

GLMsim: a GLM-based single cell RNA-seq simulator incorporating batch and biological effects

Jianan Wang^{1*}, Lizhong Chen¹, Rachel Thijssen³,
Belinda Phipson¹, Terence P. Speed^{1,2*}

^{1*}Bioinformatics Division, Walter and Eliza Hall Institute of Medical Research, 1G Royal Parade, Melbourne, 3052, VIC, Australia.

²School of Mathematics and Statistics, University of Melbourne, Grattan Street, Melbourne, 3010, VIC, Australia.

³Blood Cells and Blood Cancer Division, Walter and Eliza Hall Institute of Medical Research, 1G Royal Parade, Melbourne, 3052, VIC, Australia.

*Corresponding author(s). E-mail(s): wang.j@wehi.edu.au;
terry@wehi.edu.au;

Abstract

With development of the single cell RNA-seq technologies, large numbers of cells can now be routinely sequenced by different platforms. This requires us to choose an efficient integration tool to merge those cells, and computational simulators to help benchmark and assess the performance of these tools. Although existing single cell RNA-seq simulators can simulate library size, biological and batch effects separately, they currently do not capture associations among these three factors. Here we present GLMsim, the first single cell RNA-seq simulator to simultaneously capture the library size, biology and unwanted variation and their associations via a generalized linear model, and to simulate data resembling the original experimental data in these respects. GLMsim is capable of quantitatively benchmarking different single cell integration methods, and assessing their abilities to retain biology and remove library size and batch effects.

Keywords: scRNA-seq, simulation, generalized linear model, library size, batch, biology, associations

28 Introduction

29 During the past few years, there has been significant progress in single cell sequenc-
 30 ing technologies, which allow scientists to explore gene expression at individual cell
 31 resolution. In contrast to bulk-RNA seq that studies average expression at the cell
 32 population level, single cell RNA-seq technologies are capable of exploring the tran-
 33 scriptomics of each individual cell. Advanced single cell protocols offer researchers
 34 new ways to understand biologically and medically relevant questions, such as the
 35 response of immune cells to anti-tumour drugs[1], the dynamic progress of embry-
 36 onic cell state evolution[2], cellular compositional changes across healthy and diseased
 37 tissues[3], detection of pathogenic pathways for neurodegenerative patients[4], and the
 38 potential regulatory gene changes of diabetes patients[5].

39 As single cell transcript sequencing became more popular, a rapid growth of tools
 40 occurred to answer different biological questions. The online single cell RNA sequenc-
 41 ing (scRNA-seq) tool database[6] has recorded more than 1440 software packages
 42 available for different analysis tasks. The choice of the tools determines the analysis
 43 results; therefore, using the most appropriate tool to carry out an analysis is essential
 44 for researchers. Benchmarking, which applies a number of tools to several datasets
 45 to decide the best-performing tool, is a direct solution to this multiple-choice issue.
 46 Despite remarkable progress having been made, benchmarking studies still face great
 47 challenges. One of the challenges is to find suitable datasets, because benchmarking
 48 results are highly dependent on the datasets that are used[7]. Inappropriate datasets
 49 can lead to biased selection of tools. Hence, researchers should make sure that each
 50 dataset is suitable for evaluating different methods. Another challenge is relevant to
 51 the characteristics of single cell data[8]. The complex structure of single cell data
 52 makes it hard to get complete information from experimental data directly. Unknown
 53 information left in the dataset can undermine the objectivity of the benchmarking
 54 results.

Simulation in this context refers to the creation of a computational model which represents and displays essential characteristics of real-world single cell RNA-seq data. Access to a faithful simulated dataset is of vital importance for the conduct of comparative studies and to help developers to check their methods[9–13]. Such assessments are difficult to carry out on the original single cell data if the experiments are not specifically designed. Simulation, however, can easily realize different scenarios and create extreme cases that can be used for testing. Simulation provides developers with an opportunity to investigate the influence of different parameters, examine the robustness of a method and validate the assumptions behind the method. In addition, it is hard to obtain real data with a wide range of conditions due to limited budgets. Simulation is much less costly than real biological experiments. Lastly, the quality of the original data is always doubtful, such as when the experiment damages droplets and lyses cells. Such low-quality real data will influence the accuracy of benchmarking results. However, simulation is able to address this issue via quality control at the beginning and produce high quality data.

Existing single cell simulation strategies fall into two major categories. One is based on the distributional models. Some popular methods, such as Splatter[14], SPARSim[15], SPsimSeq[16], scDesign[17], scDesign2[18], POWSC[19] and powsimR[20] use this strategy. These methods fit the data into some statistical distributions, obtain the parameter estimates, and then sample random values from the fitted distributions. Even though this strategy is quick and easy to follow step by step, it is not always known how well the estimated parameters fit the data. If the data fail to fit the distributions, simulation using the parameter estimates will lead to synthetic data which diverges from the original data. The second type of simulator, such as SymSim[21] and Minnow[22], models the key steps in RNA synthesis and in the sequencing process such as the enrichment of transcripts, the polymerase chain

81 reaction (PCR), and molecular fragmentation. This type of model succeeds in quan-
82 tifying technical errors from the beginning of the sequencing procedure, yet still finds
83 it difficult to accurately simulate all steps of gene expression.

84 One of the essential applications for the simulated data is to benchmark differ-
85 ent single cell data integration methods. Single cell RNA sequencing has been widely
86 applied during the past decade for its strength in exploring biology at single cell res-
87 olution. Plenty of integration tools[23–28] designed for single cell analysis purposes
88 have been developed, but none of above-mentioned tools use simulated data to exam-
89 ine their method. As a result, those methods require reliable simulated datasets to
90 evaluate their performances. That is because the unknown ground truth makes it hard
91 to use the original data to finish the assessment task and provide a clear explana-
92 tion of benchmarking results. In addition, benchmarking different integration methods
93 calls for consideration of the library size, batch and biological information and their
94 associations, because a good integration method is expected to keep the biological
95 differences and remove library size effects and the batch differences. Unfortunately,
96 existing single cell simulators do not satisfy all those conditions. Hence, synthetic data
97 with suitably designed information is necessary to help evaluate the performance of
98 different integration tools.

99 Although most existing single cell simulation methods[14–16, 18, 21] satisfac-
100 torily simulate library size, biology and batches separately, none of the methods
101 currently simulate the associations among the three factors. Here, we present GLMsim
102 (**G**eneralized **L**inear **M**odel based **s**imulator), a single cell simulator which aims to
103 tackle this issue. GLMsim fits each gene’s counts into a negative binomial generalized
104 linear model (GLM), estimating mean gene expression as a function of the estimated
105 library size, biology and batch parameters and then samples counts from negative
106 binomial distributions. If outlier values arise in the initial simulation, GLMsim has
107 procedures to set outliers back to a standard level. Overall, GLMsim outperforms

108 other methods in producing single cells counts which resembling those in the original
109 data, as a GLM is robust in capturing most essential characteristics of single cell data.

110 Results

111 An overview of GLMsim framework

112 Briefly, GLMsim includes three steps: (1) estimating parameters for each gene, (2)
113 simulating single cell gene counts and (3) rescuing outlier genes (Fig. 1). Initially,
114 GLMsim starts from an observed scRNA-seq count matrix that includes the cell type
115 and batch information. GLMsim captures the main characteristics of the data by
116 fitting a generalized linear model, returning estimated parameter values for each gene.
117 Finally, a synthetic count matrix with same number of genes and cells is generated
118 using the estimated coefficients from the previous step. In order to retain most of the
119 essential properties from the original data, GLMsim simulates gene by gene and keeps
120 the same library size as that data. Since it is possible to get outlier values from the
121 simulation, an additional step in GLMsim checks and corrects for outliers if they exist
122 after the initial simulation.

123 GLMsim captures associations between library size, biological 124 effects and batch effects

125 In order to evaluate the performance of single cell integration methods, it is important
126 to simulate scRNA-seq data that captures library size, biology and batches. Library
127 size refers to the sum across genes of all counts within a cell. Biology commonly
128 stands for cell types, subtypes or conditions, such as control or stimulated states.
129 Batch factors here are technical variations that cause differences in gene expression
130 across cells, such as different storage conditions, lab operations, protocols or sequencing
131 platforms. Current simulation methods capture biology, some capture batches, while
132 others are good at simulating library size. None of the methods seem to be able to

133 simulate the associations between these three factors (Table 1). Association here means
 134 that one factor has potentially different effect across different categories of another
 135 factor. For example, cells from different biological groups may have different library
 136 size distributions, which is an association between library size and biology. It is also
 137 likely that each batch includes heterogenous cell types. Similarly, one cell type may
 138 belong to multiple but not all batches. These two scenarios are regarded as involving
 139 an association between batch and biology. If the cells with two out of three factors
 140 satisfy any of the above conditions, we consider that these associations potentially
 141 exist in the dataset.

142 The Celline2 dataset is a typical one showing association between batch and biol-
 143 ogy. In the original dataset, batch 3 includes cells from both the Jurkat and 293T cell
 144 lines (Fig. 2a). GLMsim enables one to retain the state of mixture of batch 2 and
 145 batch 3 cells within the 293T group (Fig. 2b), while Splatter divides the cells into 2
 146 groups and 3 batches without showing the partial mixture between batch 2 and batch
 147 3 cells (Fig. 2c). scDesign2 can simulate the two biological groups, but the isolated
 148 pattern between Jurkat cells is not shown (Fig. 2d). SymSim embedded three batch
 149 groups into each biological group, and the three batch groups do not have any overlap
 150 with each other (Fig. 2e). SPARSim did not represent the batch effect as seen in the
 151 original data, though it did demonstrate an association between batch and biology in
 152 batch 3 group (Fig. 2f).

