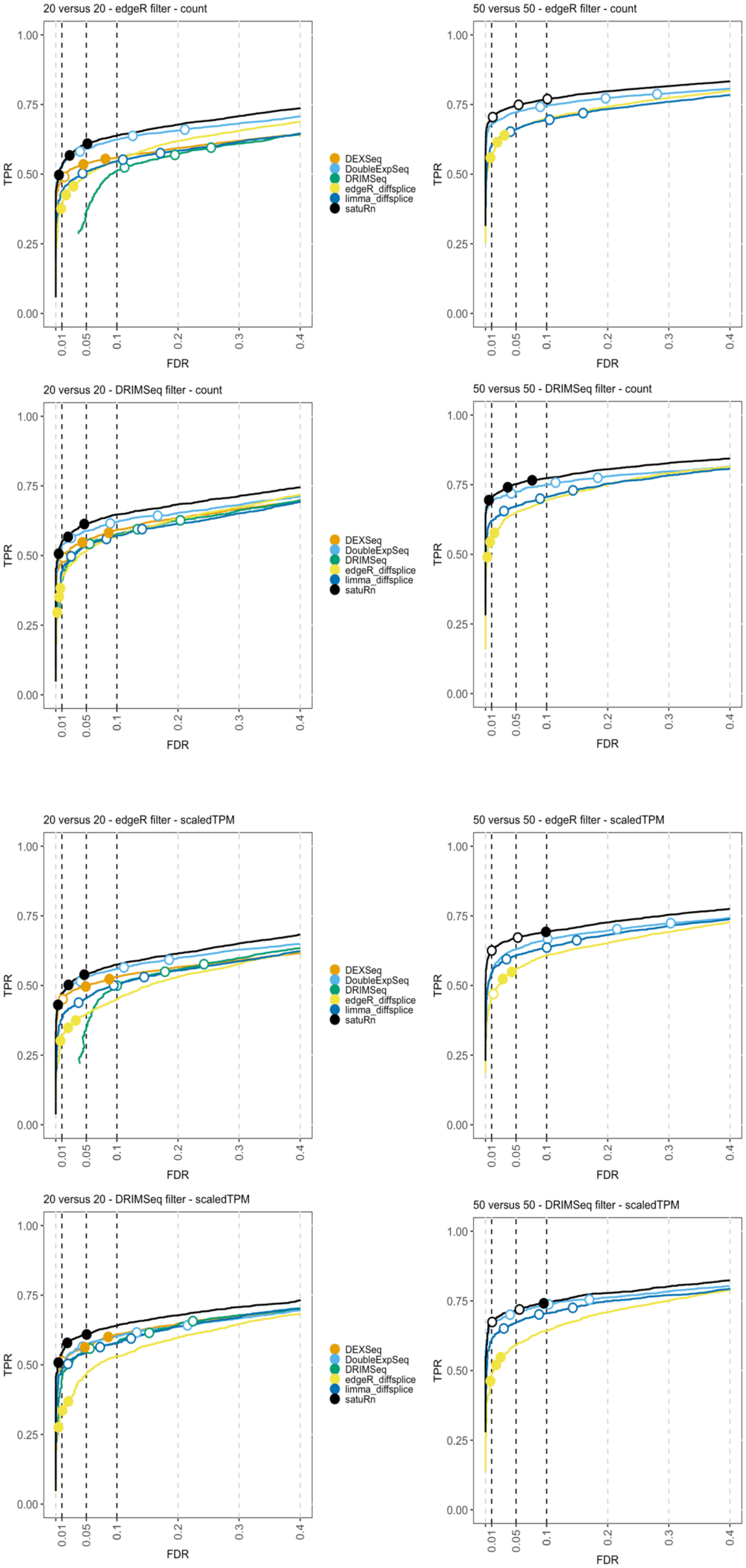
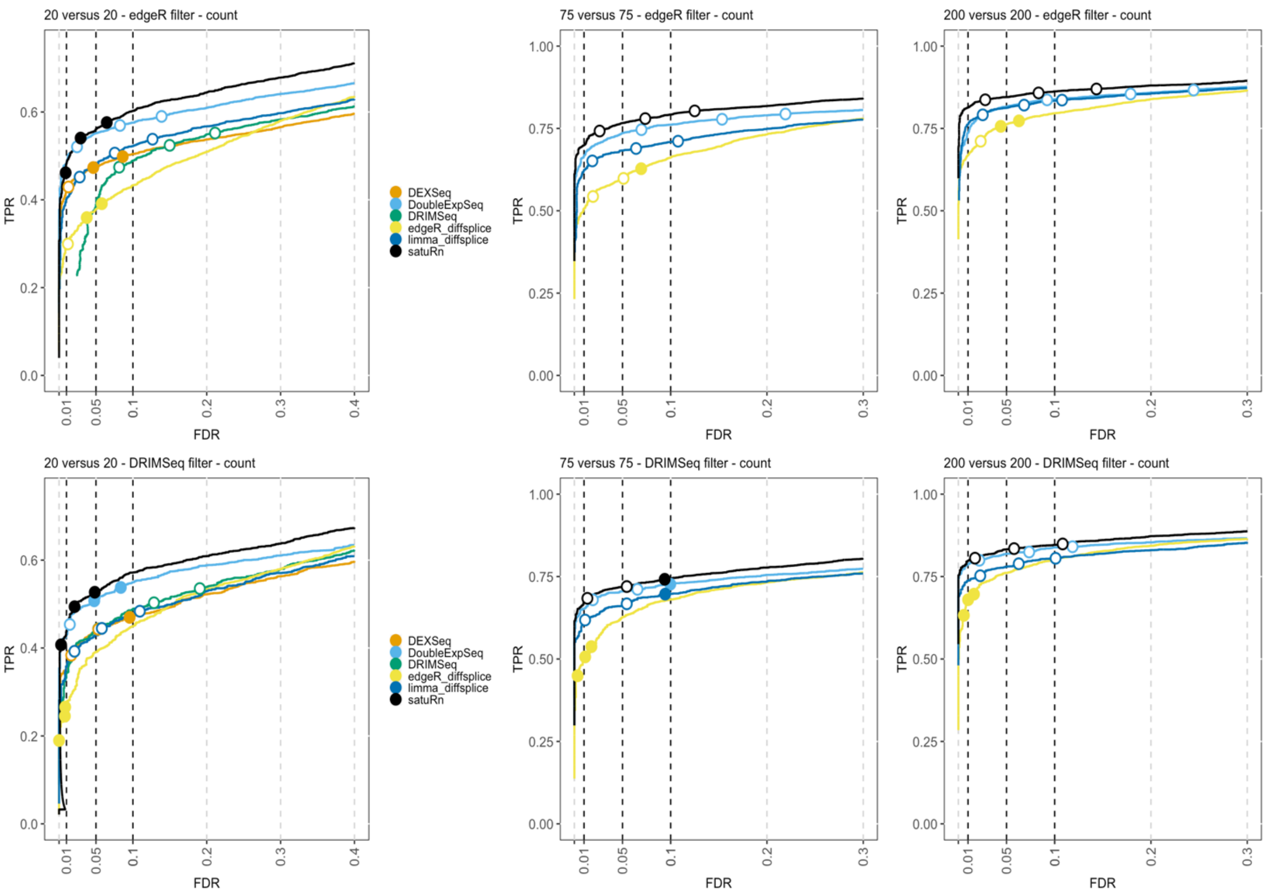

20 versus 20 - edgeR filter - count
50 versus 50 - edgeR filter - count
20 versus 20 - DRIMSeq filter - count
50 versus 50 - DRIMSeq filter - count
20 versus 20 - edgeR filter - scaledTPM
50 versus 50 - edgeR filter - scaledTPM
20 versus 20 - DRIMSeq filter - scaledTPM
50 versus 50 - DRIMSeq filter - scaledTPM
TPR
FDR
0.00
0.25
0.50
0.75
1.00
0.01
0.05
0.1
0.2
0.3
0.4
DEXSeq
DoubleExpSeq
DRIMSeq
edgeR_diffsplice
limma_diffsplice
satuRn

Figure S7: Performance evaluation of DTU methods on the real scRNA-seq dataset by Chen *et al.* FDR-TPR curves visualize the performance of each method by displaying the sensitivity of the method (TPR) with respect to the false discovery rate (FDR). The three circles on each curve represent working points when the FDR level is set at nominal levels of 1%, 5% and 10%, respectively. The circles are filled if the empirical FDR is equal or below the imposed FDR threshold. The benchmark was performed both on the raw counts **(rows 1 and 2)** or on scaled transcripts-per-million (TPM) **(rows 3 and 4)** as obtained with the Bioconductor R package tximport[1]. We additionally adopted two different filtering strategies; an edgeR-based filtering **(rows 1 and 3)** and a DRIMSeq-based filtering **(rows 2 and 4)**. The performance of satuRn is at least on par with the best tools from the literature. Note that the performance of DEXSeq is clearly lower. In addition, our method consistently controls the FDR close to its imposed nominal FDR threshold, while DoubleExpSeq becomes more liberal with increasing sample sizes. DEXSeq and DRIMSeq were omitted from the largest comparison (two groups with 50 cells each), as these methods do not scale to large datasets (Figure 1). NBSplice was omitted from all comparisons, as it does not converge on datasets with many zeros, such as scRNA-seq datasets.


20 versus 20 - edgeR filter - count
75 versus 75 - edgeR filter - count
200 versus 200 - edgeR filter - count
20 versus 20 - DRIMSeq filter - count
75 versus 75 - DRIMSeq filter - count
200 versus 200 - DRIMSeq filter - count
TPR
FDR
0.0
0.2
0.4
0.6
0.00
0.25
0.50
0.75
1.00
0.01
0.05
0.1
0.2
0.3
0.4
DEXSeq
DoubleExpSeq
DRIMSeq
edgeR_diffsplice
limma_diffsplice
satuRn

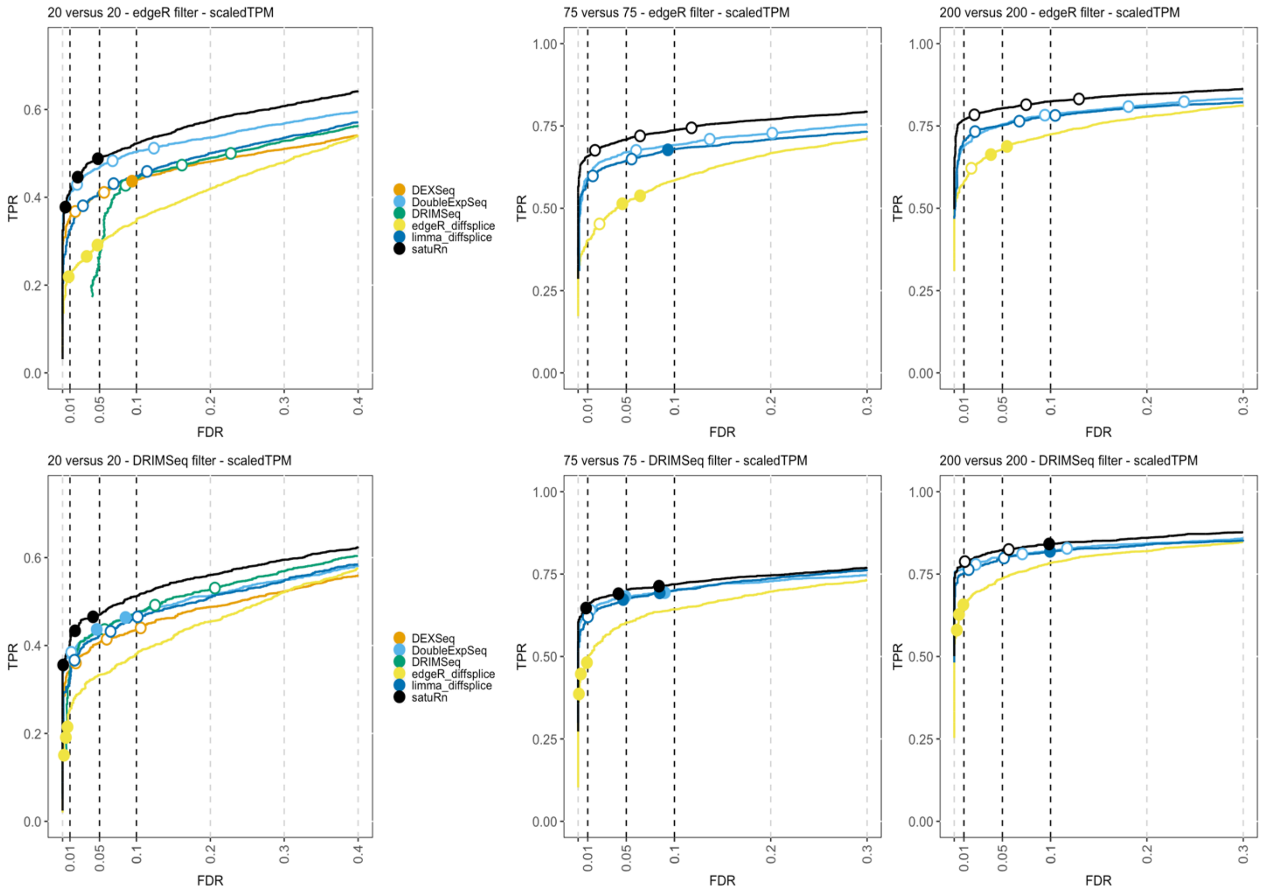
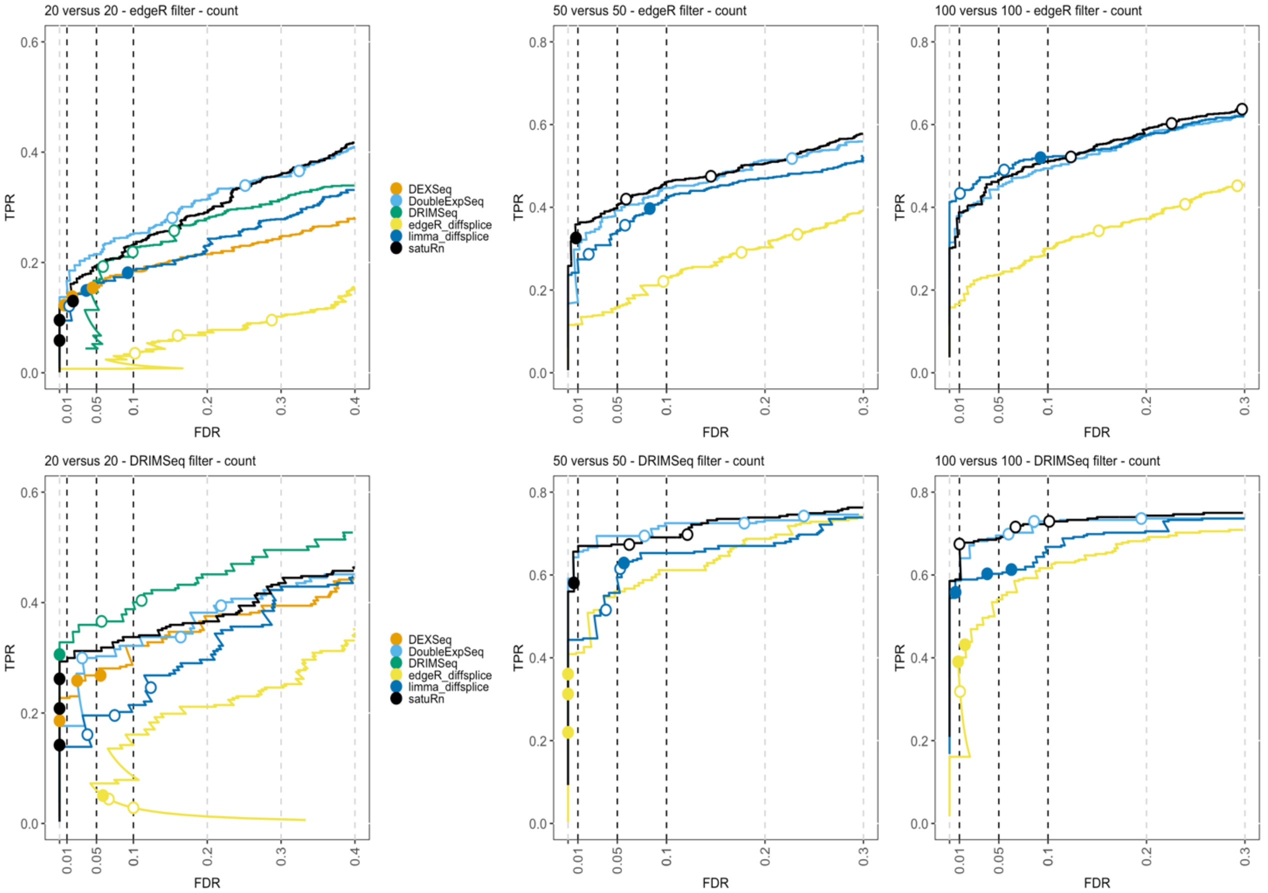

