Inferring Cell-Type-Specific Co-Expressed Genes from Single Cell Data

Xinning Shan[1] and Hongyu Zhao[1*]

[1]Department of Biostatistics, Yale University, New Haven, CT, USA.

*Corresponding author(s). E-mail(s): hongyu.zhao@yale.edu;

Abstract

Background: Cell-type-specific gene co-expression networks are widely used to characterize gene relationships. Although many methods have been developed to infer such co-expression networks from single-cell data, the lack of consideration of false positive control in many evaluations may lead to incorrect conclusions because higher reproducibility, higher functional coherence, and a larger overlap with known biological networks may not imply better performance if the false positives are not well controlled.
Results: In this study, we have developed an efficient and effective simulation tool to derive empirical p-values in co-expression inference to appropriately control false positives in assessing method performance. We studied the power of the p-value-based approach in inferring cell-type-specific co-expressions from single-cell data using both simulated and real data. We also highlight the need to adjust for random overlaps between the inferred and known networks when the number of selected correlated gene pairs varies substantially across different methods. We further illustrate the expression level bias in known biological networks and the impact of such bias in method assessment.
Conclusion: Our study indicates the importance of controlling false positives in the inference of co-expressed genes to achieve more reliable results and proposes a simulation-based p-value method to achieve this.

Keywords: Single cell, Cell-type-specific, Co-expression networks

1 Background

With the development of single-cell technology and the potential of inferring cell-type-specific gene co-expression networks from these data to understand relationships among genes, many methods have been developed for estimating such networks through single-cell RNA-sequencing (scRNA-seq) data. A common way to evaluate the performance of a specific estimation method using real data is to compare the inferred networks with known biological networks including protein-protein interaction (PPI) networks (e.g., HIPPIE [1], STRING [1–5]), metabolic pathways (e.g., Reactome [1]), signaling networks (e.g., OmniPath [1]), and TF-targets [3, 4, 6]. Under the assumption that the inferred gene pairs are likely to be re-discovered in independent datasets of the same system and disease if they are truly correlated, another commonly adopted evaluation metric is reproducibility across independent datasets [1, 3, 7, 8]. Apart from these two approaches, there exist additional methods for evaluating the estimated networks, such as functional

coherence that assesses the capability of predicting a gene's functions based on its neighbors' functions [1, 8], Gene Ontology (GO) enrichment analysis on modules identified through the estimated gene co-expression networks [4, 9, 10], and comparing inferred networks under normal and disease conditions [4, 9].

To evaluate the estimated gene co-expression networks, we first need to identify correlated gene pairs. There are two commonly used approaches for selecting correlated gene pairs in real data analysis: p-value-based and correlation-strength-based. The p-value-based approach is meaningful only if the p-value is well-calibrated. However, Su et al. [11] reported that there is type I error inflation for some existing cell-type-specific gene co-expression estimation methods. Furthermore, not all existing methods output p-values for rigorous statistical inference, leaving uncertainty regarding the potential inflation of type I errors. The p-values for the Pearson and Spearman correlations reported in Su et al. [11] were calculated based on Student's t-distribution, which may not be proper; this issue is discussed in detail in Section 2.1. The correlation-strength-based approach selects correlated gene pairs based on the estimated correlation strength, instead of p-values, and is more commonly used because of its convenience [1, 3–6, 8, 12, 13]. However, the correlation-strength-based approach does not consider false positive control.

In this paper, we propose a simulation-based method to mimic real data with uncorrelated gene pairs to generate empirical p-values for gene pairs in Section 2.2. We investigate the performance of the two commonly used gene pair selection approaches in Sections 2.3 and 2.4, and show that the p-value-based approach may lead to more reliable results. Furthermore, we recommend considering random overlaps between the inferred and known networks and adjusting the expression bias in known biological networks when benchmarking gene co-expression estimation methods in Section 2.5. Our pipeline offers a fair comparison among different gene co-expression inference methods, which is crucial for method comparisons and future developments on this important topic.

2 Results

2.1 More suitable p-values are needed for evaluations of different methods

Su et al. [11] examined the p-value distribution of Noise Regularization [5], Normalisr [10], Pearson correlation on log-transformed data (Pearson), Spearman correlation on log-transformed data (Spearman), Pearson correlation on the analytic Pearson residuals (Analytic PR) [14], Pearson correlation on the normalized data obtained via sctransform (sctransform) [15] and CS-CORE [11] on permuted single-cell RNA count data. It was found that only Normalisr, Pearson correlation on the sctransform normalized data and CS-CORE have well-calibrated type I errors and the other methods have inflated false positives [11]. These results suggest that we need more proper p-values for those methods without appropriate type I error control.