153 In order to explore associations between all three factors, we examine the CLL
 154 dataset. It involves 6 biological groups and batches originating from 7 CEL-Seq2 plates.
 155 The key point in this dataset is that associations exist between all pairs of the three
 156 factors (Fig. 3a,b). None of the popular single cell simulators can deal with this com-
 157 plex dataset. GLMsim simulates data that most resembles the original data (Fig. 3c).
 158 For the Granta biological group, GLMsim simulates cells from LC89, LC91, LC93,
 159 LC95, LC96, and LC99. Further, a cluster of cells are surrounded by VEN and other

160 treatments, and GLMsim shows a mixture of the groups of cells. Splatter completely
 161 failed to simulate this dataset accurately (Fig. 3d), because cells from all batches and
 162 all groups were mixed together. Due to the absence of batch effects in their model,
 163 scDesign2 can only capture the biology. Gaussian copulas permit scDesign2 to capture
 164 part of the biology well, such as in the DMSO and Granta groups, but it is not able to
 165 distinguish certain subgroup differences, such as the VEN and VEN+BCLxLi groups
 166 that were separated from VEN+MCL1i and VEN+NAV in the original data (Fig. 3e).
 167 The performance of SymSim is strange. It splits batches into equal size groups 5,6
 168 and 7, and evenly divided groups 2, 3 and 4 into all batches (Fig. 3f). In addition, the
 169 majority of cells have high library sizes which suggests that SymSim failed to simulate
 170 the library size effect. The performance of SPARSim is next best after GLMsim (Fig.
 171 3g). SPARSim is able to partition cells into the different biological and batch groups,
 172 but it mixed cells in the wrong way. For example, the Granta cells are separated from
 173 DMSO cells in the original data, but were mixed together by SPARSim.

174 The success of GLMsim in simulating associations between library size, biological
 175 effects and batch effects using GLMsim lies in its ability to accurately capture biolog-
 176 ical and unwanted variation simultaneously from original data. The failure of other
 177 methods to do so is mainly caused by their estimating library size, biological and
 178 batch parameters separately. For instance, Splatter simulates these factors in three
 179 independent steps. The implicit assumption of Splatter is that the three factors are
 180 independent, and hence it cannot capture associations among the three factors. Fur-
 181 ther, sampling DE genes for biological and batch factors together will lead to overlap
 182 of DE genes among the biological or batch groups and disordered gene rankings for the
 183 two factors, and this leads to its inability to simulate complex datasets with multiple
 184 biological groups and batches. Gaussian copulas are a powerful strategy to capture
 185 gene-gene correlations, and so scDesign2 is able to simulate heterogenous cell clusters.
 186 However, the absence of batch terms in the model implies that scDesign2 is unable to

187 simulate a dataset with technical variation. SymSim cannot mimic any given data set
188 for two reasons: (1) It estimates parameters from its own database instead of the given
189 dataset, which limits its simulation ability; (2) It simulates biological and batch effects
190 in sequential order, which prevents it considering the effects jointly. The performance
191 of SPARSim is better, because it simulates library size, mean gene expression and bio-
192 logical variability independently for each cell type. While the SPARSim simulation is
193 competitive, it still includes different groups of cells not present in the original data.
194 That is probably caused by its assumption of a common distribution for batch effects,
195 even if different batches display different features.

196 **GLMsim keeps most of the essential features of real data**

197 We now compare the simulated data from GLMsim and 4 other single cell RNA-
198 seq simulators on all 6 reference datasets. Utilizing gene-level and cell-level metrics
199 (see "Methods"), we systematically evaluate the performance of all 5 simulators. The
200 gene-level metrics consist of gene means, total gene UMIs, gene variances and gene
201 proportions of zeros, and the cell-wise metrics include library size and cell proportions
202 of zeros.

203 The gene-level comparisons are all scatter plots between features of the simulated
204 and the reference data. We see that GLMsim outperforms the other methods (Fig.
205 4a-d). The similarity between GLMsim and the reference data indicates that GLMsim
206 maintains the gene-wise features. scDesign2 and SPARSim ranked in the middle in
207 these comparisons, with small deviations from the reference datasets for several genes
208 (Fig. 4b,c), suggesting that the gene-gene correlations of scDesign2 and individual
209 gamma distributions of SPARSim can effectively recapitulate gene-level properties.
210 Splatter and SymSim consistently have poor performances for all comparisons and
211 all datasets (Fig. 4a-d). This is due to the fact that Splatter simulates mean gene
212 expression across all genes via a common gamma distribution, while SymSim simulates

counts by mimicking the mRNA capturing procedure. In addition, we also investigate how well the methods preserve the relationship between gene means and gene variances as well as gene proportions of zeros (Fig. S1). The non-UMI hESC data has a wider range of mean-variance and mean-zero proportions compared to the UMI datasets.

For the cell-level comparisons (Fig. 5a,b), GLMsim ranked best, and SPARSim ranked second in simulation of library sizes and cell proportions of zero. SPARSim performed well in simulation of simple data, such as Celline1, Tung, and HCC1395 datasets, but it failed to simulate more complex datasets, which indicates that sampling DE factors from a common distribution limits its ability to simulate complicated situations, despite the fact that the multivariate hypergeometric distribution can handle simple cases. On the contrary, GLMsim incorporates library sizes from reference data directly in the model and recovers cell-level properties. Moreover, GLMsim is the only method successfully simulating the cell proportions of zero on the non-UMI hESC dataset, whereas all other methods show a large difference from this reference data. The remaining methods show weakness in keeping the cell-level attributes. For Splatter, the log normal distribution does not preserve the original library size information, and the logistic regression has drawbacks shaping the dropout events. For scDesign2 and SymSim, their failure is not unexpected since they do not consider library size in their model.

Overall, GLMsim is better than the other methods in simulating data similar to the reference data in the features we present for both UMI and non-UMI dataset (Fig. 6). SPARSim and scDesign2 perform well on gene-level metrics but fail to capture certain cell-level characteristics, such as library sizes and cell proportions of zero. Splatter and SymSim consistently have lower Spearman correlation with the real data, indicating that the two methods are unable to reproduce data similar to the reference data. The poor performance of these two methods suggests that sampling single cell characteristics from statistical distributions will distort the shape of the simulated

240 datasets as a whole. Additionally, the failure of SymSim lies in estimating parameters
241 from its internal database with the dataset that most similar to the real data, instead
242 of obtaining the parameters from the reference data directly.

243 GLMsim has high computational efficiency

244 The computational scalability varied across different datasets (Fig. 7). Most of the
245 methods finish the simulation tasks using less than 3 hours CPU time and 10 gigabytes
246 of memory. However, scDesign2 takes much longer than other methods, and SymSim
247 requires much more memory than the others. This puts an emphasis on the balance
248 between the accuracy of the model and the computational efficiency. scDesign2, for
249 example, explicitly captures the gene-gene correlation, but at the cost of runtime,
250 spending more than 10 hours to simulate 1,344 cells. In contrast, Splatter and SPAR-
251 Sim take a much shorter time to simulate, and their runtime does not differ with the
252 size of the reference data, which demonstrates that sampling each feature from statis-
253 tical distribution is quicker, but sacrifices simulation accuracy. In general, GLMsim is
254 in the middle tier among the simulators in terms of computational time, and its run-
255 time is stable, the curve not changing much with the complexity of the dataset (Fig.
256 5.7a). Even the most complex CLL dataset, with multiple cell types, batches and their
257 associations, does not require more time for GLMsim. In addition, GLMsim has the
258 lowest memory usage, especially for the non-UMI dataset.

259 GLMsim is robust to outliers

260 Some datasets will give extreme simulation values if outliers are not dealt with prop-
261 erly. For example, there is one extreme outlier gene after an initial simulation by
262 GLMsim (Fig. 8a), which is caused by poorly estimated parameters from the Cellline1
263 data. The estimated parameters intercept $\hat{\beta}_0$, biological coefficient $\hat{\beta}_1$, library size coef-
264 ficient $\hat{\alpha}$ and the dispersion $\hat{\phi}$ are -9.43, 28.88, 0.66 and 2.18×10^{-4} respectively. All

265 estimated parameters are within a reasonable range except for $\hat{\phi}$ (Fig. 8e), and that
 266 leads to an abnormally large negative binomial variance (Equation 3). Consequently,
 267 the simulated negative binomial counts can be unreasonably large integers that shift
 268 the mean gene expression out of a realistic range.

269 GLMsim provides three strategies (see "Methods") to address outlier issues. The
 270 first method is based on a robust negative binomial GLM, which utilizes the esti-
 271 mated coefficients from robust negative binomial as starting values to refit negative
 272 binomial GLM. After applying this strategy, the gene mean expression was assigned
 273 to an acceptable level (Fig. 8b). The second approach is winsorizing the fitted coeffi-
 274 cients. We found that the β_1 coefficient and the dispersion parameter ϕ are beyond the
 275 thresholds (Fig. 8e). Then we use the thresholds directly as the new estimated coeffi-
 276 cients, which are $Q(0.9)(2.68)$ and $Q(0.1)(0.27)$ for the distributions of β_1 and $\log \phi$
 277 across genes, respectively. Finally, the clipped coefficients give counts with a sensible
 278 gene mean (Fig. 8c). The third strategy is to construct a relationship between gene
 279 means and estimated coefficients in the reference data by fitting a loess line. Now,
 280 each predicted coefficient is from the loess line corresponding to the mean expression
 281 of the outlier gene (Fig. 8f). Eventually, the outlier gene mean value is optimized when
 282 correcting the outlier values by each loess trend across genes (Fig. 8d). In addition to
 283 the gene mean, we also examine the library size before and after rescuing the outlier
 284 genes (Fig. S2). The simulated library sizes are closer to those from reference data
 285 after refining the outliers.

286 The outlier problem also exists in two other datasets: hESC and CLL. We applied
 287 all three methods to those datasets and found that the robust negative binomial
 288 GLM and trended coefficient strategies performed stably compared to the winsorizing
 289 strategy (Fig. S3a,b). The total computational time among the three methods showed
 290 no differences (Fig. S4). Considering the robustness and stability of the three methods,
 291 we chose the trended coefficient as the default outlier handling strategy.