20 versus 20 - edgeR filter - scaledTPM
75 versus 75 - edgeR filter - scaledTPM
200 versus 200 - edgeR filter - scaledTPM
20 versus 20 - DRIMSeq filter - scaledTPM
75 versus 75 - DRIMSeq filter - scaledTPM
200 versus 200 - DRIMSeq filter - scaledTPM
TPR
FDR
0.0
0.2
0.4
0.6
0.00
0.25
0.50
0.75
1.00
0.01
0.05
0.1
0.2
0.3
0.4
DEXSeq
DoubleExpSeq
DRIMSeq
edgeR_diffsplice
limma_diffsplice
satuRn

Figure S8: Performance evaluation of DTU methods on the real scRNA-seq dataset by Tasic *et al.* FDR-TPR curves visualize the performance of each method by displaying the sensitivity of the method (TPR) with respect to the false discovery rate (FDR). The three circles on each curve represent working points when the FDR level is set at nominal levels of 1%, 5% and 10%, respectively. The circles are filled if the empirical FDR is equal or below the imposed FDR threshold. We generated three two-group comparisons of 20, 75 and 200 cells each (left, middle and right panel, respectively). The benchmark was performed both on the raw counts **(rows 1 and 2)** or on scaled transcripts-per-million (TPM) **(rows 3 and 4)** as obtained with the Bioconductor R package tximport[1]. We additionally adopted two different filtering strategies; an edgeR-based filtering **(rows 1 and 3)** and a DRIMSeq-based filtering **(rows 2 and 4)**. Overall, satuRn slightly outperforms DoubleExpSeq, the best tools from the literature. Note that the performance of DEXSeq is clearly lower. In addition, our method consistently controls the FDR close to its imposed nominal FDR threshold, while DoubleExpSeq becomes more liberal with increasing sample sizes. DEXSeq and DRIMSeq were omitted from the largest comparison (two groups with 75 cells and 200 cells each, respectively), as these methods do not scale to large datasets (Figure 1). NBSplice was omitted from all comparisons, as it does not converge on datasets with many zeros, such as scRNA-seq datasets.


20 versus 20 - edgeR filter - count
50 versus 50 - edgeR filter - count
100 versus 100 - edgeR filter - count
20 versus 20 - DRIMSeq filter - count
50 versus 50 - DRIMSeq filter - count
100 versus 100 - DRIMSeq filter - count
TPR
FDR
0.0
0.2
0.4
0.6
0.8
0.01
0.05
0.1
0.2
0.3
0.4
DEXSeq
DoubleExpSeq
DRIMSeq
edgeR_diffsplice
limma_diffsplice
satuRn

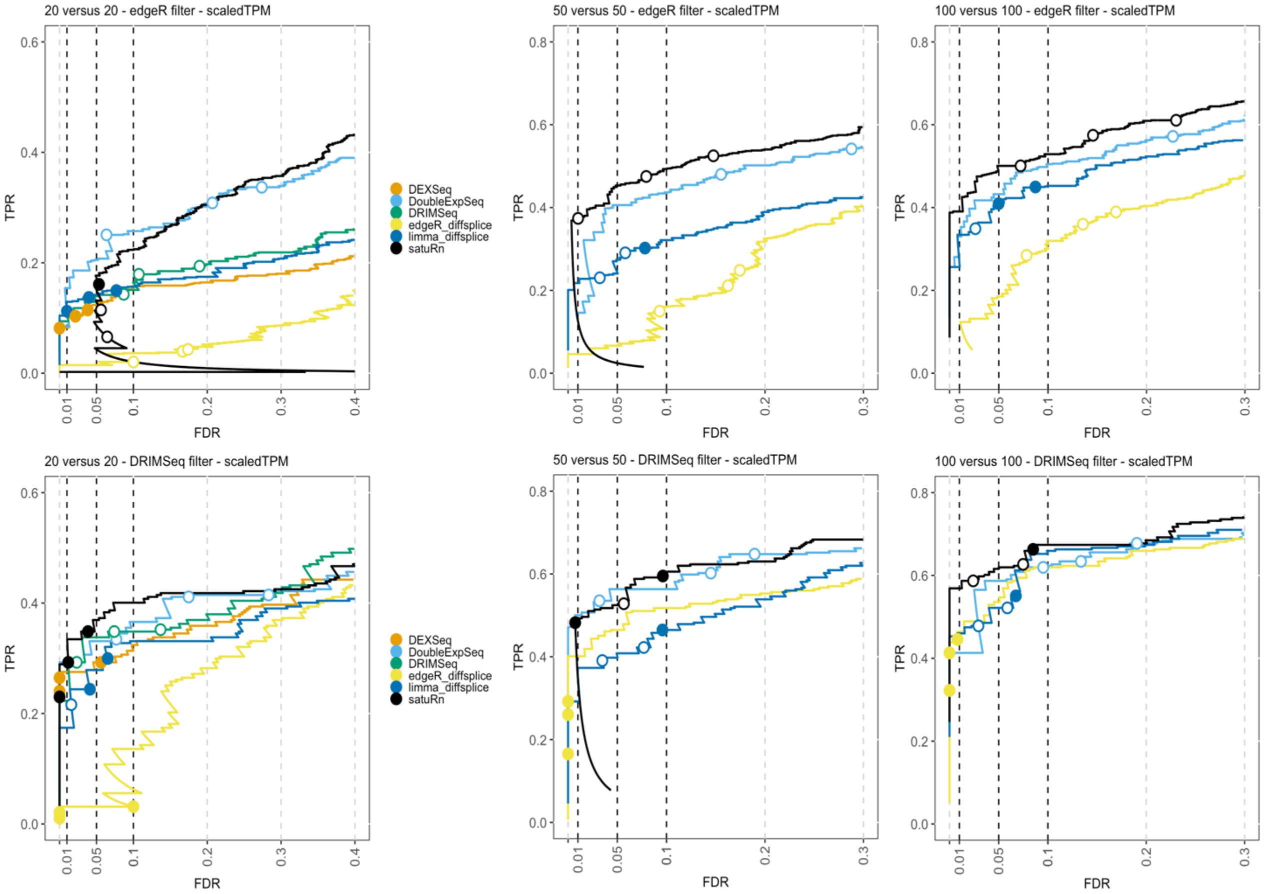

20 versus 20 - edgeR filter - scaledTPM
50 versus 50 - edgeR filter - scaledTPM
100 versus 100 - edgeR filter - scaledTPM
20 versus 20 - DRIMSeq filter - scaledTPM
50 versus 50 - DRIMSeq filter - scaledTPM
100 versus 100 - DRIMSeq filter - scaledTPM
TPR
FDR
0.0
0.2
0.4
0.6
0.8
0.01
0.05
0.1
0.2
0.3
0.4
DEXSeq
DoubleExpSeq
DRIMSeq
edgeR_diffsplice
limma_diffsplice
satuRn

Figure S9: Performance evaluation of DTU methods on the real scRNA-seq dataset by Darmanis *et al*. FDR-TPR curves visualize the performance of each method by displaying the sensitivity of the method (TPR) with respect to the false discovery rate (FDR). The three circles on each curve represent working points when the FDR level is set at nominal levels of 1%, 5% and 10%, respectively. The circles are filled if the empirical FDR is equal or below the imposed FDR threshold. We generated three two-group comparisons of 20, 50 and 100 cells each (left, middle and right panel, respectively). The benchmark was performed both on the raw counts **(rows 1 and 2)** or on scaled transcripts-per-million (TPM) **(rows 3 and 4)** as obtained with the Bioconductor R package tximport[1]. We additionally adopted two different filtering strategies; an edgeR-based filtering **(rows 1 and 3)** and a DRIMSeq-based filtering **(rows 2 and 4)**. Overall, the performance of satuRn is similar to DoubleExpSeq, the best tools from the literature. In addition, our method consistently controls the FDR close to its imposed nominal FDR threshold, while DoubleExpSeq becomes more liberal with increasing sample sizes. On the dataset with the smallest sample size, the FDR control of *satuRn* does become too strict.

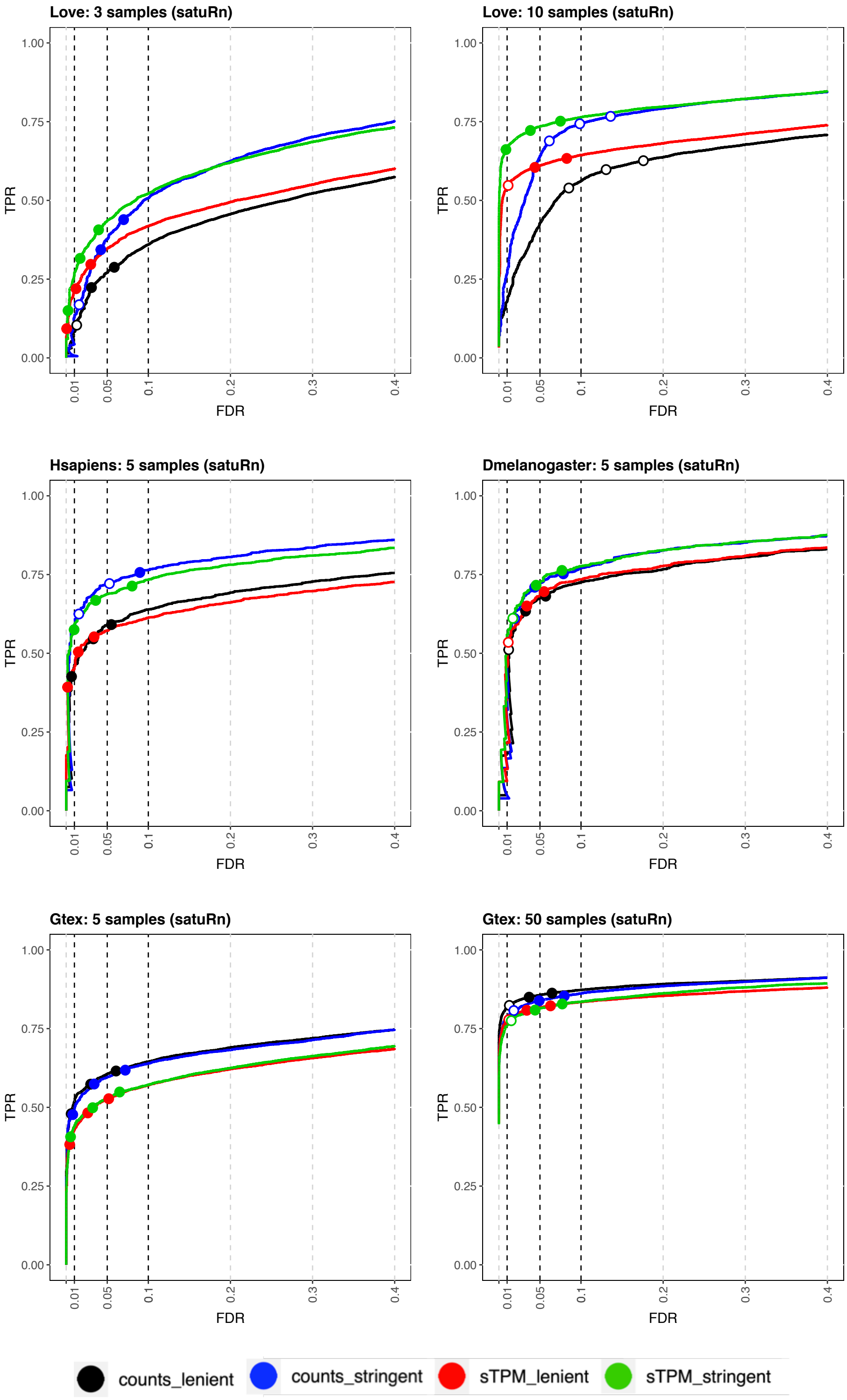


Figure S10: The effect of filtering and abundance metrics on the performance of satuRn in the different bulk RNA-seq benchmark datasets. The effect of filtering and abundance metric differs between the different datasets. **Top row:** For the dataset by Love *et al.*, filtering more stringently improves performance. In addition, both performance and FDR control are much better when using scaledTPM abundances, as compared to using counts. **Middle row:** For the simulated bulk datasets by Van den Berge *et al.*[40], we also observe a positive effect of stringent filtering, however, the difference between scaledTPM and raw count abundances is negligible. **Bottom row:** For GTEx bulk dataset, the effect of filtering is limited. However, using counts performs considerably better than using scaledTPM abundances**.**

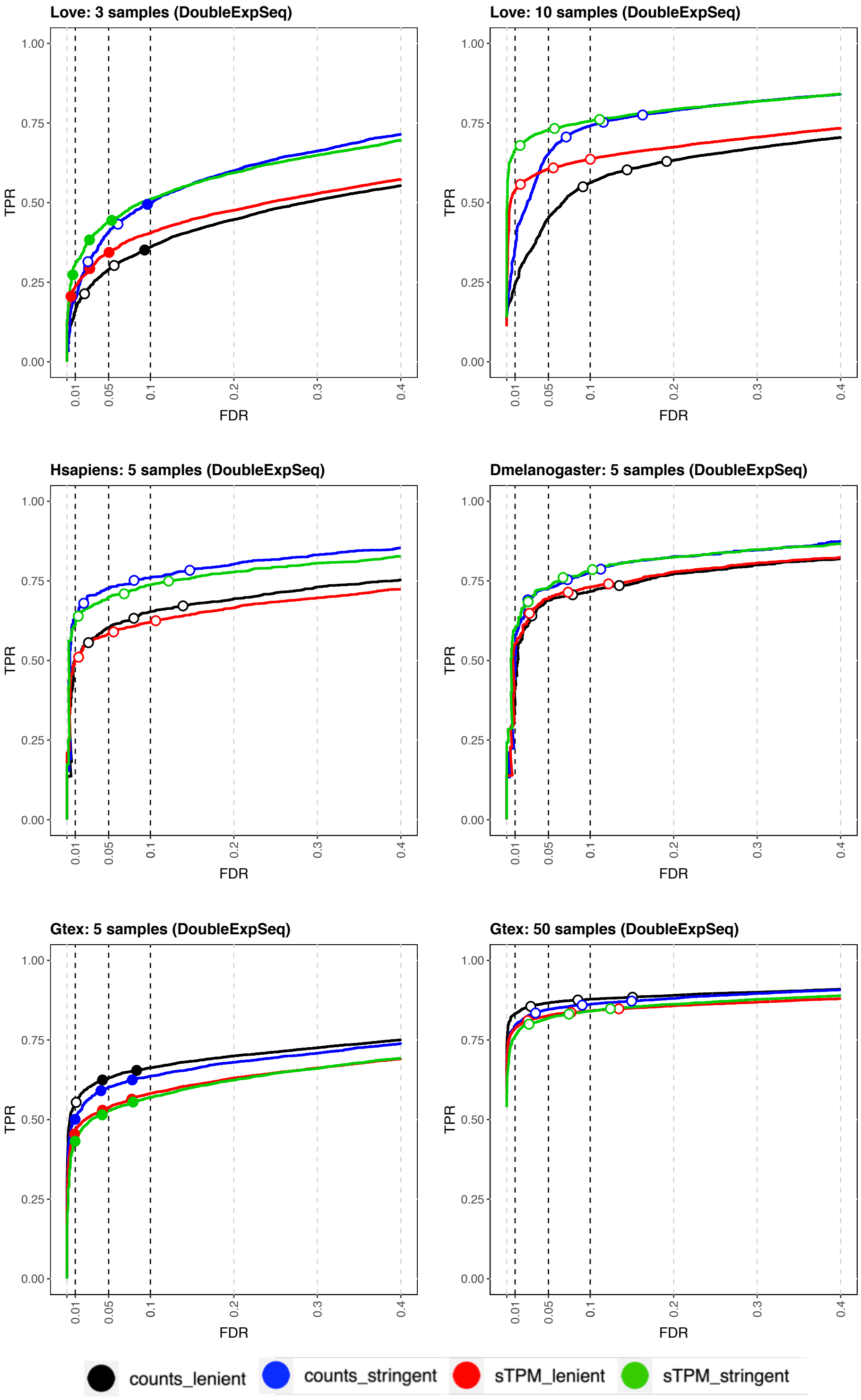


Figure S11: The effect of filtering and abundance metrics on the performance of DoubleExpSeq in the different bulk RNA-seq benchmark datasets. The effect of filtering and abundance metric differs between the different datasets. The observed effects correspond strongly with the effects of filtering and abundance metrics on satuRn (figure S10) and limma diffsplice (not shown). **Top row:** For the dataset by Love *et al.*, filtering more stringently improves performance. In addition, both performance and FDR control are much better when using scaledTPM abundances, as compared to using counts. **Middle row:** For the simulated bulk datasets by Van den Berge *et al.*[40], we also observe a positive effect of stringent filtering, however, the difference between scaledTPM and raw count abundances is negligible. **Bottom row:** For GTEx bulk dataset, the effect of filtering is limited. However, using counts performs considerably better than using scaledTPM abundances**.**