We may also need more suitable p-values for the Pearson correlation on the sctransform normalized data since the associated p-values, calculated through Student's t-distribution are questionable in this context. More specifically, the t-statistic for the Pearson correlation is calculated based on the correlation coefficient

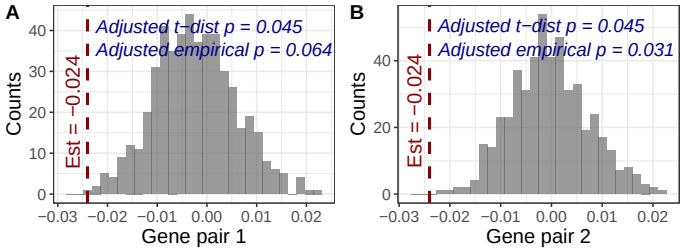

Fig. 1 P-values from Student's t-distribution may not reflect the underlying null distribution. Gene pairs were selected from the simulation described in Section 2.3 where gene pair 1 is independent and gene pair 2 is correlated. The figures display the histograms of their correlation estimates under the independent simulations, with the observed estimate indicated by a dashed dark red line. The adjusted two-sided t-distribution and empirical p-values are displayed in dark blue.

(r) and the number of single cells used (n),

$$t = \frac{r\sqrt{n-2}}{\sqrt{1-r^2}}. \tag{1}$$

By equation(1), gene pairs with the same r and n yield identical p-values when estimated via Student's t-distribution, even though they may have different means and variances under the null hypothesis (Figures 1A and 1B). There are similar concerns for the Spearman correlation.

To summarize, there was no discussion of statistical significance for some co-expression estimation methods (e.g., Noise Regularization), and the p-values of these methods used for comparison in Wang [10] and Su et al. [11] may not calibrate well for the reasons discussed above. Thus, to achieve more precise control of type I errors and fair comparisons, there is a need for a more suitable method to estimate the p-values. To achieve this goal, we have developed a pipeline to advocate the use of empirical p-values when inferring co-expression between gene pairs.

2.2 Estimation of empirical pairwise p-values

To obtain an empirical p-value for each gene pair, we need to reference the estimated correlation to a distribution for the gene pair under the null hypothesis of no correlation between the genes. The challenge of deriving such distributions arises from the difficulty in gathering real data where the genes are independent. Therefore, we opt to use in silico methods to achieve this goal through either permutation or simulation. For the permutation approach, Su et al. [11] permuted the gene ratio (counts/sequencing depth) for each gene across cells and converted it to count data based on pre-specified sequencing depth. Specht and Li [16] permuted the normalized expressions across cells for each gene. For simulation, many methods have been developed for simulating count data in the context of scRNA-seq. Crowell et al. [17] found that ZINB-WaVE [18] and scDesign2 [19] were the top two methods for simulating data containing cells from a single batch and cluster compared with 13 other simulation methods. In another benchmark paper, ZINB-WaVE and SPARSim [20] were found to exhibit greater similarity to the original data in terms of both data properties and biological signals [21]. More recently, scDesign2 has been further developed into scDesign3 [22]. Even though these methods can closely mimic real data, some limitations exists. ZINB-WaVE does

not consider the correlation structure, whereas scDesign2 and scDesign3 are computationally demanding requiring several hours for a single simulation (Figures 2C and 2D). SPARSim assumes fixed sequencing depths and independent genes.

In this work, we developed a simulation method called scSimu (Single-Cell SIMUlation), that improves the simulation method used in Su et al. [11] to generate scRNA-seq count data from real data. scSimu is specifically developed for evaluating gene co-expression networks estimated from Drop-seq scRNA-seq data. We evaluated the performance of scSimu based on two tasks, simulating genes that are independent and simulating genes that are correlated. For the independent situation, we considered nine other methods (Table 1). ZINB-WaVE, SymSim, POWSC, and Hierarchicell were excluded because they are not able to generate independent genes. We did not compare with the permutation method proposed by Specht and Li [16] as it permutes normalized data and cannot be applied to co-expression estimation methods that require a count matrix as input. For correlated genes, we compared scSimu with 11 other simulation methods (Table 1). ESCO is not suitable for studies focused on correlation analysis because the correlated gene pairs are selected randomly in ESCO although it incorporates correlation structures. We also did not consider SPsimSeq [23] for both tasks because of the extensive computational time required by this method, that utilizes a log-linear model density estimation combined with a Gaussian copula. For example, for a dataset with 33,538 genes and 4,724 cells, SPsimSeq took 2.5 hours to simulate independent genes, and 30 hours to simulate correlated genes.

Table 1 Simulation methods compared in our evaluation

Method	Simulation Model	Considered for correlated genes (COR)	Considered for independent genes (IND)	Considered for correlation structures
Splat [24]	Gamma-Poisson hierarchical model	✓	✓	✗
powsimR [25]	Negative binomial (NB)/Zero-inflated negative binomial (ZINB)	✓	✓	✗
ZINB-WaVE [18]	ZINB based wanted variation extraction	✓	✗	✗
SymSim [26]	Kinetic model (Intrinsic variation: Beta-Poisson)	✓	✗	✗
scDesign2 [19]	Poisson, Zero-inflated Poisson (ZIP), ZINB or NB with Gaussian copula	✓	✓	✓
scDesign3 [22]	Poisson, ZIP, ZINB, NB, Bernoulli, or normal with Gaussian copula or Vine copula	✓	✓	✓
SPARSim [20]	Gamma-Multivariate hypergeometric	✓	✓	✗
POWSC [27]	Mixture of ZIP and log-normal Poisson distributions	✓	✗	✗
ESCO [28]	Gamma-Poisson with Gaussian copula	✗	✓	✓
Hierarchicell [29]	NB with within sample correlation	✓	✗	✗
muscat [30]	NB or ZINB	✓	✓	✗
SCRIP [31]	Gamma-Poisson or Beta-Gamma-Poisson	✓	✓	✗
Permutation [11]	Permute gene ratio	✗	✓	✗