292 Assessing single cell integration methods using GLMsim

293 In the original Celine2 dataset, the cells are separated into the Jurkat and 293T
294 cell types, and each cell type is included in two different batches, one alone and
295 another together, which creates an association between biology and batch. We used our
296 simulated Celine2 dataset to evaluate the performance of different scRNA-seq integra-
297 tion methods. A good integration method should remove all the unwanted variation
298 across the batches but keep all the biology. We use different metrics (Supplementary
299 Methods) to assess the extent to which the integration methods achieve these goals.

300 In the GLMsim simulation of the dataset, the cells are separated into two different
301 biological groups and three different batches along the first principal component (Fig.
302 9). Using plots of PC2 versus PC1 of the integrated simulated data, RUV-III-NB[26]
303 and scMerge[27] are seen to successfully remove the batch differences and keep the
304 biology. Other methods, such as scran[29], mnnCorrect[24], fastMNN[25], Seurat[30]
305 Pearson residuals and Seurat log corrected data, exhibit no differences before and
306 after data integration, as the batch differences remain. ZINB-WaVE[31] and Seurat
307 Integrated data overcorrect in removing the batch effects, as biology has been removed.
308 In summary, RUV-III-NB and scMerge maintain the biology even when it is associated
309 with batches.

310 The biological silhouette score, a score ranging from -1 to 1 indicating whether bio-
311 logical groups are clearly distinguished from each other or not, was used to evaluate
312 the ability to enhance biological patterns (Fig. S5). We identified that the RUV-III-
313 NB and scMerge integrated data show their abilities to effectively detect the biological
314 signals, while ZINB-WaVE, fastMNN, mnnCorrect, Seurat Pearson residual and Seu-
315 rat integrated data fail to pick up the biological signals after data integration. The
316 Seurat log corrected data and scran also performed well to keep the different biological
317 groups. The relative log expression (RLE) plot[32] was then applied to evaluate the
318 performance of removing library sizes. RUV-III-NB and scMerge are the top ranked

319 methods for the RLE metrics (Fig. S6). Except for those two methods, scan also per-
 320 formed well to remove the library size effect although this method is straightforward
 321 to reduce the effect by scaling the library size directly. Other methods, such as Seurat,
 322 ZINB-WaVE, fastMNN and mnnCorrect still have high correlation with library size
 323 after batch removal, which indicates that those methods have limitations in mitigat-
 324 ing the library size effect. Another metric to determine the performance of removing
 325 library size effects is the Pearson correlation between library size and gene UMI counts.
 326 scMerge is the best method to remove the library size effects, because its range of cor-
 327 relation is narrower than other methods (Fig. S7). Other methods performed similarly
 328 with correlations close to 0 for almost all genes. fastMNN only provides a data format
 329 in low dimensions for visualization purpose. As a result, it cannot be used for down-
 330 stream analysis and performed badly with a wider range of correlation to library size.
 331 In regard to the batch effects, the proportion of DE genes across batches has been
 332 applied to benchmark different integration methods. Theoretically, the proportion of
 333 DE genes for the same cell type should be low after removing the batch effects. RUV-
 334 III-NB is the unique method that shows low proportion of DE genes, suggesting that
 335 RUV-III-NB log PAC data is an ideal choice to carry out downstream DE analysis
 336 (Fig. S8). All other methods exhibit a high proportion of DE genes after data inte-
 337 gration, indicating that those methods are unable to provide appropriate integrated
 338 data format used for downstream analysis.

339 Overall, RUV-III-NB outperforms other methods in gaining high scores across all
 340 metrics (Fig. S9), which implies that RUV-III-NB successfully removed the library
 341 size effect, the batch effect and retained the biological effect in the simulated data. In
 342 contrast, other methods obtained a low score for at least one metric. For example, the
 343 Seurat integrated data is another example that performed badly in almost all metrics
 344 except for the technical silhouette score. It illustrates that the Seurat integrated data

345 has the advantage of removing library size in the principal components, but no more
346 benefits are shown by this method.

347 **GLMsim exhibits simulation stability**

348 It is possible that the random numbers generated by a single cell simulator can influ-
349 ence the simulation results and will further influence the downstream analysis results.
350 In order to check the random effect by the simulator, we simulated 5 Cellline2 datasets
351 by setting different random seeds. Then we compared the benchmarking results from
352 the original and all 5 simulated datasets (Fig. S10-S15). We found that the original
353 and all simulated data showed similar performances for all benchmarking metrics. This
354 indicates that GLMsim simulated data is stable and will not be influenced by random
355 aspects of the simulations. Since GLMsim can capture most of the basic features of
356 the original data, the benchmarking results from the simulated data are similar to
357 those from the original data.

358 **Discussion**

359 In this paper, we have proposed GLMsim, a practical method to simulate the library
360 size, biological and batch effects present in scRNA-seq data. Currently, none of the
361 existing scRNA-seq simulation methods are able to capture associations between these
362 three factors, despite numerous experimental datasets exhibiting such associations.
363 GLMsim achieves this goal by incorporating library size, batch and biology in the
364 model. In this way, GLMsim not only recovers the information relevant to these
365 three factors from experimental data, it also efficiently handles challenging large-scale
366 datasets with multiple batches and biological groups. Since most single cell simula-
367 tors simulate the three factors separately in their models, the simulation patterns for
368 those methods will be poor representations of the actual data, especially with complex
369 scenarios. In particular, if the method simulates different single cell groups through

370 multiplying by DE factors, the assignment of DE genes to highly diverse batch and
371 biological clusters will be problematic, because those methods cannot avoid the DE
372 assignments across different clusters.

373 We have compared GLMsim to other single cell simulators by a series of gene-
374 level and cell-level summaries to evaluate the performance of GLMsim in terms of
375 its ability to capture the characteristics of experimental data and its fidelity to that
376 data. Utilizing 6 datasets with different numbers of genes and cells, sequenced by
377 different platforms, we found that GLMsim ranked best in simulating data similar
378 to experimental data. In particular, GLMsim is the only method that enables us to
379 precisely reproduce the cell level proportions of zeros in non-UMI data. scDesign2 and
380 SPARSim performed well in the gene-level metrics, but poorly in simulating cell-level
381 features. Splatter and SymSim have a poor performance in every respect. In summary,
382 GLMsim is the most accurate single cell simulator of basic single cell properties, with
383 the notable exception of gene-gene associations. Its accuracy lies in the GLM being
384 able to estimate parameters for each gene.

385 Single cell data typically includes 10-20 thousand of genes and at least hundreds, if
386 not thousands, of cells. As a result, it can be time consuming fitting every gene using a
387 GLM and so. We parallelized the computations when estimating the parameters, and
388 we also provide a sequential option for users to obtain the fitted coefficients in case
389 of runtime errors. Although GLMsim is slightly slower than the distribution-based
390 methods such as Splatter and SPARSim, GLMsim successfully balances the accuracy
391 of simulation and the computational time. Moreover, GLMsim has lower memory usage
392 than the other methods, especially for non-UMI data.

393 Current single cell simulators ignore or filter out outlier simulated values. GLM-
394 sim addresses this problem with three different approaches instead. It offers a robust
395 negative binomial GLM, the winsorizing of estimate and the trended approaches. All
396 these methods address the outlier problem well, but the performance of the trended

method is more stable than the other two methods. The runtime of the three strategies does not differ significantly, and thus, we choose the trended coefficient approach as the default option.

GLMsim simulated data has been applied to give comprehensive benchmarking across popular single cell integration methods. We have shown that RUV-III-NB outperformed other methods in most of the metrics, while scMerge was in second place, as it is slightly weaker in removing batch effects on the metric relevant to the proportion of DE markers across batches. Some of the methods do not show any differences before and after integration, such as scran, mnnCorrect, fastMNN, and Seurat, suggesting that those integration methods lack the ability to deal with library size, batch, and associations between batch and biology. On the other hand, other methods demonstrate overcorrection problems, like ZINB-WaVE and Seurat integrated data. We also found that some integration methods can only provide data in reduced dimensions for visualization purpose, including mnnCorrect and fastMNN. However, those integrated data in low dimensions cannot support for certain downstream analyses such as detecting DE markers. In addition, we have demonstrated that GLMsim performs stably on benchmarking, since same conclusions were given across simulations when offering different random seeds to simulate the same real dataset. Benchmarking results based on the simulated data offer researchers an objective standard with which to select an appropriate approach for single cell analysis.

GLMsim simulation is robust, reproducible, user-friendly, and the framework is distinctive compared to any other method. Users only need four steps to finish simulation. First, provide basic information from experimental data, which includes the count matrix and the biological and batch labels for each cell. Second, it estimates parameters for each gene. This step is time-consuming, so we encourage users to carry it out on a high performance computer system if possible. Next, simulate an initial

count matrix, which requires the parameter estimates from the previous step to calculate the estimated mean for each entry of the matrix. Finally, check if outlier genes exist in the initial simulated data, and if so choose an appropriate method to correct for these outlier values. At present, GLMsim only works for simulation of scRNA-seq data.

Methods

Estimating coefficients for each gene

Let $Y = (y_{ij}^0)$ represent the count matrix from the original dataset, whose G rows correspond to genes and N columns correspond to cells, respectively. Write

$$y_{ij}^0 \sim \text{NB}(\mu_{ij}^0, \phi_i^0) \quad (1)$$

$$\mathbf{E}[y_{ij}^0] = \mu_{ij}^0 \quad (2)$$

$$\mathbf{Var}[y_{ij}^0] = \mu_{ij}^0 + \frac{(\mu_{ij}^0)^2}{\phi_i^0} \quad (3)$$

to denote that for gene i in cell j , y_{ij}^0 is distributed according to a negative binomial (NB) distribution[33] with mean parameter μ_{ij}^0 and dispersion parameter ϕ_i^0 .

Assume that in the original data, there are M biological groups and K batches.