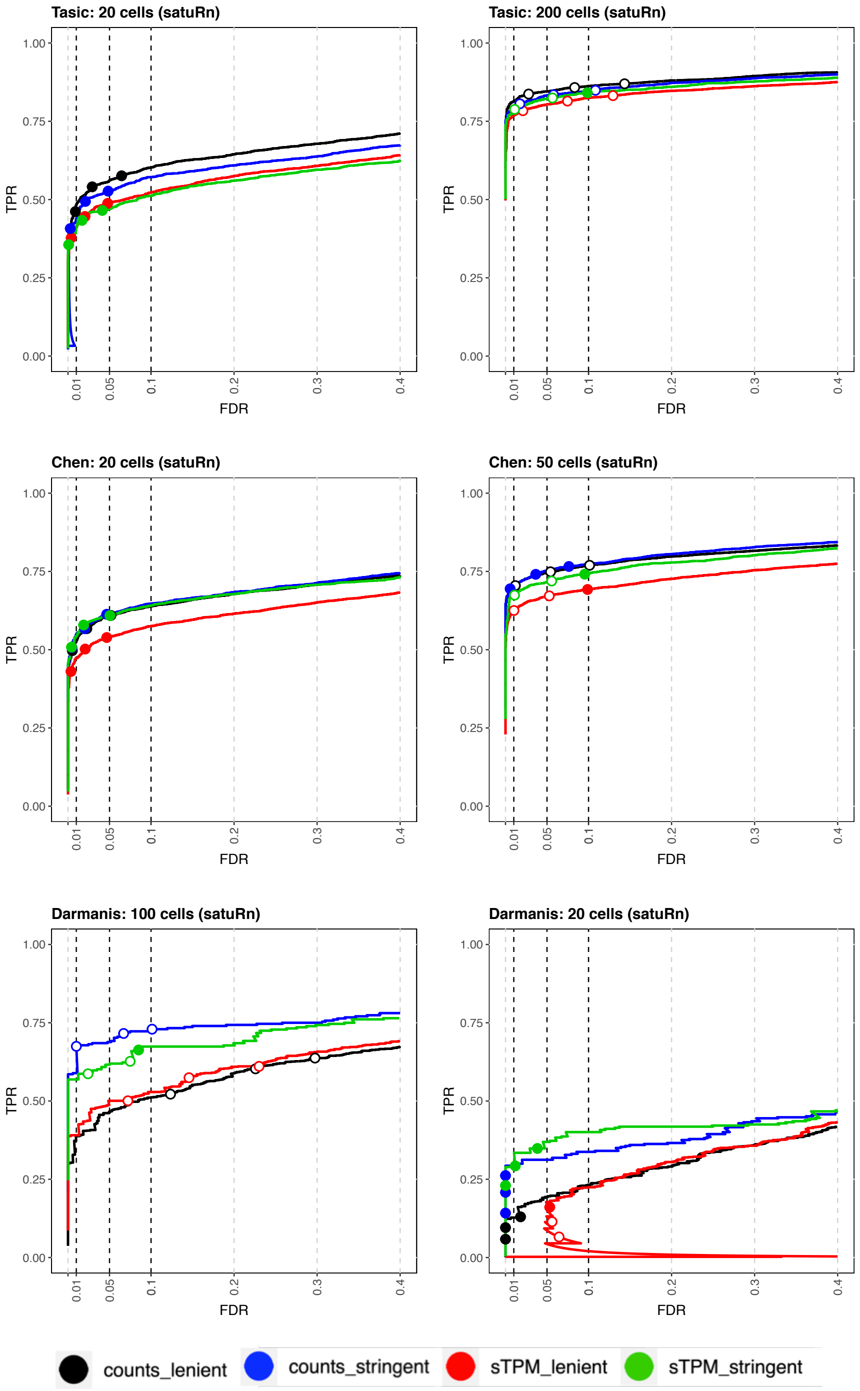


Figure S12: The effect of filtering and abundance metrics on the performance of satuRn in the different single-cell RNA-seq benchmark datasets. For the Tasic **(top row)** and Chen **(middle row)** datasets, the effects of filtering are limited and using counts performs slightly better than using *scaledTPM* abundances. For the Darmanis dataset **(bottom row)**, which is the sparsest dataset (see Figure S30 and table S1), a positive impact of the more stringent DRIMSeq filtering criterion is observed.

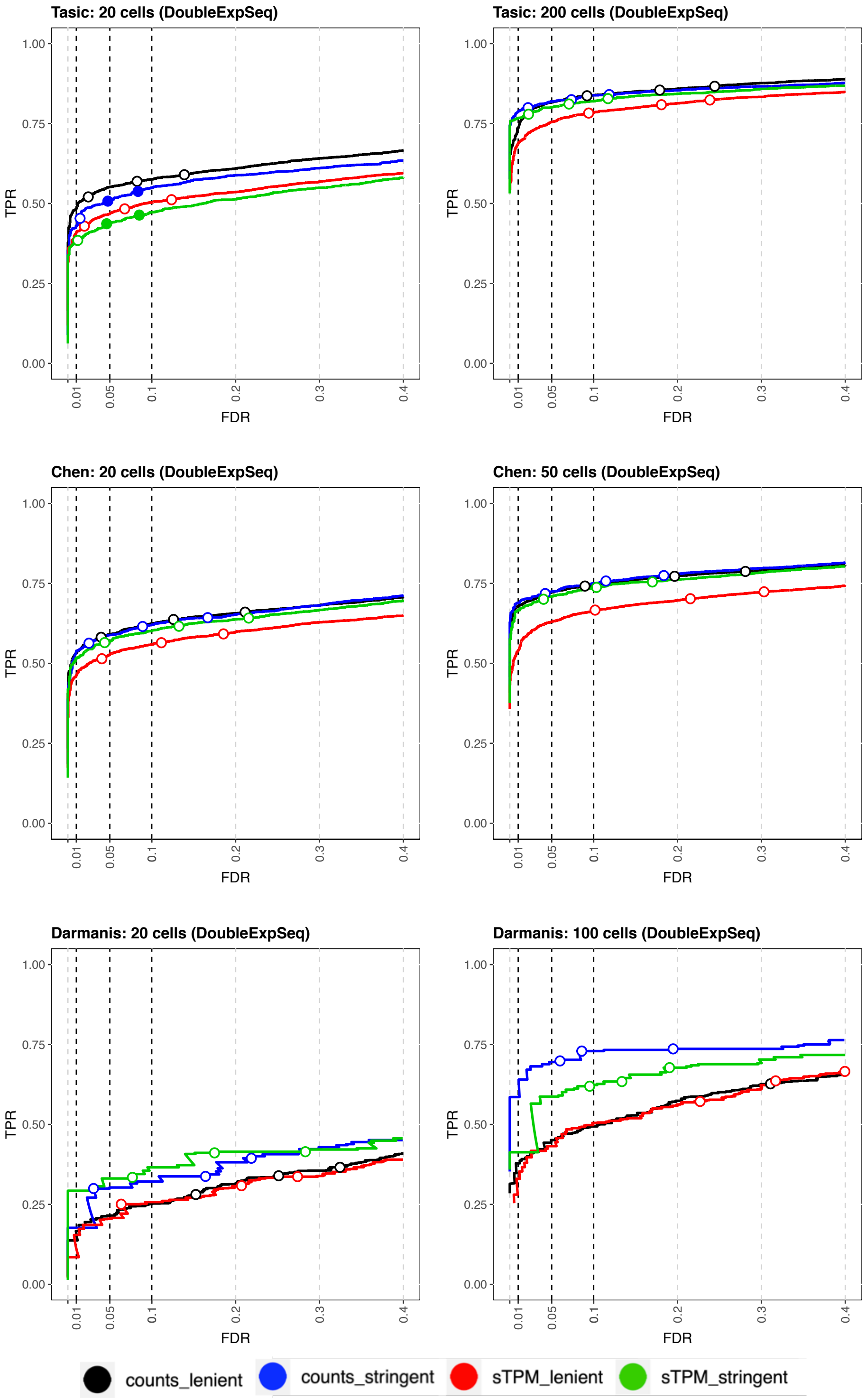


Figure S13: The effect of filtering and abundance metrics on the performance of DoubleExpSeq in the different single-cell RNA-seq benchmark datasets. The observed effects of filtering and abundance metric correspond strongly with the effects observed for on satuRn (figure S12) and limma diffsplice (not shown). For the Tasic **(top row)** and Chen **(middle row)** datasets, the effects of filtering are limited and using counts performs slightly better than using *scaledTPM* abundances. For the Darmanis dataset **(bottom row)**, which is the sparsest dataset (see Figure S30 and table S1), a positive impact of the more stringent DRIMSeq filtering criterion is observed.

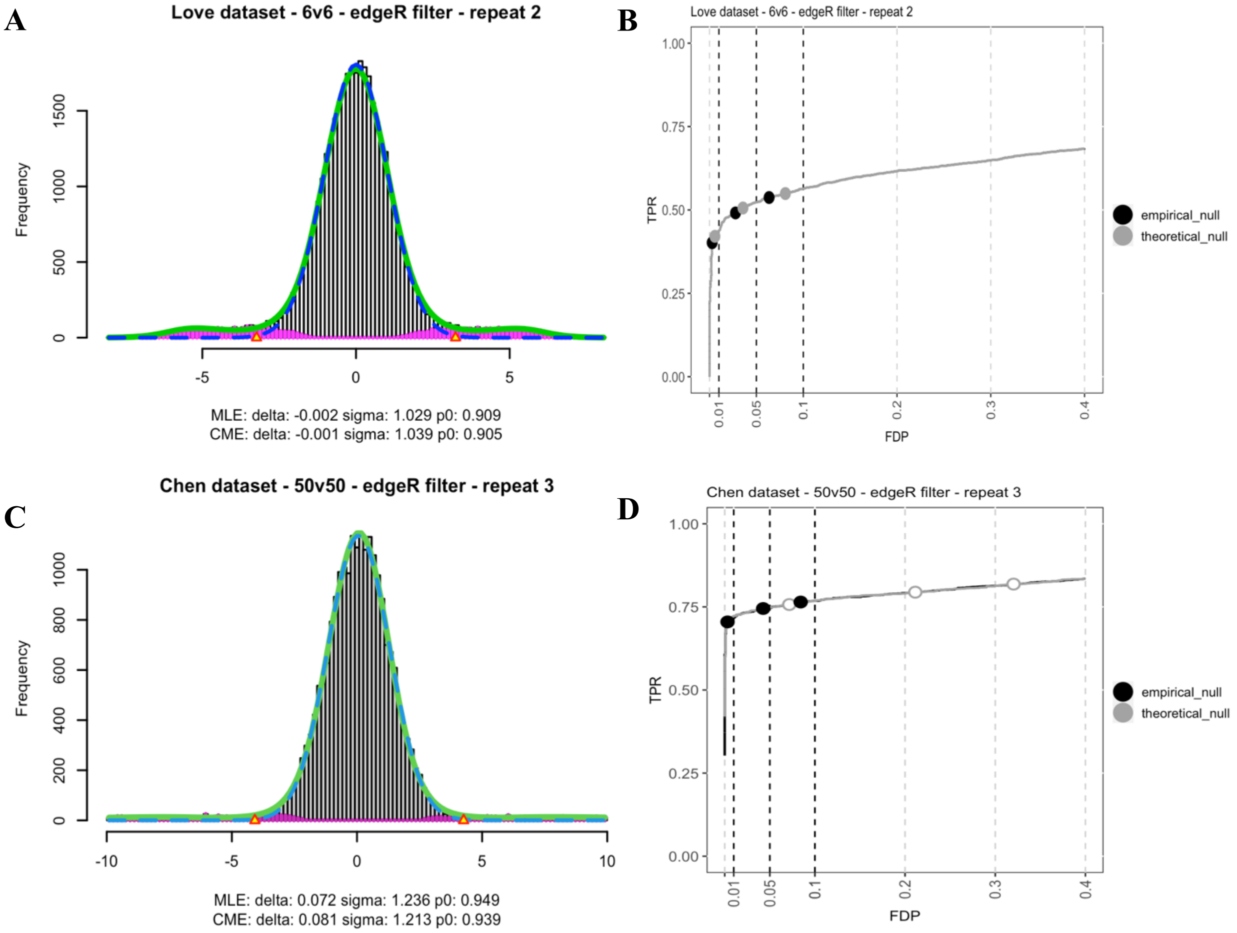


Figure S14: The effect of using an empirical null distribution on the false discovery control of satuRn. Panel A: Empirical distribution of the satuRn test statistics in one of the bulk transcriptomics benchmark datasets adapted from Love *et al***.** The test statistics are z-scores, calculated from satuRn p-values as described in formula 5 (see Methods). This benchmark dataset is constructed to have 15% DTU transcripts and thus 85% non-DTU or null transcripts. The *z*-scores corresponding to the null transcripts are expected to follow a standard normal distribution (mean = 0, standard deviation = 1). This corresponds well with the maximum likelihood estimates (MLE) for the mean and variance of the empirical null distribution (mean = -0.002, standard deviation = 1.029) as obtained with the *locfdr* package[2]. In brief, these estimates are obtained by assuming that the *z*-scores of all transcripts follow a mixture distribution, where the *z*-scores of the null transcripts are expected to follow a normal distribution and the *z*-scores of the DTU transcripts follow some other distribution. Two models are fitted to the *z*-scores. The blue dashed curve is a normal distribution that is fitted to the mid 50% of the *z*-scores, which are assumed to originate from null genes, thus representing the estimated empirical null component densities. The MLE and central matching estimates (CME) for the mean and standard deviation of the estimated empirical null distribution are provided in the caption at the bottom of the plot. Finally, the green solid curve represents the estimated marginal density across all *z*-scores and is obtained by fitting a spline model to the histogram counts. **Panel B:** FDP-TPR curve for the bulk transcriptomics benchmark dataset. As the theoretical null distribution and the empirical null distribution are virtually identical, we observe a negligible difference between both strategies, both in terms of performance and FDR control. **Panel C:** Empirical distribution of the satuRn test statistics in one of the single-cell benchmark datasets adapted from Chen *et al*. Again, most of these z-scores are expected to follow a standard normal distribution as this benchmark dataset is also constructed to have 15% DTU transcripts. However, the empirical distribution is considerably wider than expected (standard deviation = 1.236). We additionally observe a small shift of the distribution (mean = 0.072). **Panel D:** FDP-TPR curve for the single-cell benchmark dataset. While the inference for satuRn is overly liberal when working under the theoretical null, FDR control is restored by adopting the wider empirical null distribution. Note that the performance (the ranking of the transcripts according to their p-values) will only be affected when the empirical null distribution is shifted with respect to the theoretical null (i.e., when the MLE for the mean is clearly different from zero), which was not the case in this example nor in any other dataset from our analyses.