Our GLM then takes the form

$$\log(\mu_{ij}^0) = \beta_{i0} + \beta_i X_j + \alpha_i W_j \quad (4)$$

where β_{i0} is the baseline expression of gene i for a reference biological group and reference batch group, β_i is a vector of parameters for the biological influences and α_i

is a vector of parameters relevant to the unwanted variations. $X_j = (0, x_{j2}, \dots, x_{jM})^T$ is a vector of parameters related to the biological groups that if cell j belongs to the m -th group other than the reference group, $x_{jm} = 1$ and other entries are 0. $W_j = (L_j, w_{j1}, \dots, w_{jK})^T$ is a vector of unwanted variation including library size and batches. L_j corresponds to log library size for cell j :

$$L_j = \log \left(\sum_{i=1}^G y_{ij}^0 \right) \quad (5)$$

w_{jk} corresponds to batch for cell in non-reference groups that if cell j is in the k -th batch, $w_{jk} = 1$ and other entries are 0. We use the `glm.nb`[33] function from the package `MASS` to get the estimated parameters: $\hat{\beta}_{0i}$, $\hat{\beta}_i$, $\hat{\alpha}_i$ and $\hat{\phi}_i^0$ for each gene from the dataset being fitted.

Simulating single cell gene counts

In this step, the counts for each gene are simulated independently using the estimated coefficients from the previous step. The estimated mean expression parameter $\hat{\mu}_{ij}$ of the simulated count y_{ij} for gene i in cell j is defined as:

$$\hat{\mu}_{ij} = e^{\hat{\beta}_{0i} + \hat{\beta}_i X_j + \hat{\alpha}_i W_j} \quad (6)$$

After computing the estimated mean expression for the count to be simulated, we sample the counts from either the negative binomial distribution or the Poisson distribution. The majority of genes are able to get an estimate $\hat{\phi}_i^0$ from the data, and their simulated counts, y_{ij} , can then be drawn from the negative binomial distribution:

$$y_{ij} \sim \text{NB}(\hat{\mu}_{ij}, \hat{\phi}_i^0) \quad (7)$$

455 However, a small proportion of genes fail to return an estimated dispersion $\hat{\phi}_i^0$,
 456 which is likely caused by their dispersion characteristics. In such cases, we use the
 457 Poisson distribution to simulate their counts:

$$y_{ij} \sim \text{Pois}(\hat{\mu}_{ij}) \quad (8)$$

458 Rescuing outlier genes

459 It is possible to introduce outlier values in the initial simulation of gene counts. Thus,
 460 in this additional step, we aim to check if outlier gene counts exist. If they do, we can
 461 use one of three optional methods to correct them. We check for outliers by comparing
 462 the mean gene expression levels of the simulated data to those of the original data. We
 463 define the mean expression of gene i from the simulated data by λ_i and the original
 464 data λ_i^0 by:

$$\lambda_i = \log \left(\frac{\sum_{j=1}^N y_{ij} + 1}{N} \right) \quad (9)$$

$$\lambda_i^0 = \log \left(\frac{\sum_{j=1}^N y_{ij}^0 + 1}{N} \right) \quad (10)$$

465 For each gene, we can get the absolute difference λ_i^D between the simulated mean
 466 expression and real expression:

$$\lambda_i^D = |\lambda_i - \lambda_i^0| \quad (11)$$

467 Then we obtain the median absolute deviation λ_{MAD} of λ_i^D across all genes. The
 468 genes with λ_i^D larger than a chosen cut-off are labelled as outliers. That is, gene i is
 469 an outlier if $|\lambda_i - \lambda_i^0| > c \cdot \lambda_{\text{MAD}}$. The default value for c we use is 30.

470 • Robust negative binomial

471 The first way to rescue outliers is the robust negative binomial method[34]. Most of

472 outliers are caused by poorly estimated $\hat{\phi}_i^0$. Outlier genes are refitted using a robust
 473 negative binomial regression model with the same design matrix as X_j in (4), and
 474 again obtain the refitted coefficients $\hat{\beta}_{0i}$, $\hat{\beta}_i$, $\hat{\alpha}_i$, and $\hat{\phi}_i^0$. Then we refit a classical neg-
 475 ative binomial GLM with the `glm.nb` function but using the above robustly estimated
 476 coefficients as the starting values. This gives a new set of estimated coefficients and
 477 we use them to update the estimated mean and sample gene counts.

478 • Winsorizing

479 The second strategy to deal with outliers is winsorizing. For each fitted coefficient, we
 480 set a cut-off based on the quantile of its distribution across all genes. The default cut-
 481 offs are the 5% quantile $Q(0.05)$ and the 95% quantile $Q(0.95)$ of the distribution. If
 482 the coefficient of the outlier gene falls in the top or bottom 5% of the distribution,
 483 we use the cut-off value directly as its new fitted coefficients. For example, for an
 484 outlier gene g , suppose we find that its estimated coefficients $\hat{\beta}_{0g}$ is within the range
 485 of 5%-95% of the distribution $\hat{\beta}_{0i}$ across all genes, and the same for $\hat{\beta}_g$, $\hat{\alpha}_g$, then
 486 we keep those three estimated coefficients. But if we find that $\log(\hat{\phi}_g^0)$ is outside
 487 the 5%-95% of the distribution of $\log(\hat{\phi}_g^0)$, and is closer to the $Q(0.05)$ of that
 488 distribution, we set the exponential of $Q(0.05)$ as the new $\hat{\phi}_g^0$. In the subsequent
 489 steps, negative binomial counts for gene g are sampled using $\hat{\mu}_{ij}$ with these new
 490 fitted coefficients, giving revised simulated gene counts.

491 • Trended coefficient

492 The trended coefficient approach is the default method for handling outliers. We
 493 construct the relationship across genes between λ_i^0 and each gene's coefficient by
 494 loess regression. Notice that here we use a logarithmic transformation for the dis-
 495 persion parameter $\hat{\phi}_i^0$. For outlier genes, the loess smoothed value are their new
 496 estimates. After that, counts are drawn from negative binomial distribution with
 497 the estimated $\hat{\mu}_{ij}$ computed using the corrected parameter estimates.

Benchmarking different single cell simulation methods

The version of Splatter is 1.20.0, and the version of scDesign2 is 0.1.0. We also used version 0.9.5 of SPARSim and version of 0.0.0.9000 of SymSim for benchmarking purpose. The definitions of the features compared follow. We denote the raw simulated count matrix by $Y_{G \times N}$, where i refers to gene i in rows and j refers to cell j in columns. Assume there are G genes and N cells in this count matrix. Then the log gene mean λ_i is defined as $\lambda_i = \log\left(\frac{\sum_{j=1}^N Y_{ij} + 1}{N}\right)$. The log library size L_j is: $L_j = \log\left(\sum_{i=1}^G Y_{ij}\right)$. The log gene UMI total is defined as: $T_i = \log\left(\sum_{j=1}^N Y_{ij}\right)$. The gene variance is: $S_i = \frac{1}{N-1} \sum_{j=1}^N \left[Y_{ij} + 1 - \frac{\sum_{j=1}^N (Y_{ij} + 1)}{N}\right]^2$. Denote the gene proportions of zeros by $p_{0i} = \frac{\sum_{j=1}^N I(Y_{ij}=0) + 0.5}{N+1}$. The logit transformation of p_{0i} is: $\text{logit}(p_{0i}) = \log\left(\frac{p_{0i}}{1-p_{0i}}\right)$. Denote the cell proportions of zeros by $\pi_{0j} = \frac{\sum_{i=1}^G I(Y_{ij}=0)}{G}$. The logit transformation is $\text{logit}(\pi_{0j}) = \log\left(\frac{\pi_{0j}}{1-\pi_{0j}}\right)$. Here, $I(Y_{ij} = 0) = 1$ if $Y_{ij} = 0$ is true and $I(Y_{ij} = 0) = 0$ otherwise.

Datasets

In order to benchmark different single cell simulators, we use six datasets sequenced by different platforms and include different scenarios for biological groups as well as batches (Table S1). All datasets start from the raw single cell count matrix without pre-processing. For the hESC dataset some extremely low abundance genes may inappropriately bias the estimation of the GLM parameters, hence the genes expressed in less than 4 cells were filtered out of this dataset. For the other datasets, we use the raw scRNA-seq counts directly.

• Dataset 1: Celine1

The cells in the dataset[35] are a 50-50 mixture of Jurkat and 293T cells in one batch. This dataset is batch 3 of the Celine2 dataset. It is used to study a particular biological issue.

523 • **Dataset 2: Tung**

524 The Tung dataset[36] was generated on the Fluidigm C1 platform and was used
525 to explore the sources of technical variation in scRNA-seq technology. The data
526 was collected from induced pluripotent stem cells (iPSC) of three Yoruba samples
527 (NA19098, NA19101, NA19239). Each sample was independently collected three
528 times, and each replicate was processed using the same reagents. ERCC spike-in
529 controls were added to each sample. The samples were sequenced by the SMARTer
530 protocol. The data is available via: <https://github.com/jdblischak/singleCellSeq>

531 • **Dataset 3: HCC1395**

532 The 10x breast cancer cell line dataset[37] was used as a benchmarking dataset
533 to compare different single cell methods. The cells were collected from a 43-
534 year old female donor. We selected the pure HCC1305 cells sequenced at
535 Loma Linda University. The cells in this dataset have a wide range of library
536 sizes and are used for exploring the library size-only effects. The dataset was
537 downloaded from: https://springernature.figshare.com/collections/A_Multi-center_Cross-platform_Single-cell_RNA-Sequencing_Reference_Dataset/5213468
538

539 • **Dataset 4: Celline2**

540 The dataset[35] was produced for the purpose of investigation of the 10x platform.
541 The cells come from two quite different cell lines: Jurkat and 293T. There are three
542 batches in the dataset. One batch is all Jurkat cells; another batch is all 293T cells,
543 and the third batch is a 50:50 mixture of Jurkat and 293T cells. The three batches
544 were pre-processed separately using the same standard, which involved preserving
545 the features expressed in at least 10 cells and detecting at least 200 genes in each
546 cell.