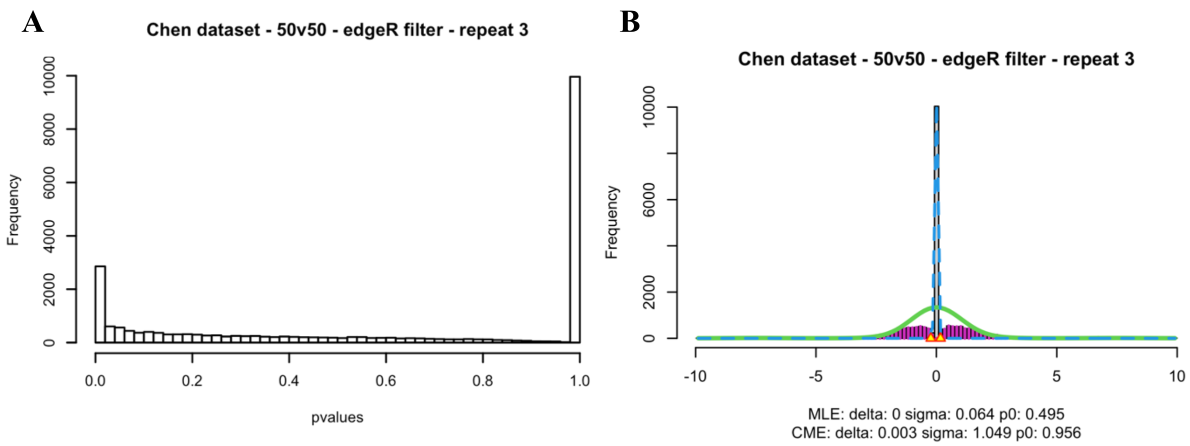


Figure S15: Adopting an empirical null distribution to improve FDR control is infeasible for DoubleExpSeq. Panel A: Distribution of the p-values from a DoubleExpSeq analysis in one of the single-cell benchmark datasets adapted from Chen *et al*. We immediately observe the large spike of p-values equal to 1, which distorts the p-value distribution. In addition, the p-values in the mid-range (e.g., from 0.1 to 0.9), which are expected to be uniformly distributed, are skewed towards smaller values, which underlies the overly liberal results of DoubleExpSeq in our single-cell benchmarks. **Panel B:** The corresponding empirical distribution of the DoubleExpSeq test statistics. The test statistics are z-scores, calculated from the original DoubleExpSeq p-values as described in formula 5 (see Methods). As all our benchmark datasets are constructed to have 15% DTU transcripts and thus 85% non-DTU or null transcripts, most of these z-scores are expected to follow a standard normal distribution (mean = 0, standard deviation =1). However, given the pathological distribution of the p-values it is not feasible to properly estimate the empirical null distribution, as also clearly shown by the widely different parameter estimates obtained using the two estimation frameworks implemented in the *locfdr* R package[2]; compare the estimates between MLE (maximum likelihood estimation) and CME (central matching estimation). For more details on the *locfdr* figures we refer to the caption of figure S10.

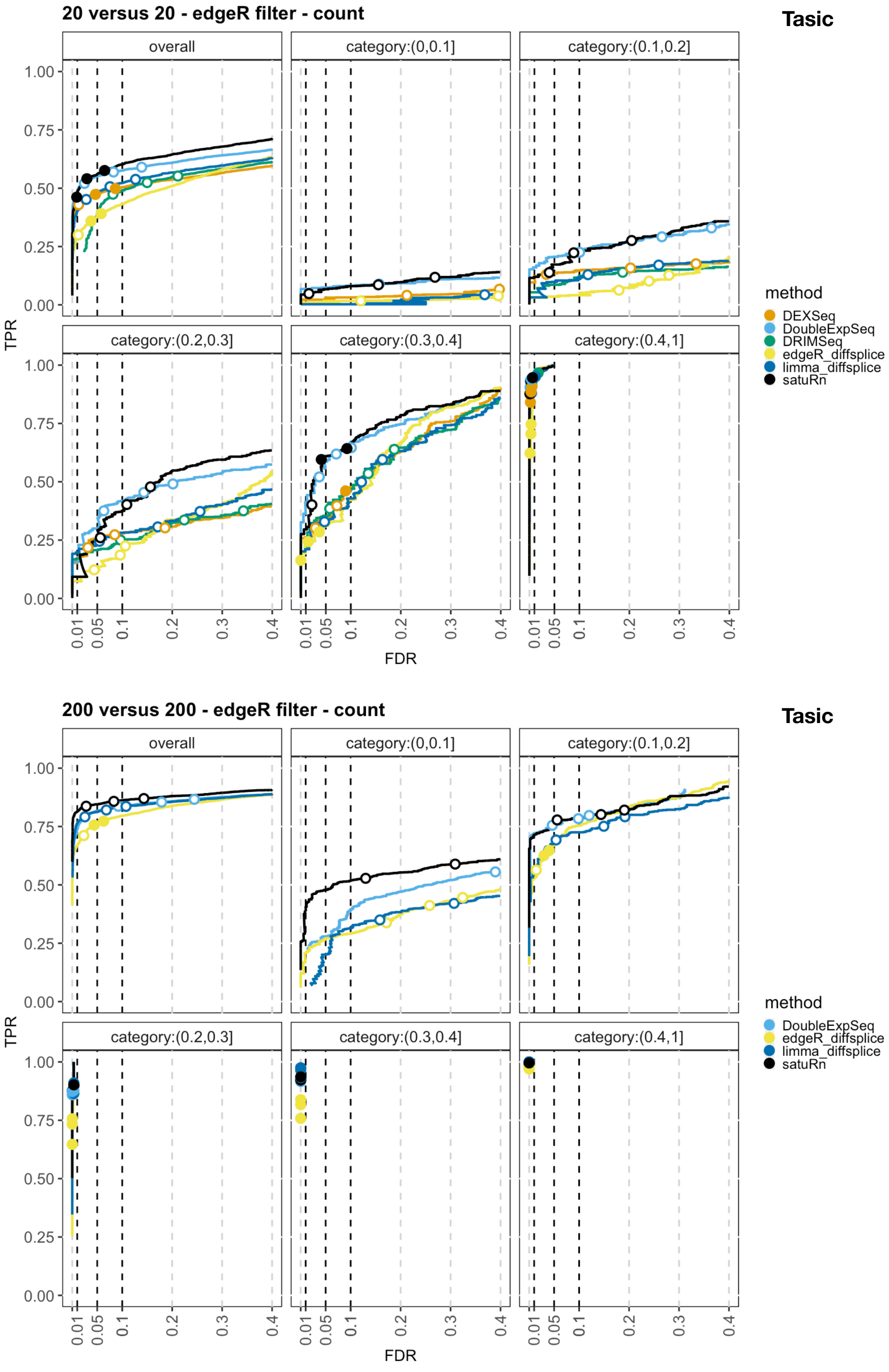


Figure S16: Performance evaluation on the real scRNA-seq dataset by Tasic *et al.*, stratified by the magnitude of the DTU signal. The FDR-TPR curves are stratified on the difference in the observed average transcript usage between the two groups of cells. The difference in the fraction of transcript usage between the two groups is indicated in the panel headers. **Panel A: Dataset with 20 cells per group.** The ability of all methods to detect DTU decreases when the strength of the DTU signal decreases. Notably, satuRn and DoubleExpSeq are more successful in detecting small differences as compared to the other methods. **Panel B: Dataset with 200 cells per group.** Given the larger number of cells, the performance of all methods is increased compared to panel A. Again, satuRn and DoubleExpSeq are the most successful in detecting small differences in transcript usage.

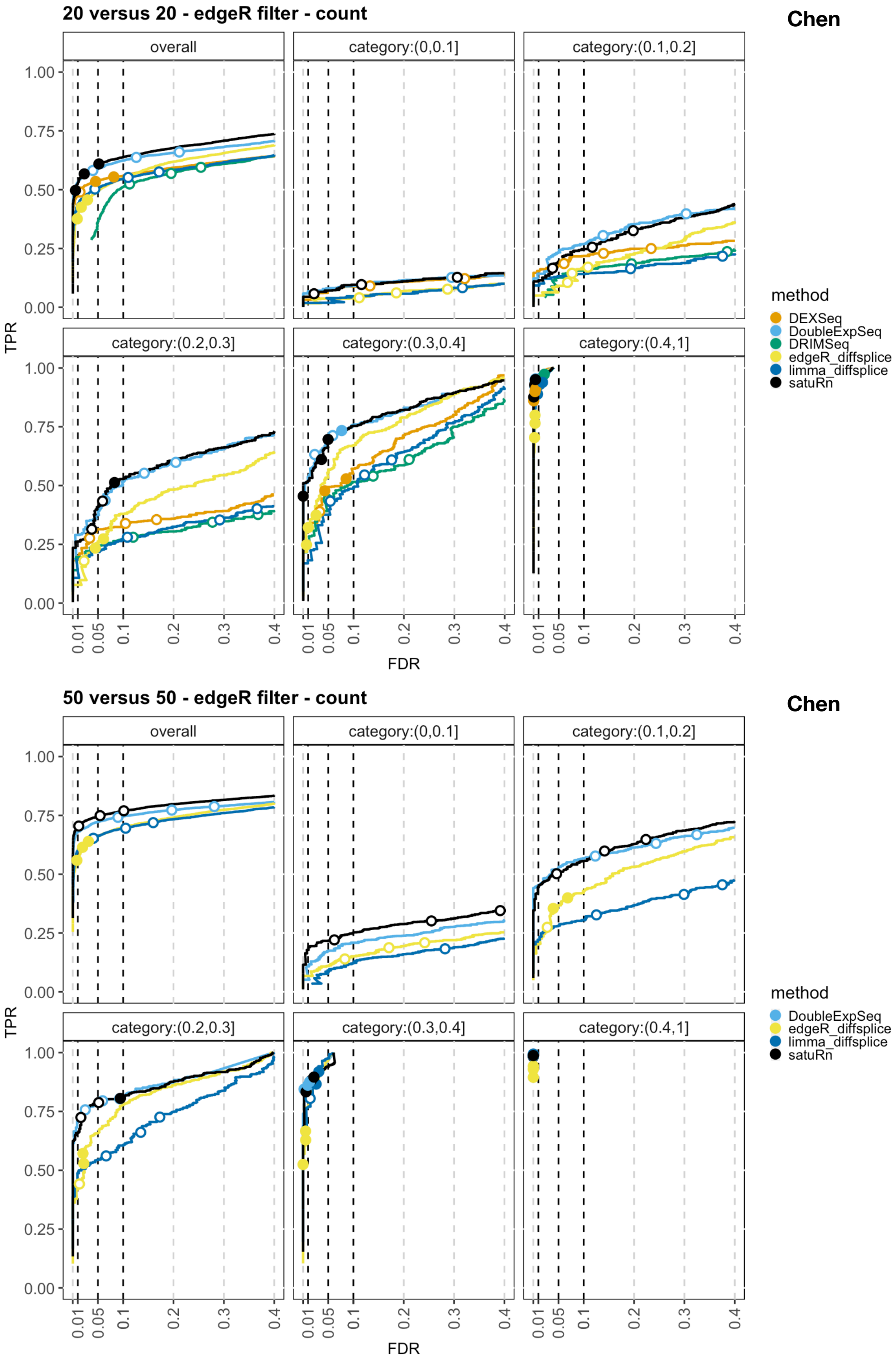


Figure S17: Performance evaluation on the real scRNA-seq dataset by Chen *et al.*, stratified by the magnitude of the DTU signal. The FDR-TPR curves are stratified on the difference in the observed average transcript usage between the two groups of cells. The difference in the fraction of transcript usage between the two groups is indicated in the panel headers. The same patterns are observed as for the Tasic *et al.* dataset from Figure S16. **Panel A: Dataset with 20 cells per group.** The ability of all methods to detect DTU decreases when the strength of the DTU signal decreases. Notably, satuRn and DoubleExpSeq are more successful in detecting small differences as compared to the other methods. **Panel B: Dataset with 50 cells per group.** Given the larger number of cells, the performance of all methods is increased compared to panel A. Again, satuRn and DoubleExpSeq are the most successful in detecting small differences in transcript usage.

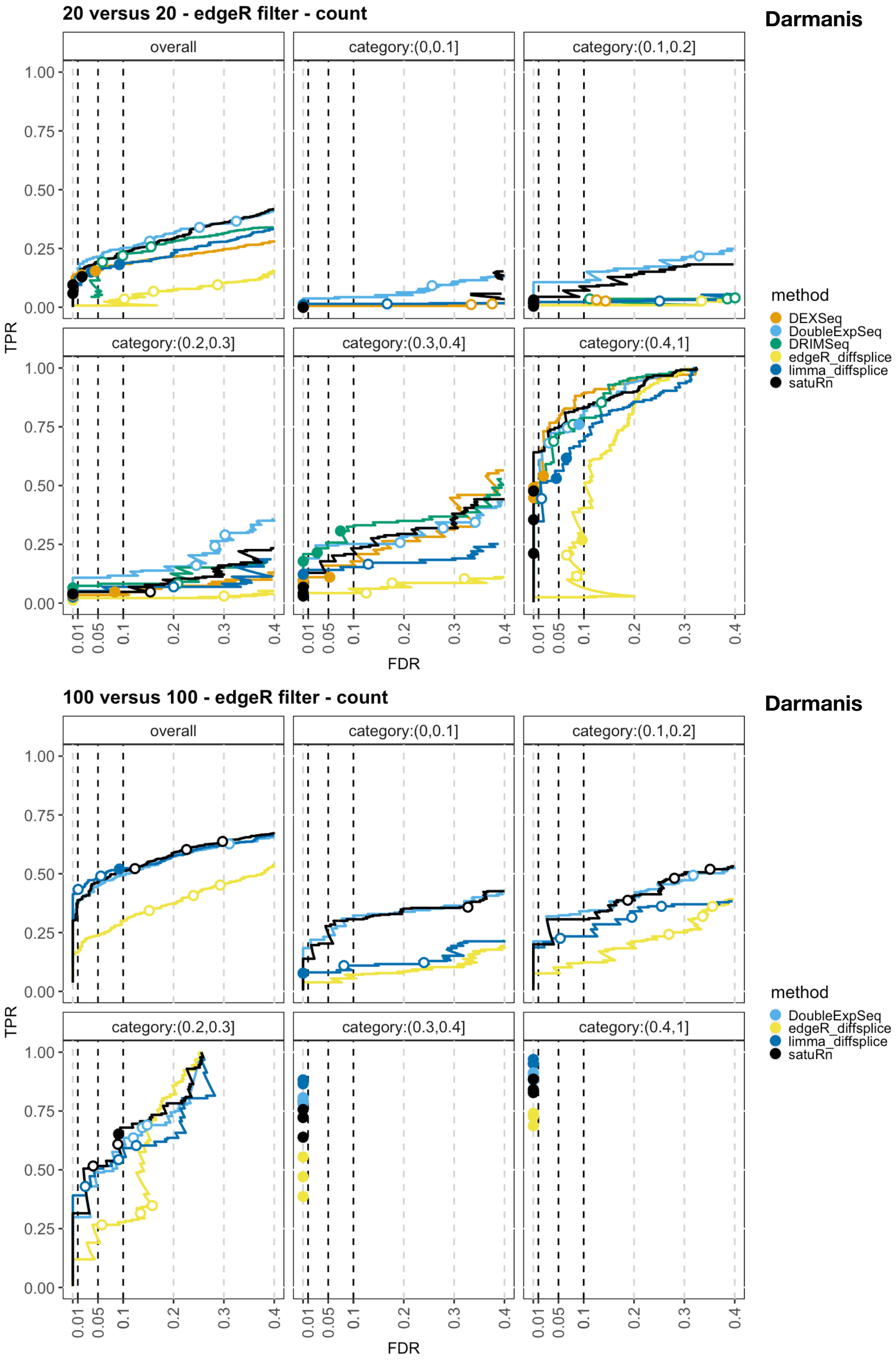


Figure S18: Performance evaluation on the real scRNA-seq dataset by Darmanis *et al.*, stratified by the magnitude of the DTU signal. The FDR-TPR curves are stratified on the difference in the observed average transcript usage between the two groups of cells. The difference in the fraction of transcript usage between the two groups is indicated in the panel headers. The same patterns are observed as for the Tasic *et al.* and Chen *et al.* datasets from Figures S16 and S17. **Panel A: Dataset with 20 cells per group.** The ability of all methods to detect DTU decreases when the strength of the DTU signal decreases. Notably, satuRn and DoubleExpSeq are more successful in detecting small differences as compared to the other methods. **Panel B: Dataset with 100 cells per group.** Given the larger number of cells, the performance of all methods is increased compared to panel A. Again, satuRn and DoubleExpSeq are the most successful in detecting small differences in transcript usage.

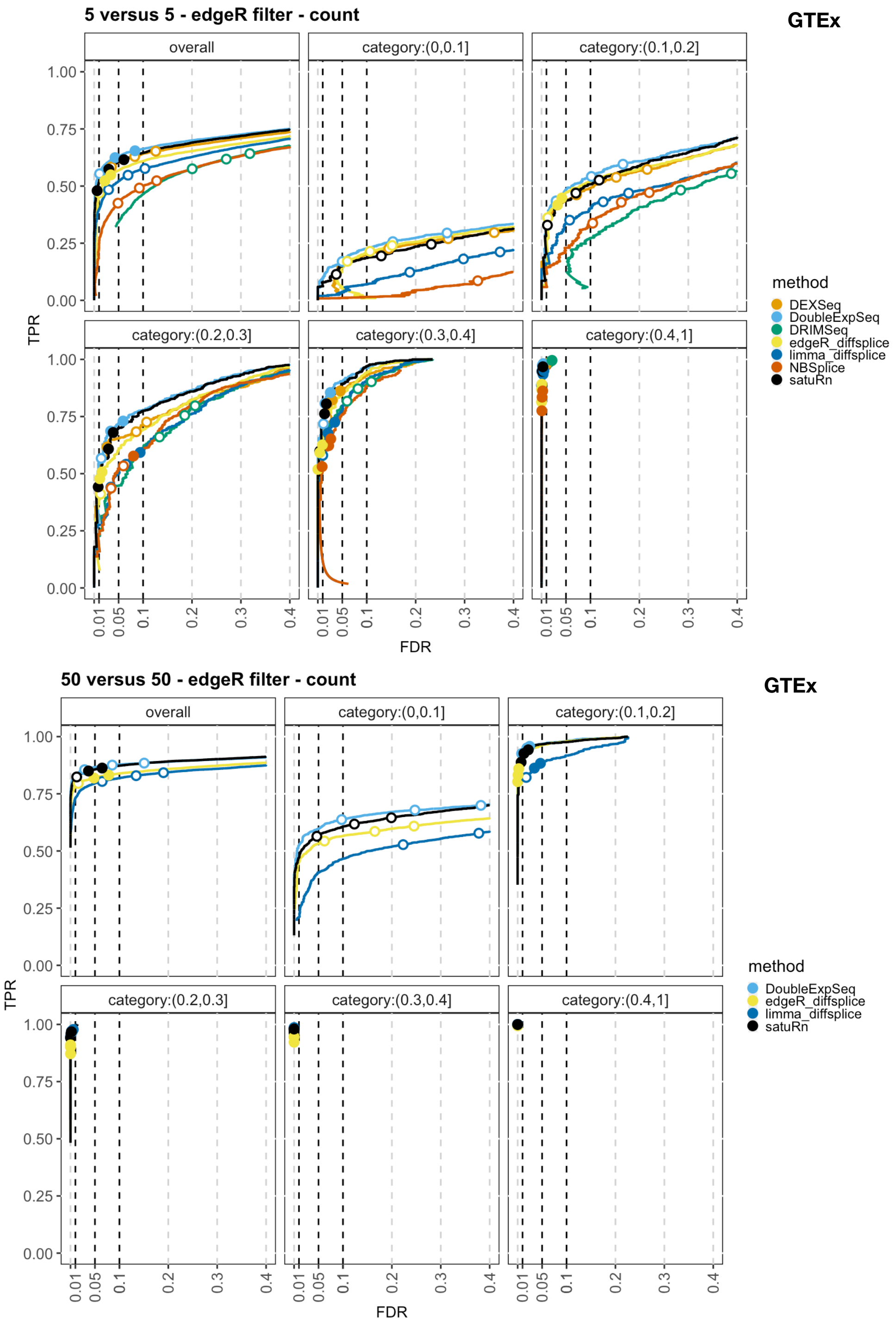


Figure S19: Performance evaluation on the GTEx bulk RNA-seq dataset, stratified by the magnitude of the DTU signal. The FDR-TPR curves are stratified on the difference in the observed average transcript usage between the two groups of cells. The difference in the fraction of transcript usage between the two groups is indicated in the panel headers. The same patterns are observed as for the single-cell datasets from Figures S16-S18. **Panel A: Dataset with 5 samples per group.** The ability of all methods to detect DTU decreases when the strength of the DTU signal decreases. satuRn and DoubleExpSeq are more successful in detecting small differences as compared to the other methods. **Panel B: Dataset with 50 samples per group.** Given the larger number of cells, the performance of all methods is increased compared to panel A. Again, satuRn and DoubleExpSeq are the most successful in detecting small differences in transcript usage. Given the larger sequencing depth of bulk RNA-seq data, fewer observations per group are required to detect small differences in transcript usage as compared to single-cell datasets.

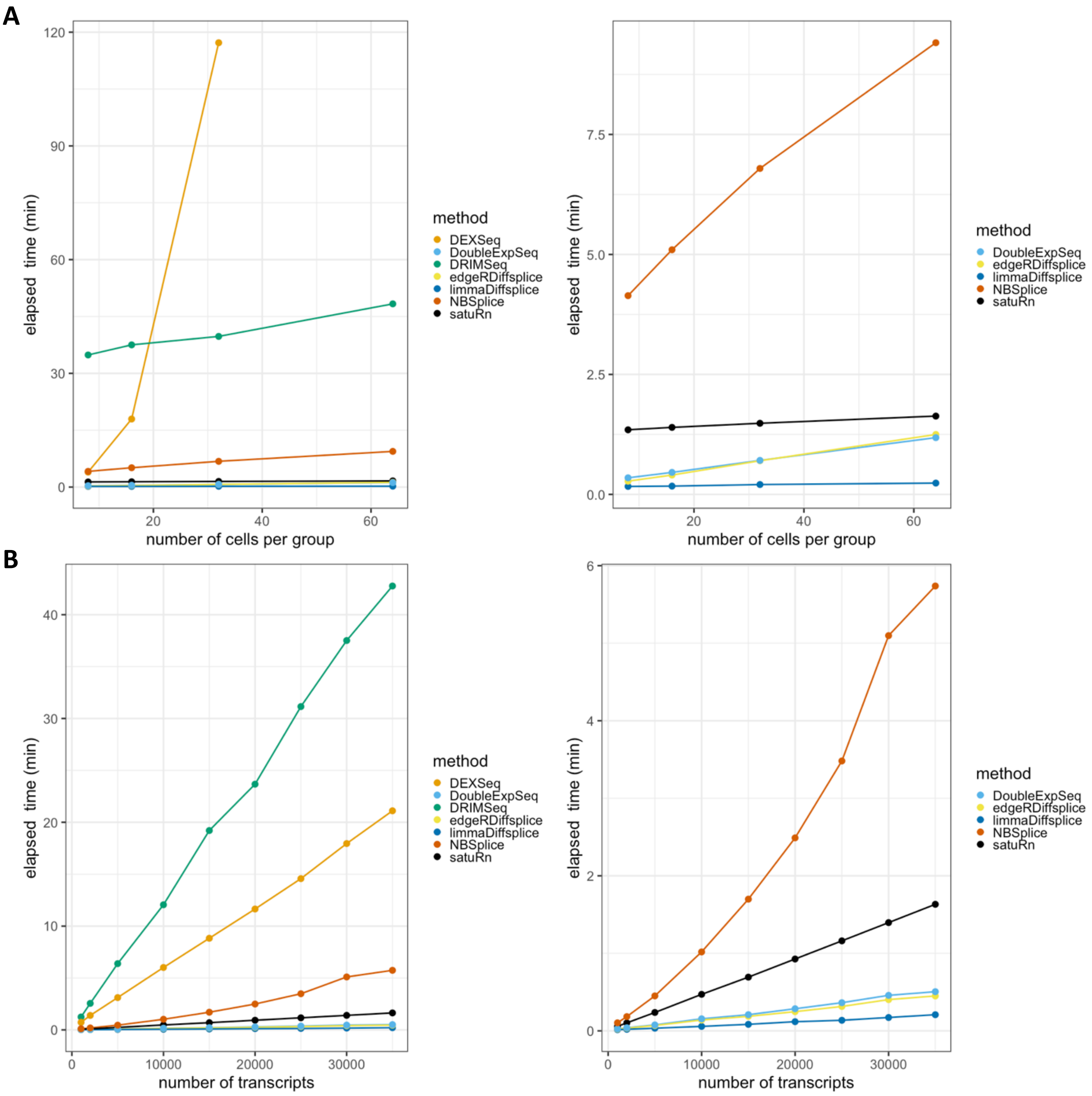


Figure S20: Scalability evaluation on bulk RNA-seq data. A: Runtime with respect to the number of samples in a bulk RNA-Seq dataset. **Left panel:** DRIMSeq and especially DEXSeq scale poorly with the number of cells in the dataset. **Right panel:** Detailed plot of the fastest methods. satuRn scales linearly with increasing numbers of samples, with a slope that is comparable to that of limma diffsplice. As such, satuRn can perform a DTU analysis on a dataset with two groups of 64 samples each and 30,000 transcripts in less than three minutes. For all sample sizes, the number of transcripts in the datasets were set at 30,000. Note that BANDITS was not included in this analysis as we did not obtain equivalence class counts for the GTEx bulk dataset. NBSplice, which was not included in the single-cell scalability benchmark of Figure 5 because it fails to converge on datasets with a large proportion of zero counts, is included here. **B: Runtime with respect to the number of transcripts in a bulk RNA-seq dataset**. **Left panel:** DEXSeq and DRIMSeq scale poorly to the number of transcripts in the dataset. **Right panel:** Detailed plot of the remaining methods. satuRn scales linearly with increasing numbers of transcripts, but with a steeper slope than edgeR diffsplice, DoubleExpSeq and limma diffsplice. The number of samples in the dataset was set fixed to two groups of 16 samples. All scalability benchmarks were run on a single core.

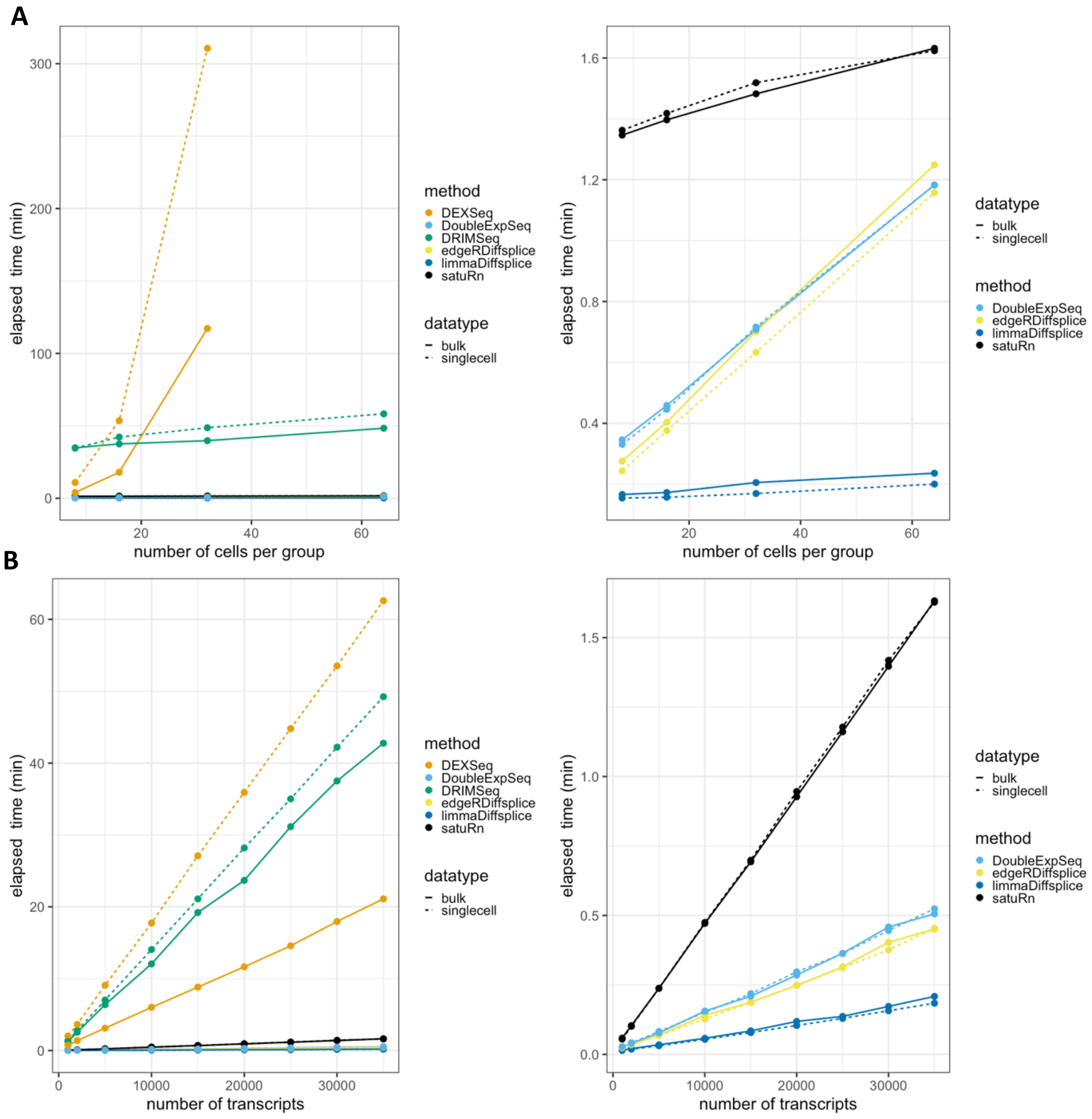


Figure S21: Comparison of the scalability profiles between bulk RNA-seq and scRNA-seq data. A: Runtime with respect to the number of cells/samples in the dataset. **Left panel:** The scalability of the different DTU tools on bulk data is indicated with a full line, while the scalability on single-cell data is displayed with a dashed line. A large effect between both data types was only observed for DEXSeq, which scales considerably worse on single-cell data, suggesting that the estimation of the GLM parameters is slower with sparse data. However, as the scalability profile of DEXSeq is quadratic with respect to the number of cells/samples in the data, it is still infeasible to adopt DEXSeq in datasets with many cells/samples, e.g., an analysis with 32 cells in each group takes approximately two hours. **Right panel:** detailed plot of the fastest methods. **B: Runtime with respect to the number of transcripts in the dataset**. The scalability of the different DTU tools on bulk data is indicated with a full line, while the scalability on single-cell data is displayed with a dashed line. Again, the largest difference in scalability between bulk and single-cell data was observed for DEXSeq. **Right panel:** detailed plot of the fastest methods.

Comparison	Cell type 1 (ALM)	Cell type 2 (VISp)	DoubleExpSeq FDR	Limma FDR	Limma Empirical FDR
1	Cpa6 Gpr88	Batf3	2142	3602	169
2	Cbln4 Fezf2	Col27a1	644	468	297
3	Cpa6 Gpr88	Col6a1 Fezf2	335	1029	77
4	Gkn1 Pcdh19	Col6a1 Fezf2	1878	2861	58
5	Lypd1 Gpr88	Hsd11b1 Endou	829	1411	249
6	Tnc	Hsd11b1 Endou	4580	4819	341
7	Tmem163 Dmrtb1	Hsd11b1 Endou	3388	5603	176
8	Tmem163 Arhgap25	Whrn Tox2	455	1387	166

Figure S22: Number of differentially used transcripts as identified by DoubleExpSeq and limma diffsplice. The first three columns indicate the comparisons between ALM cell types (column 2) and VISp cell types (column 3), respectively. Column 4 indicates the number of differentially used transcripts as identified by DoubleExpSeq. Column 5 indicates the number of differentially used transcripts as identified by a limma diffsplice analysis with default settings. Column 6 displays the number of differentially used transcripts found by limma diffsplice after correcting for deviations between the theoretical and empirical null distributions.