547 The batch1 count matrix was downloaded from: <https://www.10xgenomics.com/welcome?closeUrl=%2Fresources%2Fdatasets&lastTouchOfferName=Jurkat%20Cells&lastTouchOfferType=Dataset&product=chromium&redirectUrl=>
548
549

550 [%2Fresources%2Fdatasets%2Fjurkat-cells-1-standard-1-1-0](#)

551 The batch2 count matrix was downloaded from: [https://www.10xgenomics.com/welcome?closeUrl=%2Fresources%2Fdatasets&lastTouchOfferName=](https://www.10xgenomics.com/welcome?closeUrl=%2Fresources%2Fdatasets&lastTouchOfferName=293T%20Cells&lastTouchOfferType=Dataset&product=chromium&redirectUrl=%2Fresources%2Fdatasets%2F293-t-cells-1-standard-1-1-0)

552 [com/welcome?closeUrl=%2Fresources%2Fdatasets&lastTouchOfferName=](#)

553 [293T%20Cells&lastTouchOfferType=Dataset&product=chromium&redirectUrl=](#)

554 [%2Fresources%2Fdatasets%2F293-t-cells-1-standard-1-1-0](#)

555 The batch3 count matrix was downloaded from: [https://www.10xgenomics.com/welcome?closeUrl=%2Fresources%2Fdatasets&lastTouchOfferName=50%](https://www.10xgenomics.com/welcome?closeUrl=%2Fresources%2Fdatasets&lastTouchOfferName=50%25%3A50%25%20Jurkat%3A293T%20Cell%20Mixture&lastTouchOfferType=Dataset&product=chromium&redirectUrl=%2Fresources%2Fdatasets%2F50-percent-50-percent-jurkat-293-t-cell-mixture-1-standard-1-1-0)

556 [com/welcome?closeUrl=%2Fresources%2Fdatasets&lastTouchOfferName=50%](#)

557 [25%3A50%25%20Jurkat%3A293T%20Cell%20Mixture&lastTouchOfferType=](#)

558 [Dataset&product=chromium&redirectUrl=%2Fresources%2Fdatasets%](#)

559 [2F50-percent-50-percent-jurkat-293-t-cell-mixture-1-standard-1-1-0](#)

560 • **Dataset 5: hESC**

561 Naïve and primed human embryonic stem cells (hESCs) were profiled to investigate

562 the heterogeneity and developmental transition within each pluripotency state[38].

563 Naïve hESCs were grown in N2B27 medium with titrated 2 inhibitors (PD0325091

564 and CHIR99021), Leukemia inhibitor and Go6083 inhibitor, while primed hESCs

565 were cultured in E8 media. Naïve hESCs were processed in two batches: the first

566 batch contained 96 cells in each state, and the second batch contained 384 cells

567 in each state. Primed hESCs are in the same condition for the first two batches

568 as the naïve hESCs, but the primed cells have an additional 384 cells in a third

569 batch. The cells were prepared and sequenced by the SmartSeq2 protocol. The

570 data is available at: [https://bioconductor.org/packages/release/data/experiment/](https://bioconductor.org/packages/release/data/experiment/vignettes/scRNAseq/inst/doc/scRNAseq.html)

571 [vignettes/scRNAseq/inst/doc/scRNAseq.html](#)

572 • **Dataset 6: CLL**

573 This dataset[39] was part of an investigation of Venetoclax (VEN) resistance. The

574 majority of cells are B cells. The cells were treated with dimethyl sulfoxide (DMSO),

575 VEN and combinations of VEN and other treatments for one week. The data was

576 generated on the CEL-Seq2 platform over 6 plates (LC89, L91, L93, L95, L96,

LC99). Granta cell line cells were included in each plate. This dataset is the most challenging one for three reasons. Firstly, it has multiple batches and biological groups. Secondly, associations exist between library size, batch and biology. Lastly, except for the Granta cell line, each drug treatment condition is dominant in one batch. In other words, different cell types are not evenly mixed in each batch. This dataset can be accessed by requesting permission from the authors.

Declarations

Ethics approval and consent to participate

Not applicable

Consent for publication

Not applicable

Availability of data and materials

See the section of "Dataset". The code is available at Github (<https://github.com/jiananwehi/GLMsim.git>). The R package is under a GPL-3.0 license.

Competing interests

The authors declare that they have no competing interests.

Funding

Jianan Wang is supported by Walter and Eliza Hall International Scholarship and CSL Translational Data Science Scholarship.

Authors' contributions

JW developed the method and conducted all analysis. LC contributed to statistical methods for GLM parameter fitting and non-UMI data. RT designed and generated data for the CLL study. BP contributed to trended coefficients and supervision. TS oversaw the whole project. All authors read and approved the final manuscript.

Acknowledgements

We would like to thank to Givanna Putri for providing suggestions of parallel computation. Our thanks also to Chris Woodruff and Luke Gandolfo for their comments on the manuscript.

References

- [1] Aissa, A.F., Islam, A.B., Ariss, M.M., Go, C.C., Rader, A.E., Conrardy, R.D., Gajda, A.M., Rubio-Perez, C., Valyi-Nagy, K., Pasquinelli, M., *et al.*: Single-cell transcriptional changes associated with drug tolerance and response to combination therapies in cancer. *Nature communications* **12**(1), 1628 (2021)
- [2] Qiu, C., Cao, J., Martin, B.K., Li, T., Welsh, I.C., Srivatsan, S., Huang, X., Calderon, D., Noble, W.S., Disteche, C.M., *et al.*: Systematic reconstruction of cellular trajectories across mouse embryogenesis. *Nature genetics* **54**(3), 328–341 (2022)
- [3] Kuhn, A., Thu, D., Waldvogel, H.J., Faull, R.L., Luthi-Carter, R.: Population-specific expression analysis (psea) reveals molecular changes in diseased brain. *Nature methods* **8**(11), 945–947 (2011)
- [4] Garcia, A.X., Xu, J., Cheng, F., Rupp, E., Schäffer, A.A.: Altered gene expression in excitatory neurons is associated with alzheimer's disease and its higher

- incidence in women. *Alzheimer's & Dementia: Translational Research & Clinical Interventions* **9**(1), 12373 (2023)
- [5] Iacono, G., Massoni-Badosa, R., Heyn, H.: Single-cell transcriptomics unveils gene regulatory network plasticity. *Genome biology* **20**, 1–20 (2019)
- [6] Zappia, L., Phipson, B., Oshlack, A.: Exploring the single-cell rna-seq analysis landscape with the scrna-tools database. *PLoS computational biology* **14**(6), 1006245 (2018)
- [7] Cao, Y., Yang, P., Yang, J.Y.H.: A benchmark study of simulation methods for single-cell rna sequencing data. *Nature communications* **12**(1), 6911 (2021)
- [8] Lähnemann, D., Köster, J., Szczurek, E., McCarthy, D.J., Hicks, S.C., Robinson, M.D., Vallejos, C.A., Campbell, K.R., Beerenwinkel, N., Mahfouz, A., *et al.*: Eleven grand challenges in single-cell data science. *Genome biology* **21**(1), 1–35 (2020)
- [9] You, Y., Dong, X., Wee, Y.K., Maxwell, M.J., Alhamdoosh, M., Smyth, G.K., Hickey, P.F., Ritchie, M.E., Law, C.W.: Modeling group heteroscedasticity in single-cell rna-seq pseudo-bulk data. *Genome biology* **24**(1), 107 (2023)
- [10] Zhou, Y., Peng, M., Yang, B., Tong, T., Zhang, B., Tang, N.: scdlc: a deep learning framework to classify large sample single-cell rna-seq data. *BMC genomics* **23**(1), 504 (2022)
- [11] Karikomi, M., Zhou, P., Nie, Q.: Durian: an integrative deconvolution and imputation method for robust signaling analysis of single-cell transcriptomics data. *Briefings in Bioinformatics* **23**(4), 223 (2022)
- [12] Mallick, H., Chatterjee, S., Chowdhury, S., Chatterjee, S., Rahnavard, A., Hicks,

643 S.C.: Differential expression of single-cell rna-seq data using tweedie models.
644 *Statistics in medicine* **41**(18), 3492–3510 (2022)

645 [13] Jiang, R., Sun, T., Song, D., Li, J.J.: Statistics or biology: the zero-inflation
646 controversy about scrna-seq data. *Genome biology* **23**(1), 1–24 (2022)

647 [14] Zappia, L., Phipson, B., Oshlack, A.: Splatter: simulation of single-cell rna
648 sequencing data. *Genome biology* **18**(1), 174 (2017)

649 [15] Baruzzo, G., Patuzzi, I., Di Camillo, B.: Sparsim single cell: a count data simulator
650 for scrna-seq data. *Bioinformatics* **36**(5), 1468–1475 (2020)

651 [16] Assefa, A.T., Vandesompele, J., Thas, O.: Spsimseq: semi-parametric simulation
652 of bulk and single-cell rna-sequencing data. *Bioinformatics* **36**(10), 3276–3278
653 (2020)

654 [17] Li, W.V., Li, J.J.: A statistical simulator scdesign for rational scrna-seq experi-
655 mental design. *Bioinformatics* **35**(14), 41–50 (2019)

656 [18] Sun, T., Song, D., Li, W.V., Li, J.J.: scdesign2: a transparent simulator that gen-
657 erates high-fidelity single-cell gene expression count data with gene correlations
658 captured. *Genome biology* **22**(1), 163 (2021)

659 [19] Su, K., Wu, Z., Wu, H.: Simulation, power evaluation and sample size recommen-
660 dation for single-cell rna-seq. *Bioinformatics* **36**(19), 4860–4868 (2020)

661 [20] Vieth, B., Ziegenhain, C., Parekh, S., Enard, W., Hellmann, I.: powsimr: power
662 analysis for bulk and single cell rna-seq experiments. *Bioinformatics* **33**(21), 3486–
663 3488 (2017)

664 [21] Zhang, X., Xu, C., Yosef, N.: Simulating multiple faceted variability in single cell
665 rna sequencing. *Nature communications* **10**(1), 2611 (2019)

- 666 [22] Sarkar, H., Srivastava, A., Patro, R.: Minnow: a principled framework for rapid
667 simulation of dscrna-seq data at the read level. *Bioinformatics* **35**(14), 136–144
668 (2019)
- 669 [23] Korsunsky, I., Millard, N., Fan, J., Slowikowski, K., Zhang, F., Wei, K.,
670 Baglaenko, Y., Brenner, M., Loh, P.-r., Raychaudhuri, S.: Fast, sensitive and
671 accurate integration of single-cell data with harmony. *Nature methods* **16**(12),
672 1289–1296 (2019)
- 673 [24] Haghverdi, L., Lun, A.T., Morgan, M.D., Marioni, J.C.: Batch effects in single-
674 cell rna-sequencing data are corrected by matching mutual nearest neighbors.
675 *Nature biotechnology* **36**(5), 421–427 (2018)
- 676 [25] ATL, L.: Further MNN algorithm development (2018). [https://MarioniLab.](https://MarioniLab.github.io/FurtherMNN2018/theory/description.html)
677 [github.io/FurtherMNN2018/theory/description.html](https://MarioniLab.github.io/FurtherMNN2018/theory/description.html)
- 678 [26] Salim, A., Molania, R., Wang, J., De Livera, A., Thijssen, R., Speed, T.: RUV-III-
679 NB: normalization of single cell RNA-seq data. *Nucleic Acids Research* **50**(16),
680 96–96 (2022)
- 681 [27] Lin, Y., Ghazanfar, S., Wang, K.Y.X., Gagnon-Bartsch, J.A., Lo, K.K., Su, X.,
682 Han, Z.-G., Ormerod, J.T., Speed, T.P., Yang, P., Yang, J.Y.H.: scmerge leverages
683 factor analysis, stable expression, and pseudoreplication to merge multiple single-
684 cell rna-seq datasets. *Proceedings of the National Academy of Sciences* **116**(20),
685 9775–9784 (2019)
- 686 [28] Hie, B., Bryson, B., Berger, B.: Efficient integration of heterogeneous single-cell
687 transcriptomes using scanorama. *Nature biotechnology* **37**(6), 685–691 (2019)
- 688 [29] Lun, A.T., McCarthy, D.J., Marioni, J.C.: A step-by-step workflow for low-level
689 analysis of single-cell rna-seq data with bioconductor. *F1000Research* **5** (2016)

- [30] Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., Mauck, W.M., Hao, Y., Stoeckius, M., Smibert, P., Satija, R.: Comprehensive integration of single-cell data. *Cell* **177**(7), 1888–1902 (2019)
- [31] Risso, D., Perraudeau, F., Gribkova, S., Dudoit, S., Vert, J.-P.: A general and flexible method for signal extraction from single-cell rna-seq data. *Nature communications* **9**(1), 284 (2018)
- [32] Gandolfo, L.C., Speed, T.P.: Rle plots: Visualizing unwanted variation in high dimensional data. *PloS one* **13**(2), 0191629 (2018)
- [33] Venables, W.N., Ripley, B.D.: *Modern Applied Statistics with S*, 4th edn. Springer, New York (2002)
- [34] Aeberhard, W.H., Cantoni, E., Heritier, S.: Robust inference in the negative binomial regression model with an application to falls data. *Biometrics* **70**(4), 920–931 (2014)
- [35] Zheng, G.X., Terry, J.M., Belgrader, P., Ryvkin, P., Bent, Z.W., Wilson, R., Ziraldo, S.B., Wheeler, T.D., McDermott, G.P., Zhu, J., *et al.*: Massively parallel digital transcriptional profiling of single cells. *Nature communications* **8**(1), 14049 (2017)
- [36] Tung, P.-Y., Blischak, J.D., Hsiao, C.J., Knowles, D.A., Burnett, J.E., Pritchard, J.K., Gilad, Y.: Batch effects and the effective design of single-cell gene expression studies. *Scientific reports* **7**(1), 39921 (2017)
- [37] Chen, X., Yang, Z., Chen, W., Zhao, Y., Farmer, A., Tran, B., Furtak, V., Moos Jr, M., Xiao, W., Wang, C.: A multi-center cross-platform single-cell rna sequencing reference dataset. *Scientific Data* **8**(1), 39 (2021)

- 713 [38] Messmer, T., Meyenn, F., Savino, A., Santos, F., Mohammed, H., Lun, A.T.L.,
714 Marionni, J.C., Reik, W.: Transcriptional heterogeneity in naive and primed human
715 pluripotent stem cells at single-cell resolution. *Cell reports* **26**(4), 815–824 (2019)
- 716 [39] Thijssen, R., Tian, L., Anderson, M.A., Flensburg, C., Jarratt, A., Garnham,
717 A.L., Jabbari, J.S., Peng, H., Lew, T.E., Teh, C.E., *et al.*: Single-cell multiomics
718 reveal the scale of multilayered adaptations enabling cll relapse during venetoclax
719 therapy. *Blood, The Journal of the American Society of Hematology* **140**(20),
720 2127–2141 (2022)

Table 1 Summary of some single cell simulators.

Method	Library	Batch	Biology	Library $\times$ Batch	Library $\times$ Biology	Batch $\times$ Biology
Splatter	✓	✓	✓	×	×	×
SymSim	×	✓	✓	×	×	×
SPARSim	✓	✓	✓	×	×	×
scDesign2	×	×	✓	×	×	×

Library: library size. Library $\times$ Batch: library size associated with batch. Library $\times$ Biology: library size associated with biology. Batch $\times$ Biology: batch associated with biology.

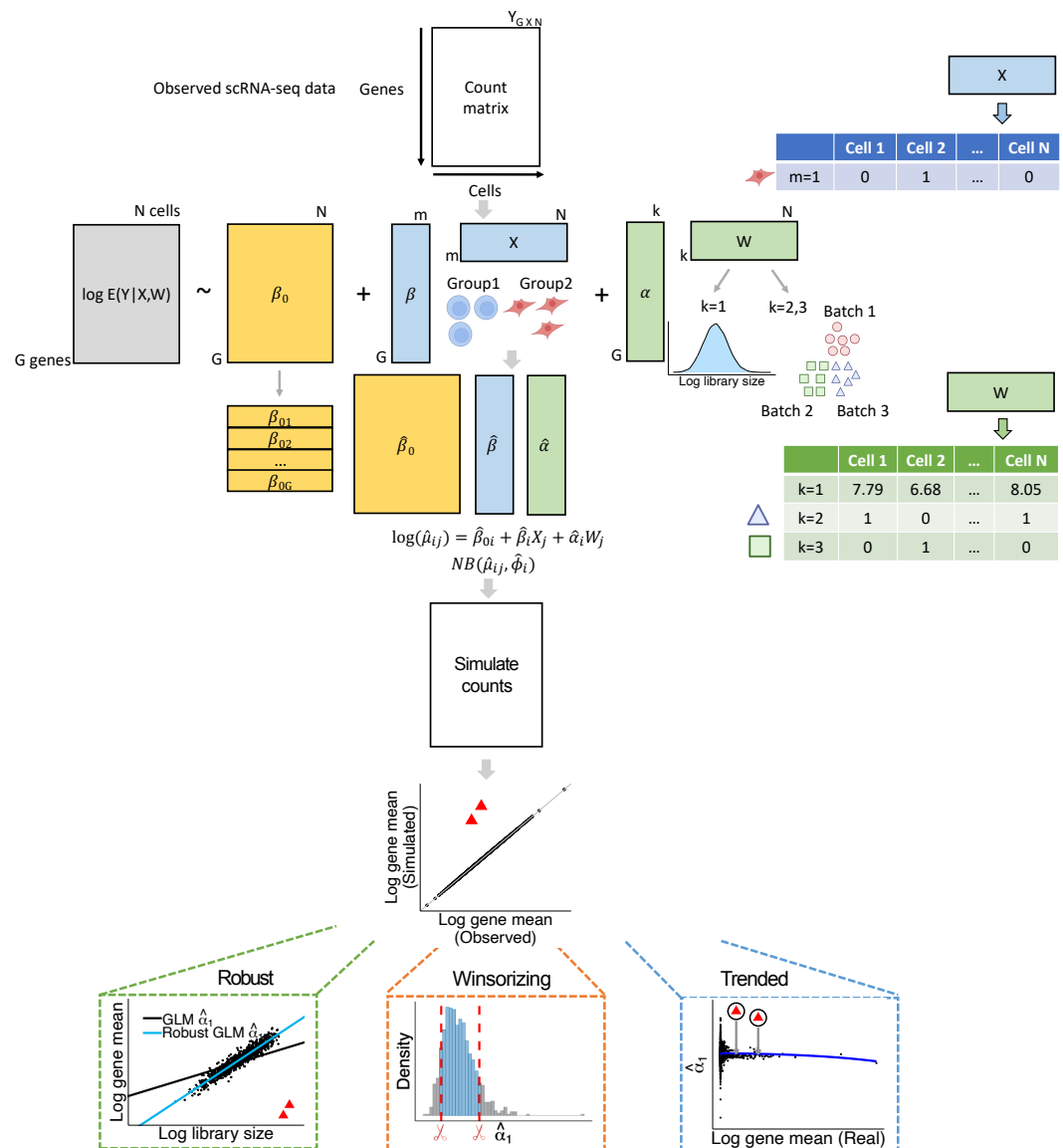


Fig. 1 Overview of GLMsim. GLMsim starts from an observed scRNA-seq count matrix, where rows are genes and columns are cells. For each gene, GLMsim applies a generalized linear model to estimate biological and technical coefficients. Next, GLMsim samples single cell counts from a negative binomial distribution with a mean computed using previously estimated coefficients, and dispersion estimated from the observed data. In the last optional step, GLMsim checks if outlier genes exist, and uses one of the three alternative trending methods to deal with outliers: robust negative binomial GLM, winsorizing the coefficients and trending the coefficients.