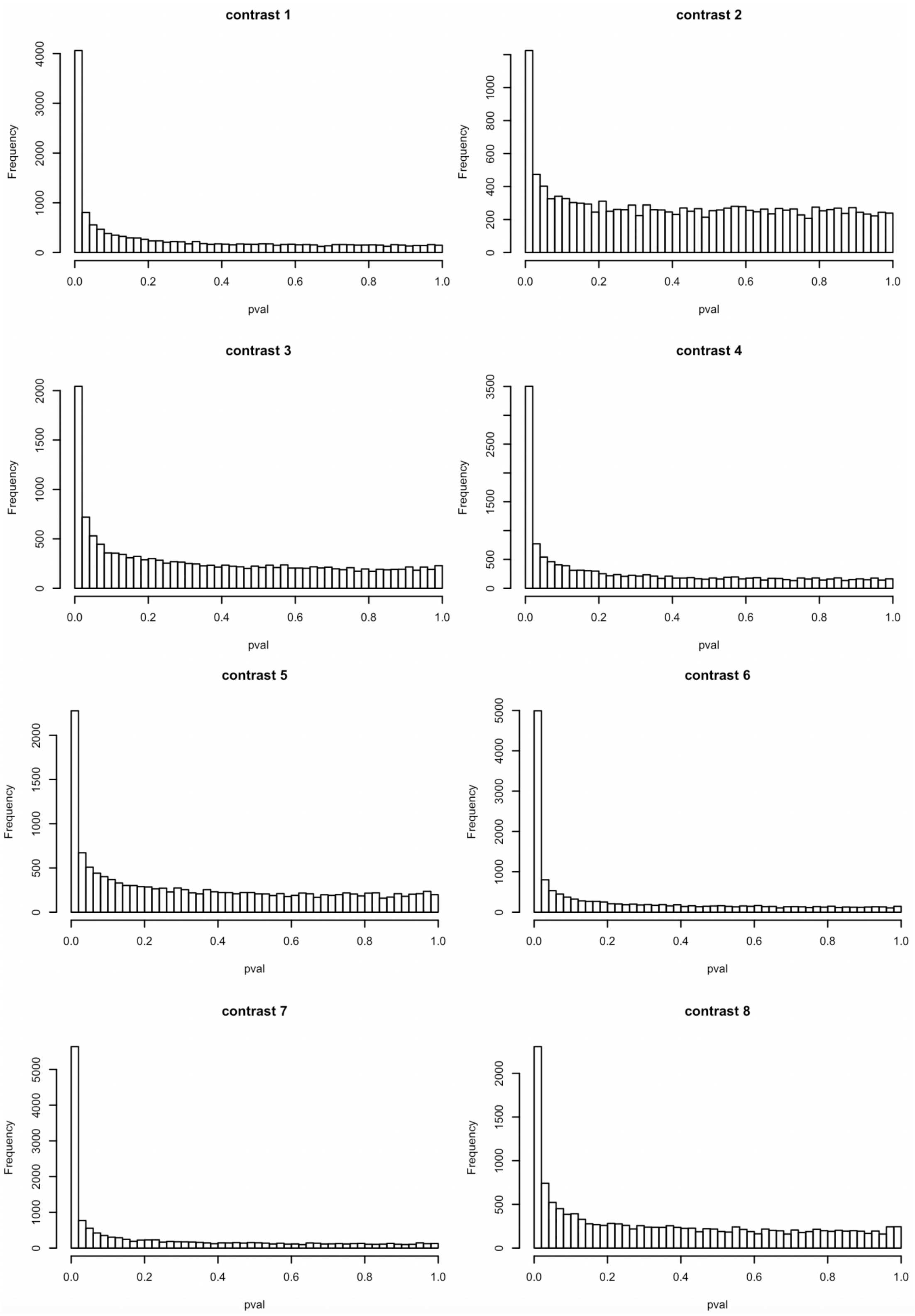


Figure S23: Histograms of the p-values from limma diffsplice. From these histograms, the huge number of DTU transcripts identified by limma diffsplice become apparent. Note that the general tendency of limma diffsplice for smaller p-values is better visible when converting the p-values into z-scores (see Figure S13).

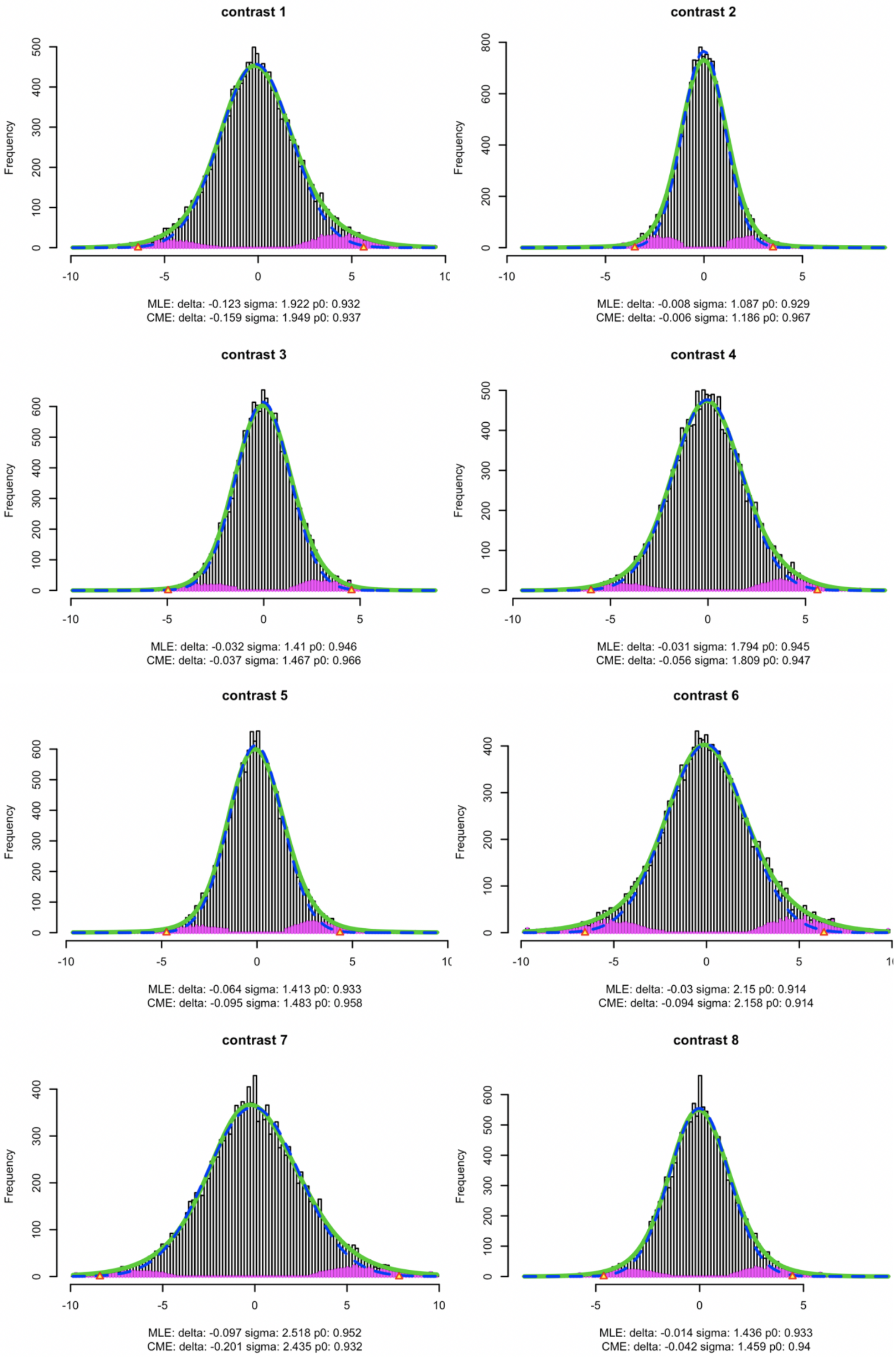


Figure S24: Empirical distribution of the limma diffsplice test statistics. The test statistics are z-scores, calculated from limma diffsplice p-values as described in formula 5. Theoretically, these z-scores are expected to follow a standard normal distribution (mean = 0, standard deviation =1). Here, however, the empirical distributions are considerably wider (standard deviation > 1), as indicated underneath the plots. This indicates that the results returned by limma diffsplice in this case study are overly liberal. For more details on the *locfdr* figures we refer to the caption of figure S14.

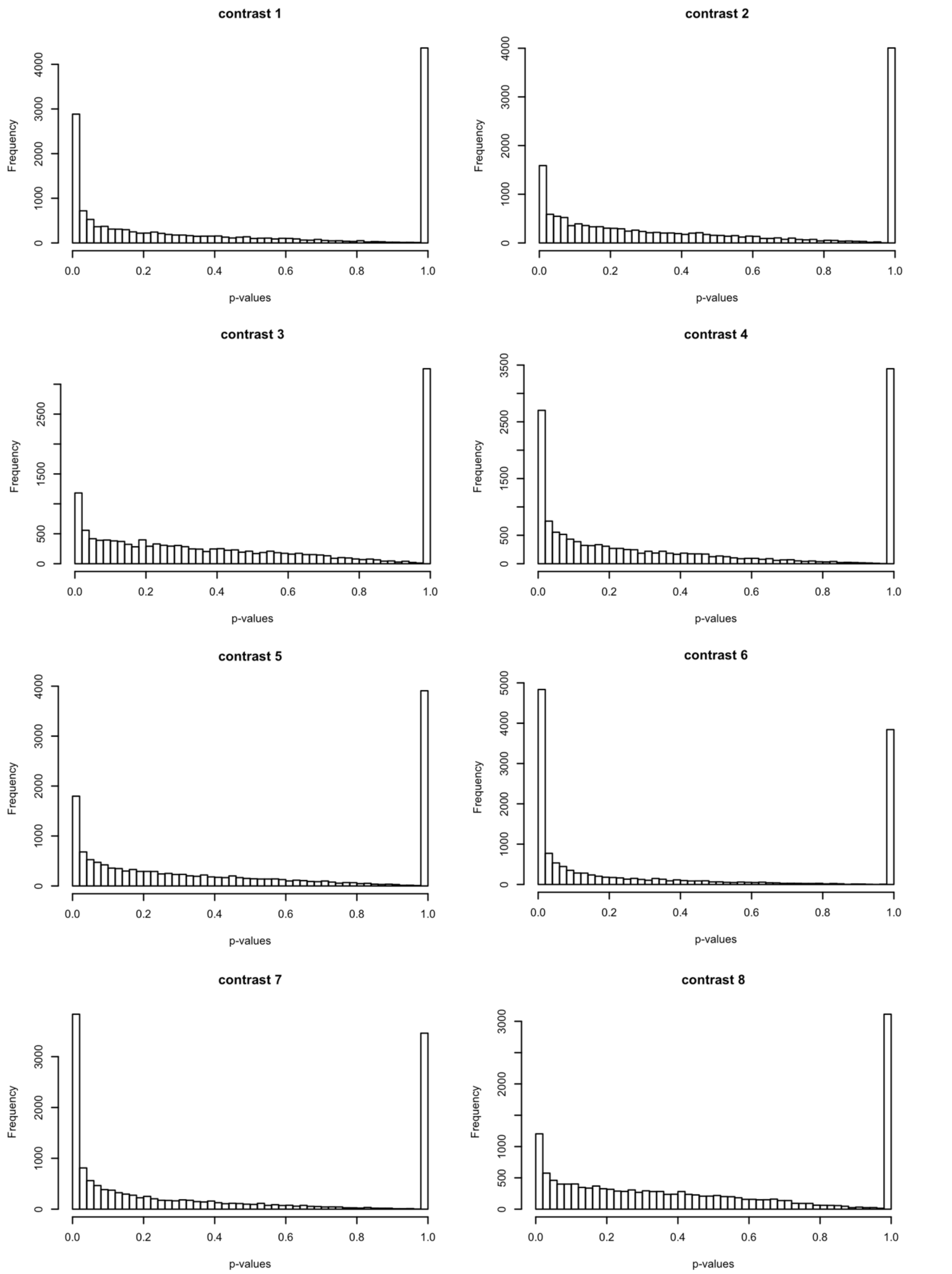


Figure S25: Histograms of the p-values from DoubleExpSeq. From these histograms, the huge number of DTU transcripts identified by limma diffsplice become apparent. In addition, we observe a gradual decrease of p-values over the interval [0.05 < p < 0.95], with a very large spike of p-values that are exactly 1 in all comparisons or contrasts of interest.

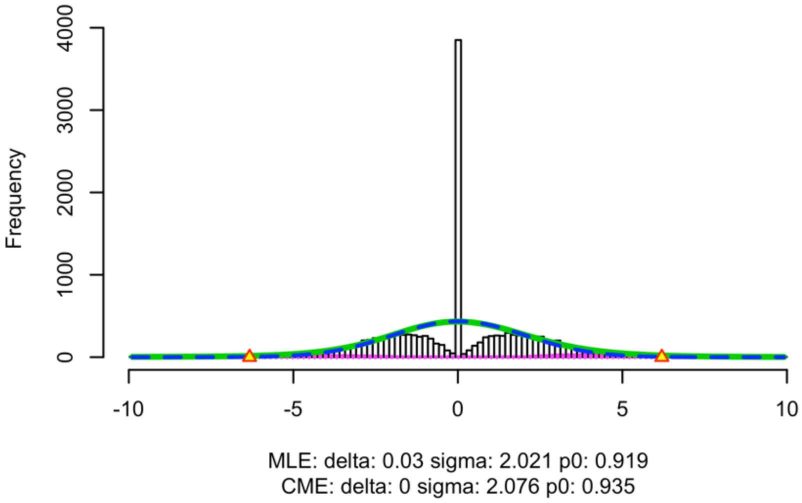
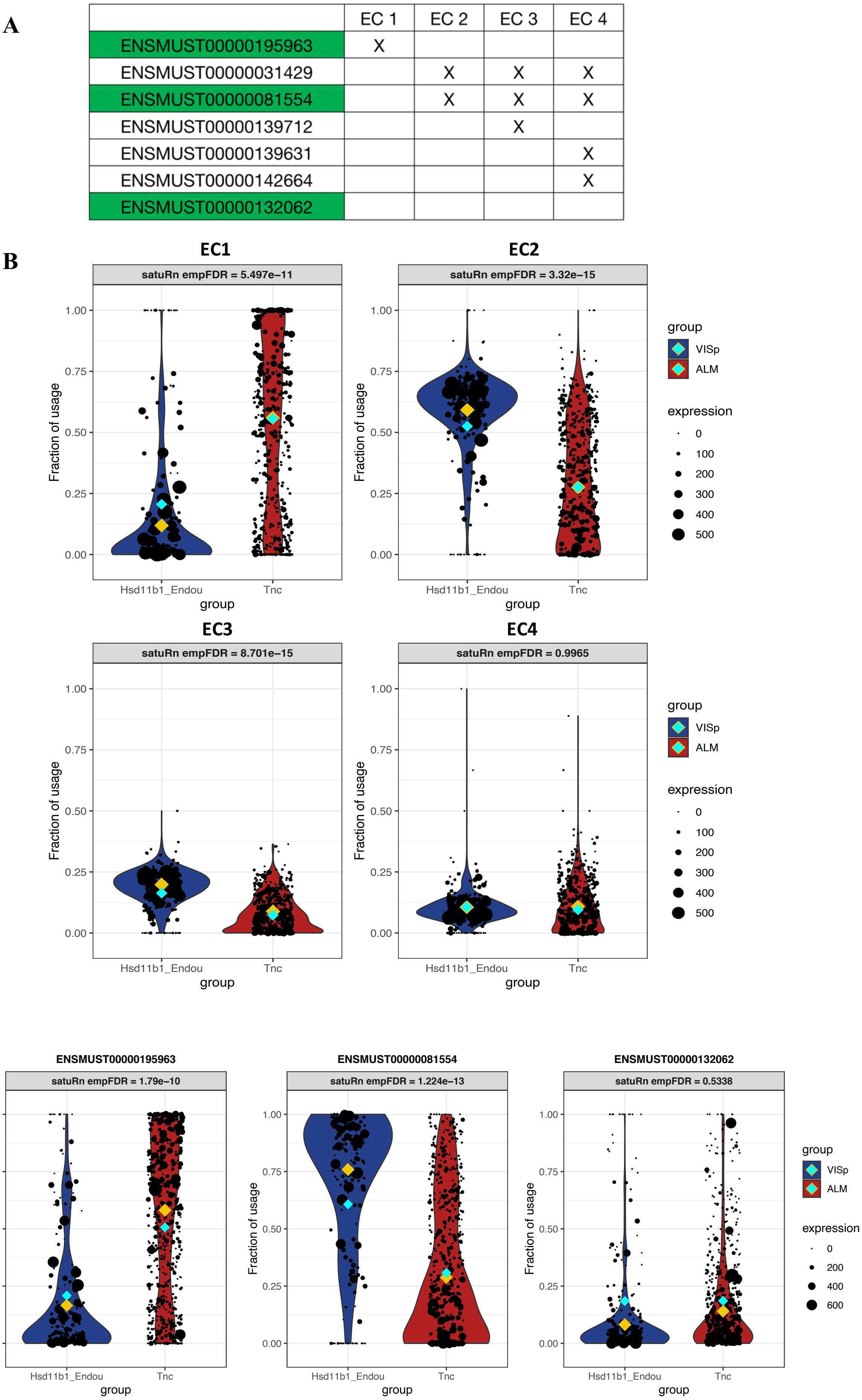