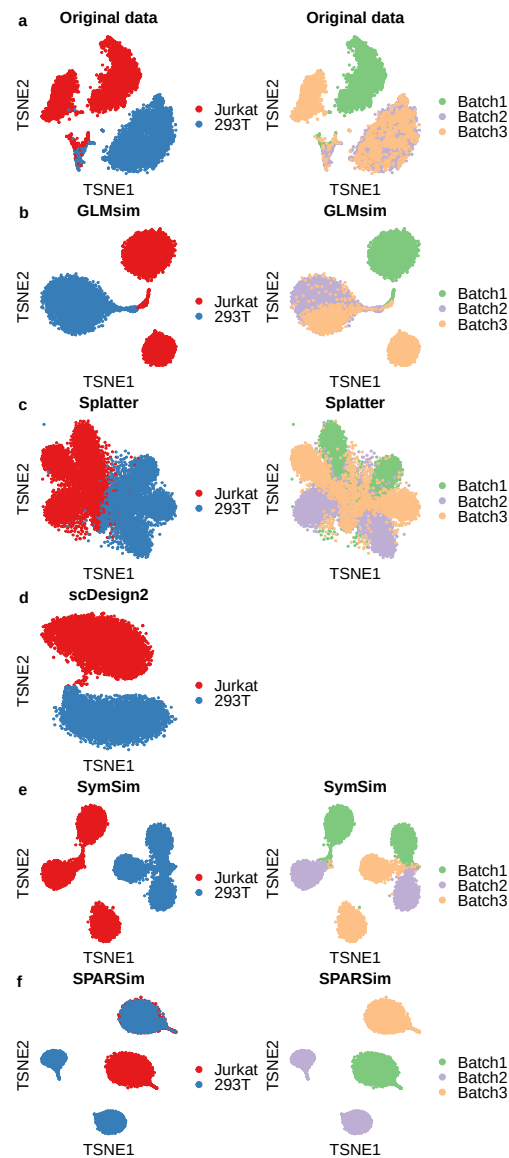


Fig. 2 The cellline2 experimental data and data simulated by different methods. Association exists between biological and batch effects. (a-f) tSNE plot of real and simulated data colored by biological groups and batch groups. (a) Real data. (b) GLMsim simulated data. (c) Splatter simulated data. (d) scDesign2 simulated data. Since scDesign2 cannot simulate batch effect, the tSNE plot colored by batch is not shown here. (e) SymSim simulated data. (f) SPARSim simulated data.

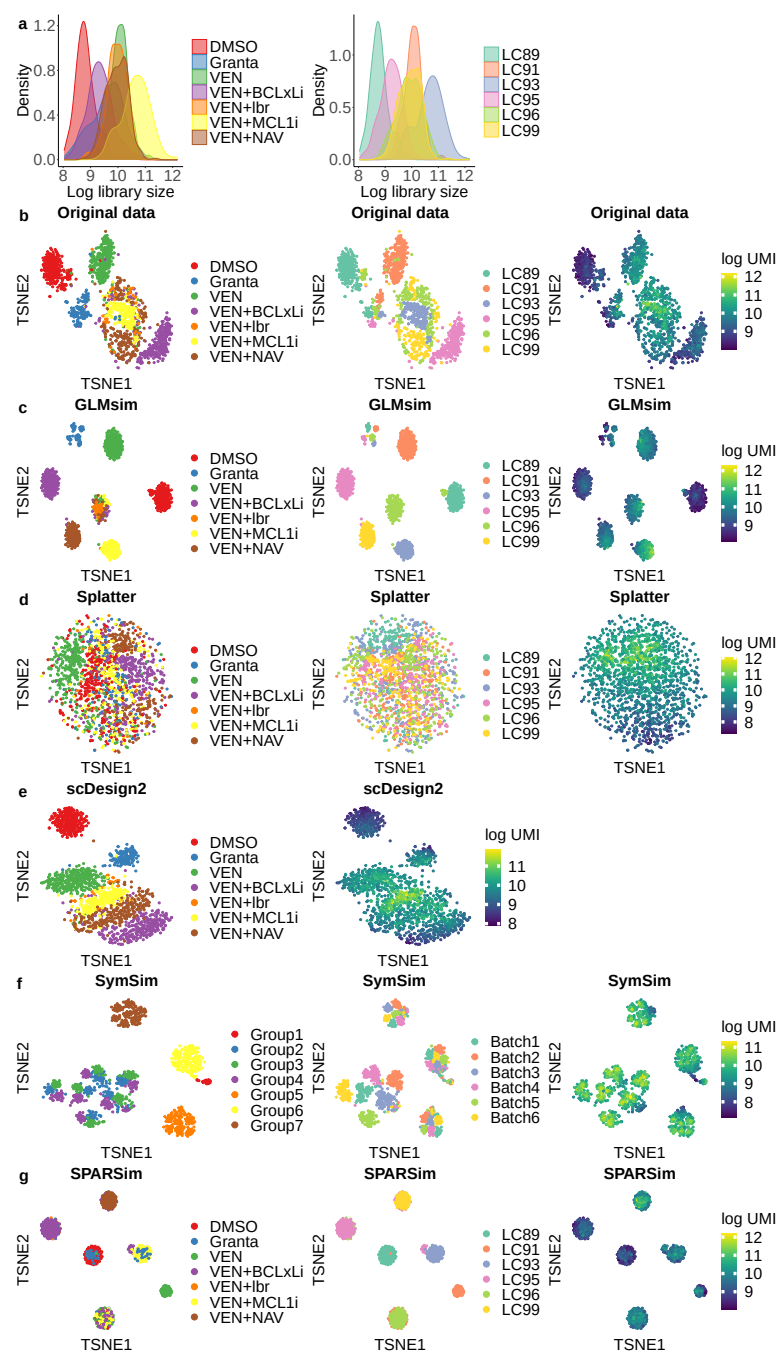


Fig. 3 The CLL experimental data and data simulated by different methods. Associations exist between library size, biological and batch effects. (a) The log library size distribution across different biological groups and batch groups. (b-g) tSNE plots of real and simulated data colored by biological groups, batch groups and library sizes. (b) Real data. (c) GLMSim simulated data. (d) Splatter simulated data. (e) scDesign2 simulated data. Since scDesign2 cannot simulate the batch effect, the tSNE plot colored by batch is not shown here. (f) SymSim simulated data. (g) SPARSim simulated data.

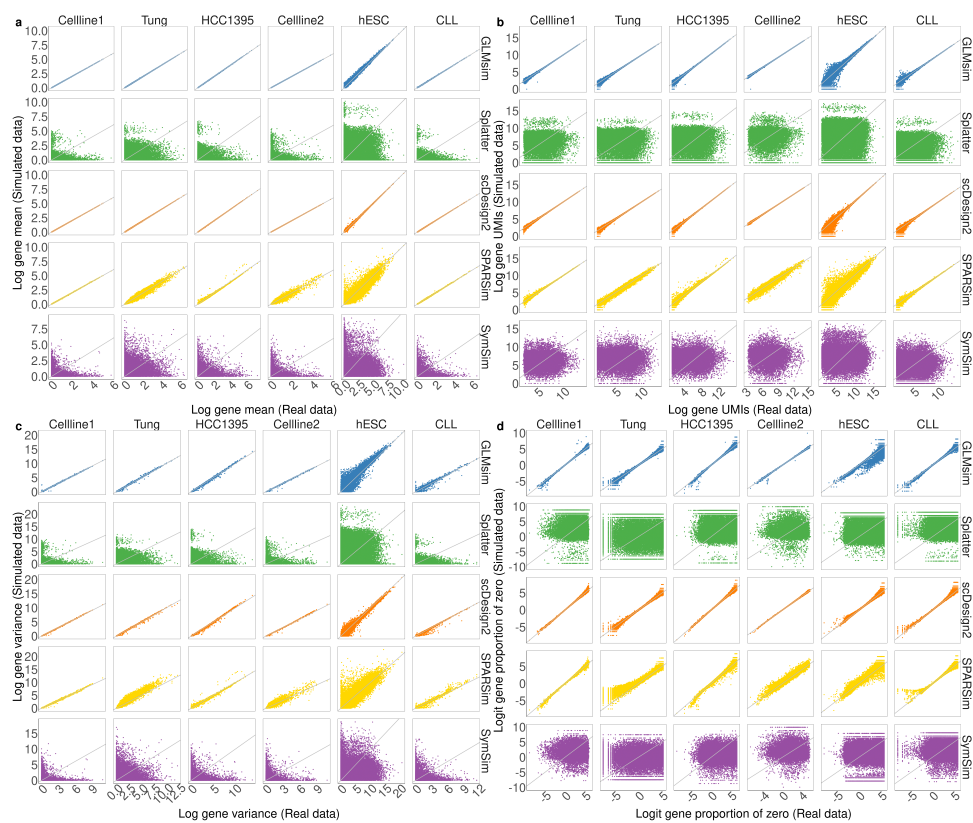


Fig. 4 Pairwise comparison of gene-specific features between simulated data and original data. Each row represents a simulation method. Each column represents a dataset. The x axis of each plot refers to the metric from the original data, and the y axis refers to the metric computed from the simulated data. Each dot represents a gene. (a) Log gene mean. (b) Log gene UMIs. (c) Log gene variance. (d) Logit gene proportion of zero.

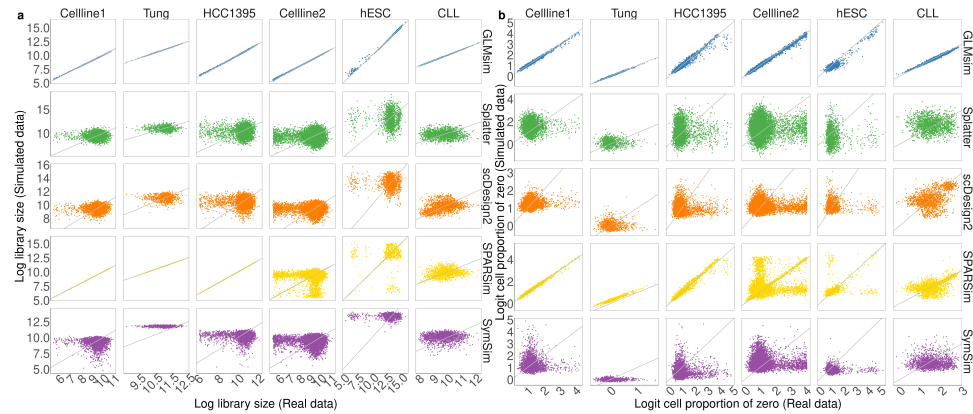


Fig. 5 Pairwise comparison of cell-wise metrics between simulated data and original data. Each row represents a simulation method. Each column represents a dataset. The x axis of each plot refers to the metric from the original data, and the y axis refers to the metric computing from the simulated data. Each dot represents a cell. (a) Log library size. (b) Logit cell proportion of zero.

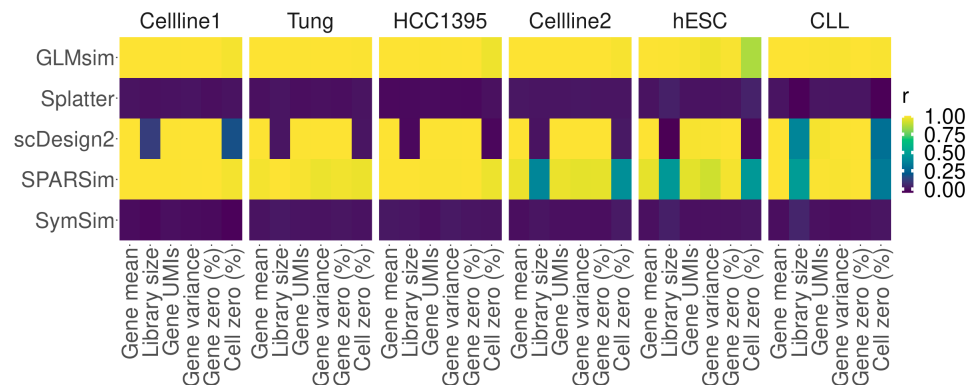


Fig. 6 Spearman correlations between features of the simulated data and reference data. Each column stands for a gene- or cell-level metric. Each row stands for a simulation method. The column panels display different datasets. The heatmap is colored by Spearman correlation of a metric between the simulated data and the reference data.

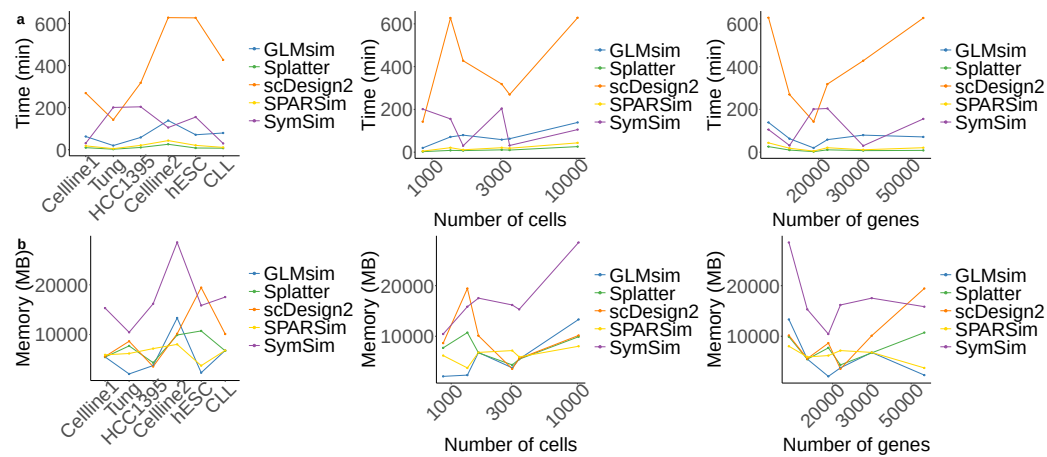


Fig. 7 Computational scalability of different simulation methods. (a) Runtime of different methods across all datasets. (b) Memory usage of different methods across all datasets. (a,b) The scalability is also measured by the scale of the data: the number of genes and cells.

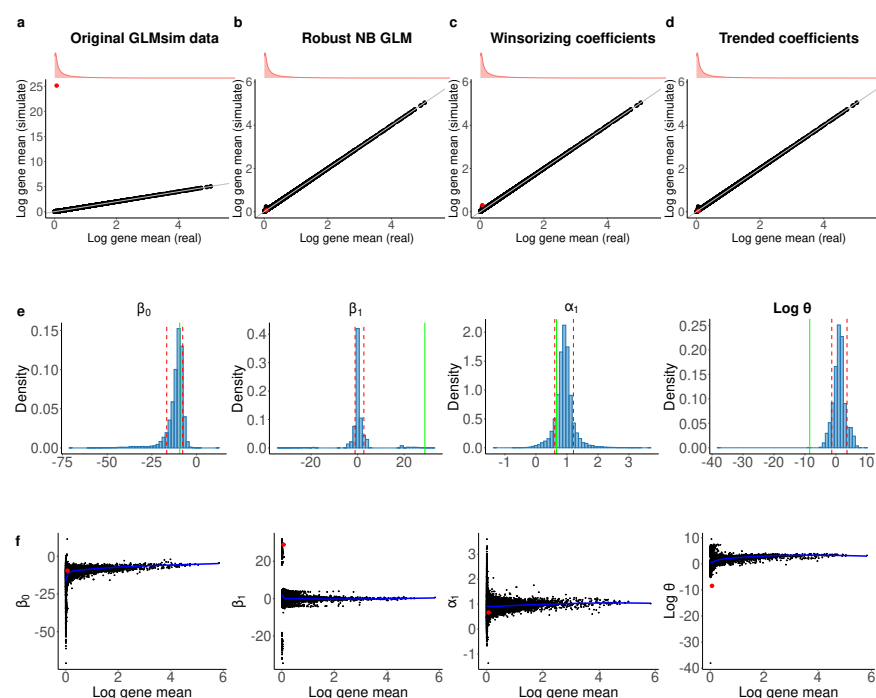


Fig. 8 GLMsim handles outlier genes in the Cellline1 dataset. The outlier gene is shown by the red dot. (a-d) The comparison of log mean expression between original data and GLMsim simulated data. (a) GLMsim simulated data is from the original simulated data without handling the outlier. (b) GLMsim simulated data after robust NB GLM dealing with outlier gene. (c) GLMsim simulated data after winsorizing the coefficients for the outlier gene. (d) GLMsim simulated data by the trended coefficients to the outlier gene. (e) The distribution of each estimated coefficient across genes. The red lines are cut-offs for each estimated coefficient, which is $Q(0.1)$ and $Q(0.9)$ for each coefficient. The green line is the estimate coefficient for the outlier gene. (f) The relationship between each estimated coefficient and the log gene mean. The blue line represents the loess trended line.

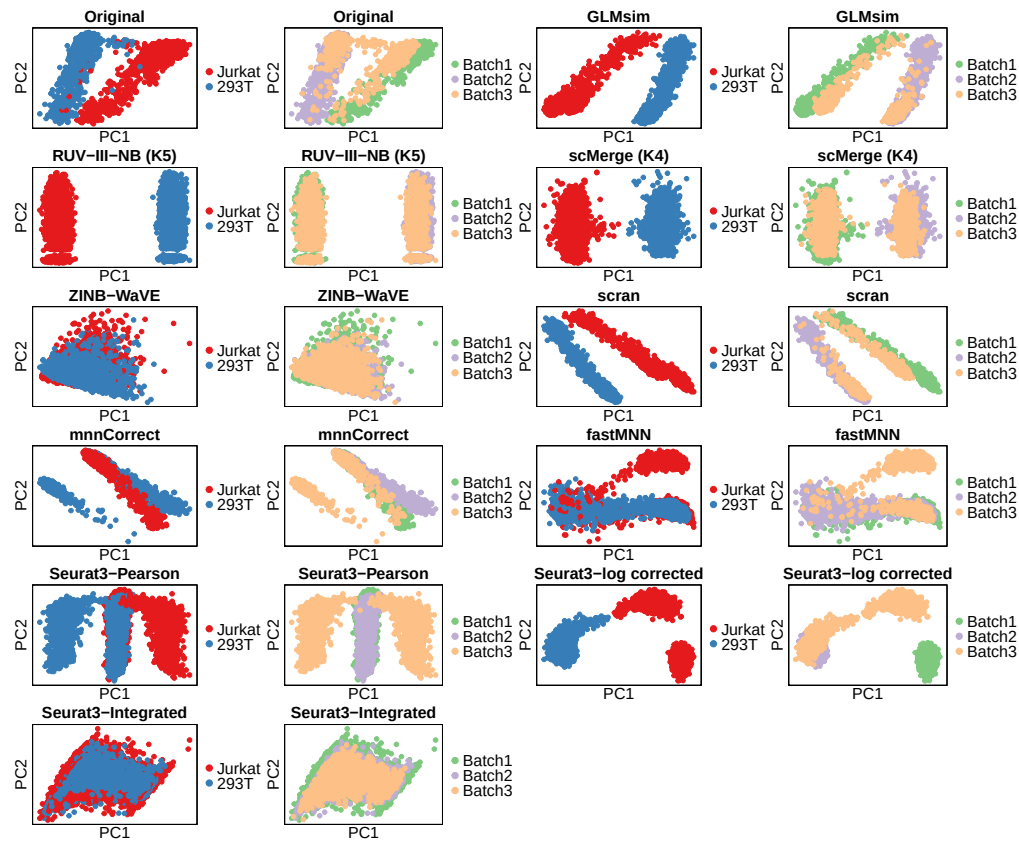


Fig. 9 PCA plot of integration on simulated Celine2 data by different methods. GLMsim was used to simulate the Celine2 dataset, then different integration methods were used on the simulated data. The first two principal components are shown in the plot. The cells were colored by cell lines and batches. The first pairs of plots show the original GLMsim simulated Celine2 data. For RUV-III-NB, the plot is made based on log PAC, the log of the percentile-adjusted counts. For Seurat, three data formats are used to do the plots: the Pearson residual, log corrected data and integrated data.