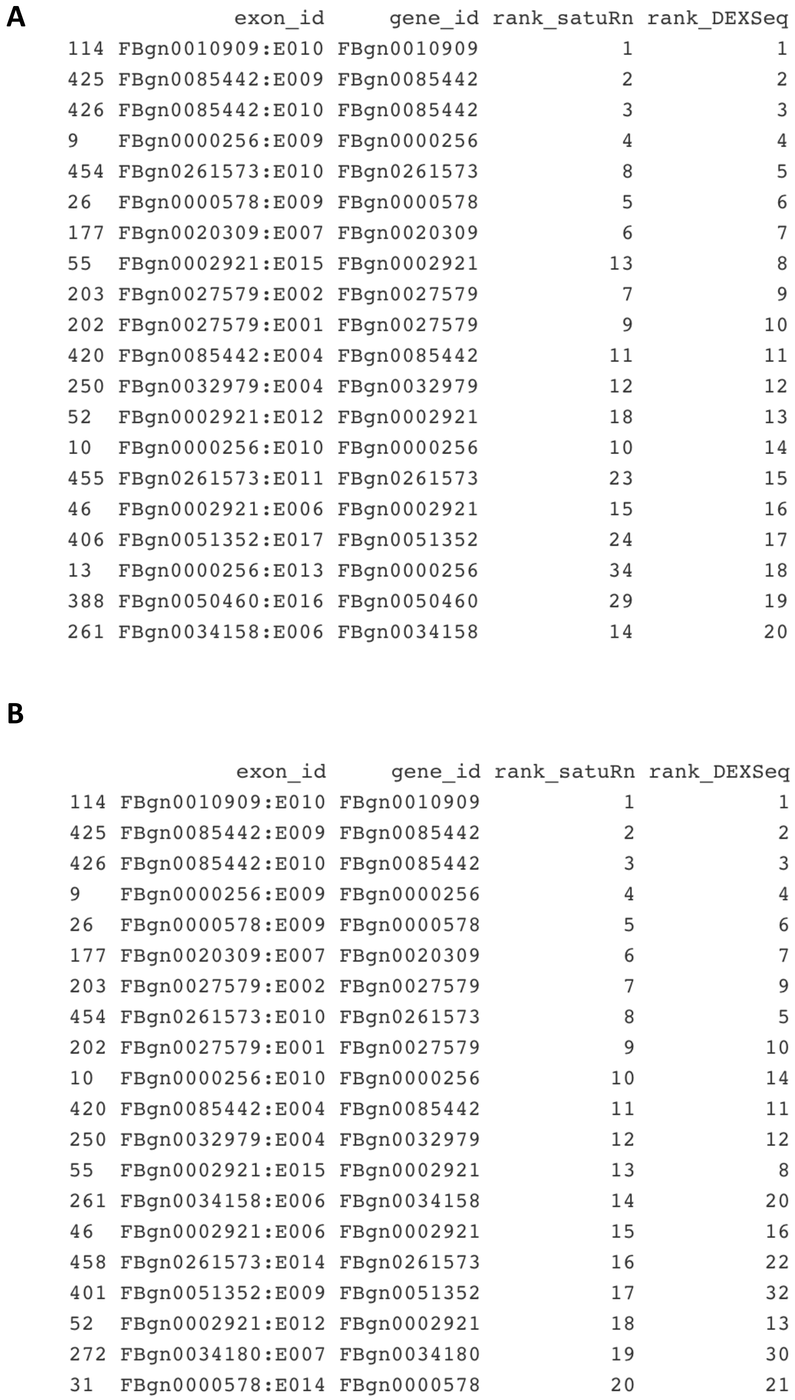
Figure S26: Empirical distribution of the test statistics in comparison #6 of the case study with DoubleExpSeq. The test statistics are z-scores, calculated from DoubleExpSeq p-values as described in formula 5 (see Methods). Theoretically, the bulk of these z-scores are expected to follow a standard normal distribution (mean = 0, standard deviation =1), i.e., assuming that most transcripts are not differentially used. However, the large spike of p-values equal to 1 (See Figure S14) results spike of z-scores equal to zero, which poses a problem when estimating the empirical null distribution (blue dashed curve). For more details on the *locfdr* figures we refer to the caption of figure S14.



A

	EC 1	EC 2	EC 3	EC 4
🟩 ENSMUST00000195963	X			
ENSMUST00000031429		X	X	X
🟩 ENSMUST00000081554		X	X	X
ENSMUST00000139712			X	
ENSMUST00000139631				X
ENSMUST00000142664				X
🟩 ENSMUST00000132062				

Figure S27: Differential usage analysis at the EC level and the transcript level for gene P2rx4. Panel A: Link between equivalence classes and transcripts. Four equivalence classes (ECs) of gene P2rx4 passed feature-level filtering. EC1 is compatible only with transcript ENSMUST00000195963. Equivalence classes two three and four are compatible with multiple transcripts. Transcripts that passed feature-level filtering in the transcript-level DTU analysis are colored green. Note that none of equivalence classes in the filtered data are compatible with the bottom transcript ENSMUST00000132062. **Panel B: Visualization of DU in the equivalence class analysis.** Evidence for differential usage is found in EC1, EC2 and EC3. **Panel C: Visualization of DTU in the transcript-level analysis.** Evidence for differential usage is found in transcript ENSMUST00000195963 and transcript ENSMUST00000081554. The DTU signal ENSMUST00000195963 corresponds directly with the DU signal in EC1, since EC1 is only compatible with ENSMUST00000195963 and vice versa (panel A). For EC2 and EC3, we cannot directly make a link with the transcript-level profiles. Because here we performed both types of analyses, we can infer that while EC2, EC3 and EC4 are compatible with multiple transcripts, the EM algorithm assigned the majority of reads to transcripts ENSMUST00000081544. If we had to rely only on the EC-level analysis, it would not be possible to unambiguously assign the differential EC usage to transcript ENSMUST00000081544, as all equivalence classes are also compatible with transcript ENSMUST00000031429.



A

```
               exon_id     gene_id rank_satuRn rank_DEXSeq
114 FBgn0010909:E010 FBgn0010909           1           1
425 FBgn0085442:E009 FBgn0085442           2           2
426 FBgn0085442:E010 FBgn0085442           3           3
9   FBgn0000256:E009 FBgn0000256           4           4
454 FBgn0261573:E010 FBgn0261573           8           5
26  FBgn0000578:E009 FBgn0000578           5           6
177 FBgn0020309:E007 FBgn0020309           6           7
55  FBgn0002921:E015 FBgn0002921          13           8
203 FBgn0027579:E002 FBgn0027579           7           9
202 FBgn0027579:E001 FBgn0027579           9          10
420 FBgn0085442:E004 FBgn0085442          11          11
250 FBgn0032979:E004 FBgn0032979          12          12
52  FBgn0002921:E012 FBgn0002921          18          13
10  FBgn0000256:E010 FBgn0000256          10          14
455 FBgn0261573:E011 FBgn0261573          23          15
46  FBgn0002921:E006 FBgn0002921          15          16
406 FBgn0051352:E017 FBgn0051352          24          17
13  FBgn0000256:E013 FBgn0000256          34          18
388 FBgn0050460:E016 FBgn0050460          29          19
261 FBgn0034158:E006 FBgn0034158          14          20
```

B

```
               exon_id     gene_id rank_satuRn rank_DEXSeq
114 FBgn0010909:E010 FBgn0010909           1           1
425 FBgn0085442:E009 FBgn0085442           2           2
426 FBgn0085442:E010 FBgn0085442           3           3
9   FBgn0000256:E009 FBgn0000256           4           4
26  FBgn0000578:E009 FBgn0000578           5           6
177 FBgn0020309:E007 FBgn0020309           6           7
203 FBgn0027579:E002 FBgn0027579           7           9
454 FBgn0261573:E010 FBgn0261573           8           5
202 FBgn0027579:E001 FBgn0027579           9          10
10  FBgn0000256:E010 FBgn0000256          10          14
420 FBgn0085442:E004 FBgn0085442          11          11
250 FBgn0032979:E004 FBgn0032979          12          12
55  FBgn0002921:E015 FBgn0002921          13           8
261 FBgn0034158:E006 FBgn0034158          14          20
46  FBgn0002921:E006 FBgn0002921          15          16
458 FBgn0261573:E014 FBgn0261573          16          22
401 FBgn0051352:E009 FBgn0051352          17          32
52  FBgn0002921:E012 FBgn0002921          18          13
272 FBgn0034180:E007 FBgn0034180          19          30
31  FBgn0000578:E014 FBgn0000578          20          21
```

Figure S28: Comparison of the exons ranked according to p-values between the DEXSeq and satuRn differential exon usage analysis. Panel A: Top 20 exons for DEXSeq and corresponding rankings for satuRn. **Panel B:** Top 20 exons for satuRn and corresponding rankings for DEXSeq. For both panels, we observe a very strong concordance between the rankings obtained with the DEXSeq analysis and the satuRn analysis.

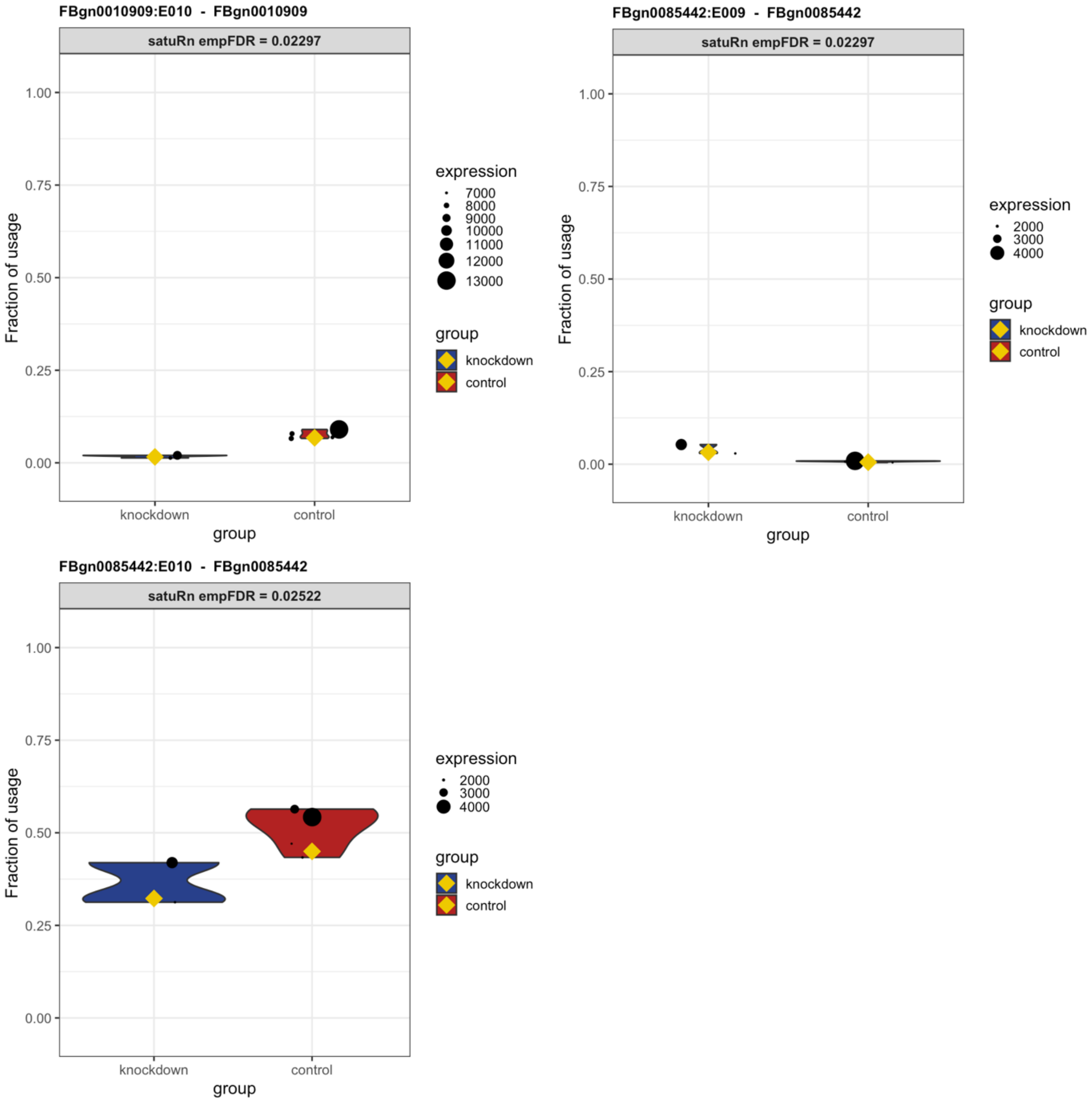


Figure S29: Visualization of differential exon usage with satuRn. satuRn visualization of the three exons with an FDR below 5% in the demonstrational differential exon analysis.

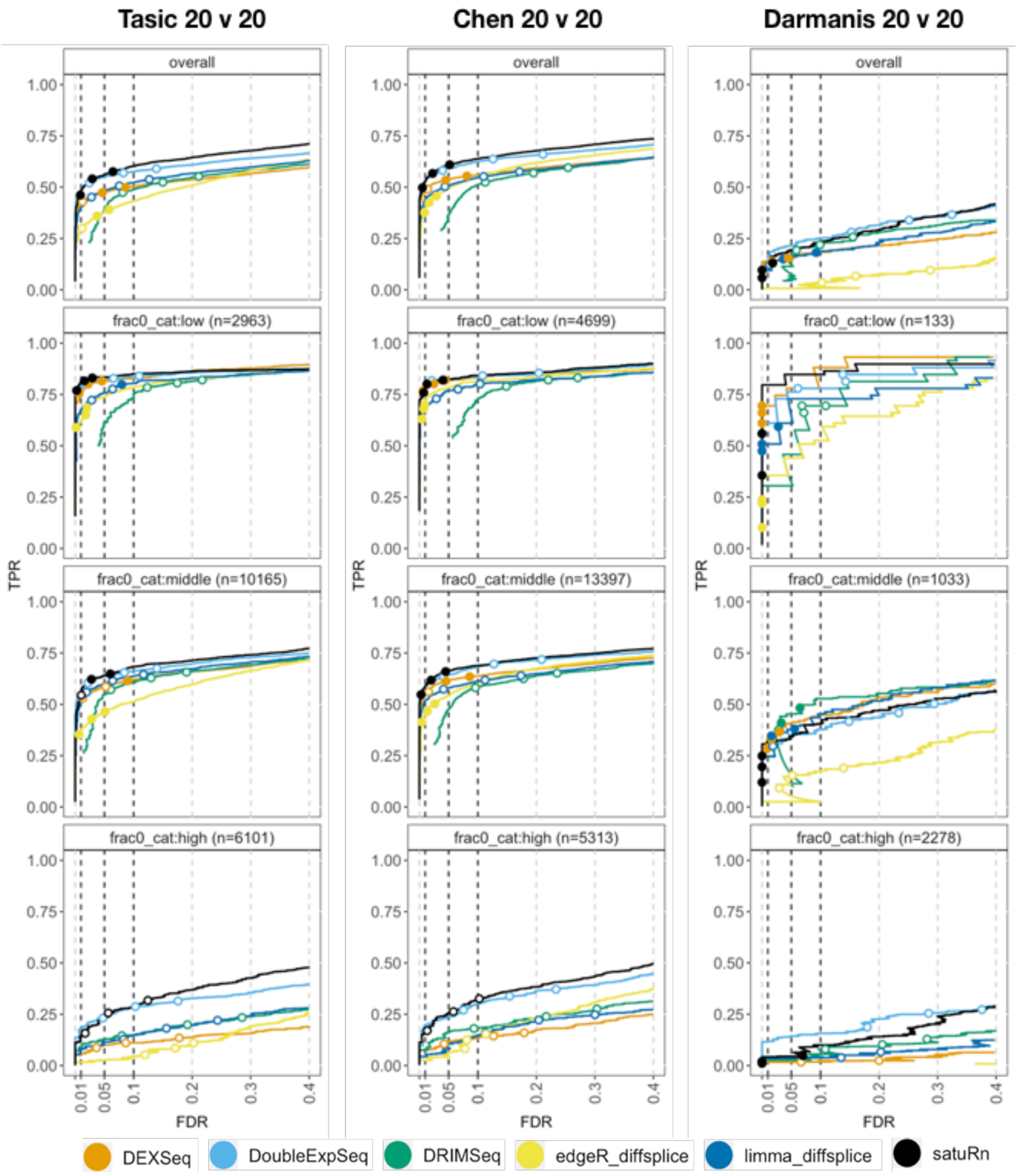


Figure S30: Performance evaluation on the smallest subset of the three scRNA-seq datasets, stratified by the percentage of zero counts. Performances are shown for datasets filtered with edgeR and using raw counts data. The top panels display the performances on the different datasets for all transcripts, as previously displayed in figures 4, S8 and S9. The other panels display the performances on different subsets of transcripts. The three strata correspond to transcripts of genes that have a low (< 25%), middle (25-50%) or high (> 50%) percentage of zero counts in their corresponding transcript-level count matrices. The number of transcripts in each stratum is indicated in the header of each panel. The performances are relatively similar between the different datasets within the same stratum. However, given that the number of transcripts in the stratum with the highest percentage zero counts is proportionally much higher in for the Darmanis dataset, the overall performances (top panel) on this dataset are markedly lower than for the other datasets.

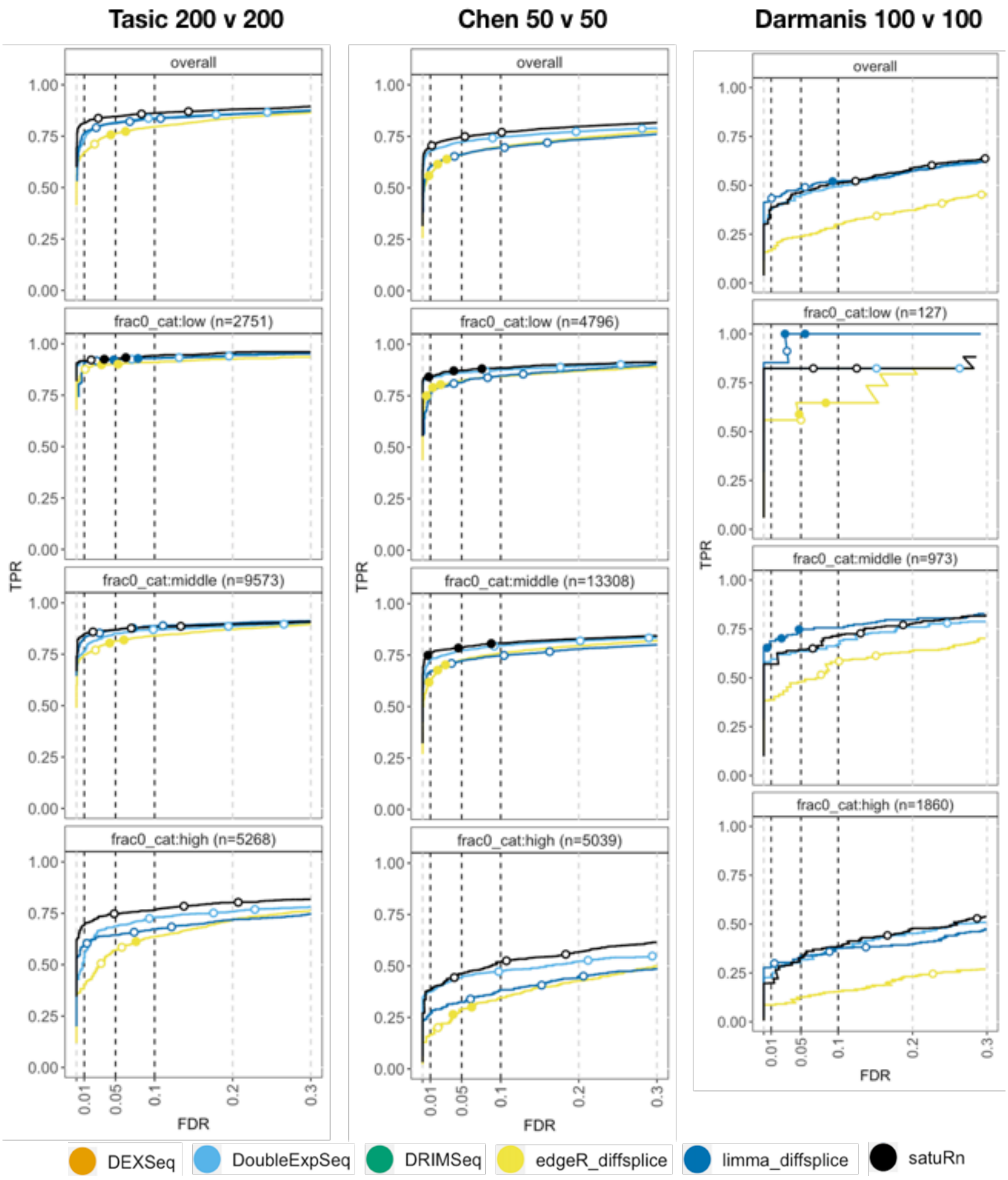


Figure S31: Performance evaluation on the largest subsets of the three scRNA-seq datasets stratified by the percentage of zero counts. Performances are shown for datasets filtered with edgeR and using raw counts data. The top panels display the performances on the different datasets for all transcripts, as previously displayed in figures 4, S8 and S9. The other panels display the performances on different subsets of transcripts. The three strata correspond to transcripts of genes that have a low (< 25%), middle (25-50%) or high (> 50%) percentage of zero counts in their corresponding transcript-level count matrices. The number of transcripts in each stratum is indicated in the header of each panel. The performances are relatively similar between the different datasets within the same stratum. However, given that the number of transcripts in the stratum with the highest percentage zero counts is proportionally much higher in for the Darmanis dataset, the overall performances (top panel) on this dataset are markedly lower than for the other datasets.

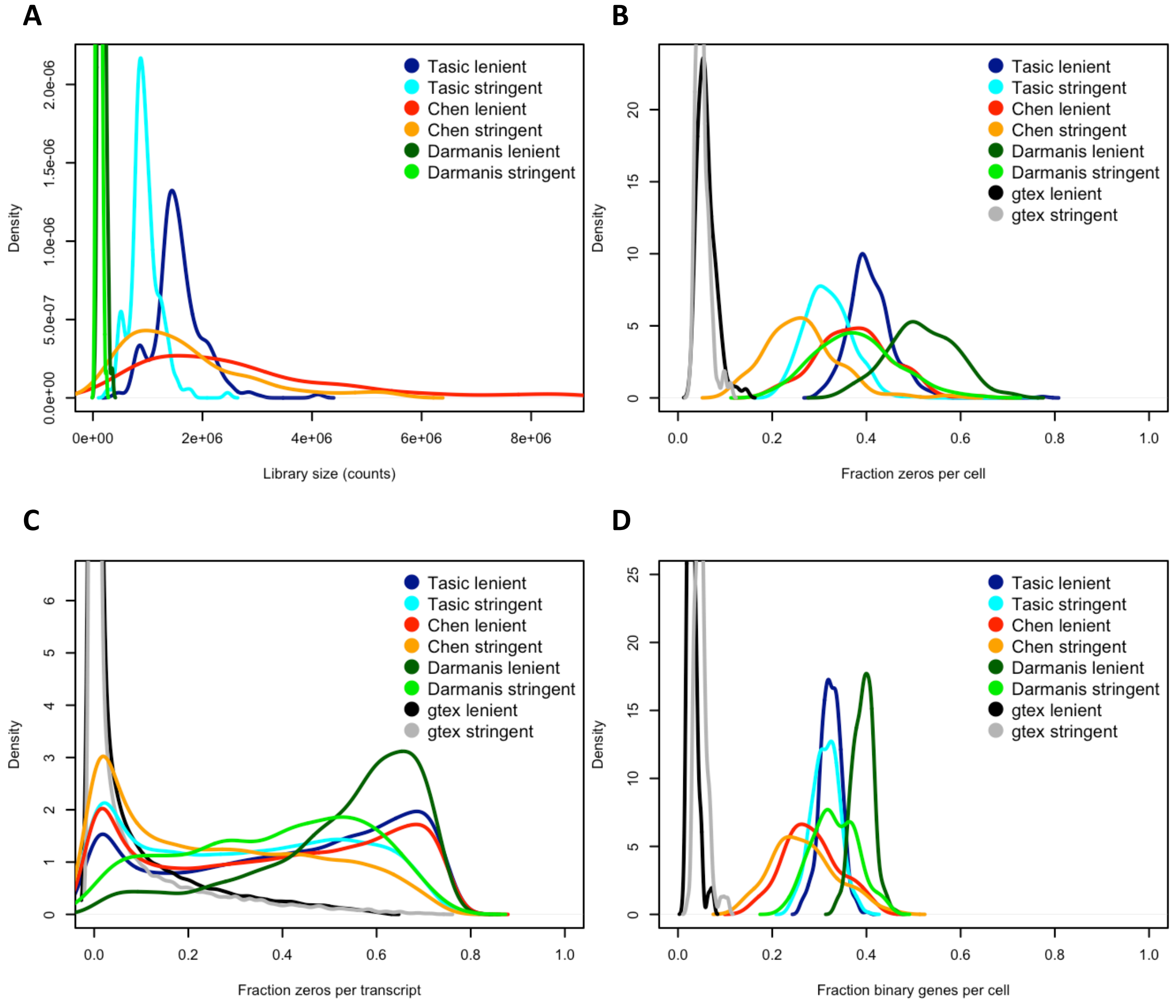


Figure S32: Properties of the three different scRNA-seq datasets. Datasets included are the largest subset of the Tasic dataset (400 cells), the Chen dataset (100 cells) and the Darmanis dataset (200 cells). The datasets were either filtered using edgeR (lenient) or DRIMSeq (stringent). **Panel A:** Density plot of the library sizes. The densities are obtained as the total sum of the counts per cell in each dataset. Library sizes are smallest for the Darmanis dataset. The mode of the densities for the Tasic dataset and the Chen dataset are similar, however, the spread is considerably larger for the Chen dataset. **Panel B:** Density plot of the fraction of zero counts per cell. The fraction of zero counts per cell is largest for the Darmanis dataset (modes of around 55% and 35%), followed by the Tasic dataset (modes of around 40% and 30%) and the Chen dataset (modes of around 35% and 25%). Adopting the more stringent transcript-level filtering criterium of DRIMSeq naturally reduces the percentage of zero counts. As a comparison, the fraction of zero counts on the bulk RNA-seq GTEx dataset (100 samples) was included as a reference (modes of around 5%). **Panel C:** Density plot of the fraction of zero counts per transcript. Similar to panel B, the percentage zero counts per transcript is highest for the Darmanis dataset, followed by the Tasic dataset, the Chen dataset and the GTEx dataset. **Panel D:** Fraction of binary genes per cell. A gene is called binary in a cell if only 1 isoform of that gene is expressed in that cell. Again, the highest fraction of fraction of binary genes is observed of cells from the Darmanis dataset, followed by the Tasic dataset, the Chen dataset and the GTEx dataset.

A

Tasic	20 v 20 lenient	200 v 200 lenient	20 v 20 stringent	200 v 200 stringent	raw
n_transcripts	19229	17591	9881	9074	99436
overall_zero (%)	41,66	41,01	32,44	32,01	83,34
binary (%)	32,1	32,41	31,19	31,33	24,86
all_zero (%)	11,17	11,46	9,11	9,24	51,7
Chen	**20 v 20 lenient**	**50 v 50 lenient**	**20 v 20 stringent**	**50 v 50 stringent**	**raw**
n_transcripts	23409	23143	11277	11209	99280
overall_zero (%)	38,29	37,58	26,46	26,07	78,26
binary (%)	29,21	28,76	27,65	27,2	25,15
all_zero (%)	8,94	8,75	5,86	5,83	42,82
Darmanis	**20 v 20 lenient**	**100 v 100 lenient**	**20 v 20 stringent**	**100 v 100 stringent**	**raw**
n_transcripts	3444	2961	844	769	175100
overall_zero (%)	53,41	51,85	39,2	37,61	95,36
binary (%)	39,62	39,34	33,88	32,79	15,69
all_zero (%)	27,99	26,97	17,91	16,87	77,55
GTEx	**5 v 5 lenient**	**50 v 50 lenient**	**5 v 5 stringent**	**50 v 50 stringent**	**raw**
n_transcripts	54019	55435	26630	26945	162972
overall_zero (%)	4,81	6,13	4,91	5,21	46,22
binary (%)	2,49	3,15	4,71	4,98	14,62
all_zero (%)	0,05	0,09	0,2	0,21	15,48

B

		Cell 1	Cell 2
Gene A	Transcript 1	0	0
Gene A	Transcript 2	5	0
Gene A	Transcript 3	0	0
Category		Binary	All_zero

Table S1: Summary statistics for the GTEx bulk dataset and the three scRNA-seq datasets. Panel A: Dataset identifiers are indicated in the top-left cell. The column headers specify the number of samples/cells of each subset and the adopted filtering strategy (lenient for edgeR, stringent for DRIMSeq). The column "raw" indicates the unfiltered count matrix including all cells and all samples, i.e., the raw output of the quantification procedures. The row "**N_transcripts**" indicates the number of transcripts retained in the dataset. "**Overall_zero**" is the percentage of zero values in the count matrix. "**Binary**" is computed on the gene level. For each gene, the fraction of cells that have a binary transcript usage pattern where only a single transcript of the gene is expressed (as indicated in panel B) is computed. Next, the mean of these fractions (over the genes) is taken. Such binary count profiles are less informative than profiles with counts for multiple transcripts within the same gene[3]. The transcript usage fractions will be zero and infinity, respectively, regardless of the count value of the expressed transcript. The computation of "**All_zero**" is similar to that of "**Binary**", however, here the fraction of cells that have only zero count values is computed for each gene and averaged over the genes, as indicated in panel B.

References

1. Soneson, C., Love, M. I. & Robinson, M. D. Differential analyses for RNA-seq: transcript-level estimates improve gene-level inferences. *F1000Research* **4**, 1521 (2016).
2. Efron, B., Turnbull, B. B. & Narasimhan, B. Locfdr: Computes Local False Discovery Rates. *R Packag.* **Version 1.**, http://CRAN.R-project.org/package=locfdr (2011).
3. Najar, C. F. B. A., Yosef, N. & Lareau, L. F. Coverage-dependent bias creates the appearance of binary splicing in single cells. *Elife* **9**, 1–23 (2020).