We used a combination of the kernel density based global two-sample comparison (KDE) test [21] and the Kolmogorov–Smirnov (KS) test [17] to evaluate the similarity between the simulated data and real data (see Section 5.2 for more information). The KDE test assesses the overall divergence between two distributions, whereas the KS test quantifies the largest discrepancy between them. For both the KDE test and the KS

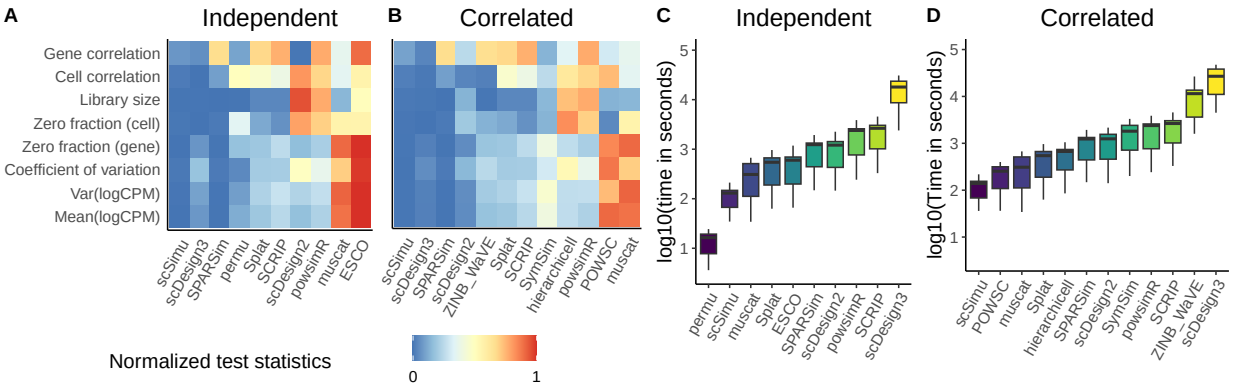

Fig. 2 Benchmark of the simulation methods. A and B. The x-axis is the simulation method. The y-axis is the evaluation metric. The color represents the normalized test statistics of the combination of the KDE test and the KS test. The methods are ordered based on the average normalized test statistics. **A.** Performance of simulating independent genes. **B.** Performance of simulating correlated genes. **C and D.** Boxplots of the computational time for simulating independent and correlated genes respectively, across 10 datasets. The y-axis represents the computational time, measured in seconds and displayed on a log10 scale. We specified the marginal distribution as negative binomial to improve the speed in scDesign2.

test, the smaller the test statistics are, the more similar the simulated data are to the original data. We used two datasets in our comparisons: the prefrontal cortex of seven subjects with a Braak Stage of 0 from Mathys et al. [32] (ROSMAP) and the prefrontal cortex of nine healthy controls from Lau et al. [33] (PNAS). We considered five cell types for each dataset (i.e., 10 datasets in total) and reduced cell types containing over 10,000 cells to 5,000 cells to save computation time. The data details can be found in Table **??**.

All methods for simulating independent data corresponding to real data lead to independent genes (Figure **??** and Table **??**). As shown in Figures 2A and 2B, scSimu, scDesign3, and SPARSim achieve the greatest similarity with real data for both tasks. This is in line with the observation of Song et al. [22] that scDesign3 had better performance and that of Cao et al. [21] where SPARSim was the second best method. However, ZINB-WaVE, the best method in two benchmark papers [17, 21], is not among the top three methods in our comparisons. As mentioned before, scSimu is designed for the Drop-seq data. Therefore, we only considered Drop-seq data for comparisons The model assumption for scRNA-seq data differs across sequencing platforms. For example, a zero-inflated model is used for the Smart-seq data [34]. In the two benchmark papers [17, 21], different types of scRNA-seq data were considered, likely contributing to conclusions that were inconsistent with ours. We also display the results for the KDE test and the KS test separately in Figure **??**, where scSimu and scDesign3 are always the top two methods. Although scSimu and scDesign3 have very similar performances, the extensive computational time hinders us from applying scDesign3 to simulate null distributions (Figures 2C and 2D).

Based on the above results, scSimu can be used to generate empirical p-values to test whether two genes are correlated as it can efficiently produce count data with independent genes that closely resemble the real data. To evaluate the ability of empirical p-values from scSimu in terms of controlling false positives, we considered the empirical p-value distribution of independent gene pairs in a real data simulation. As shown in Figure 3, the p-value distribution almost aligns with Unif(0,1), indicating effective control over type I error across the seven estimation methods we considered.

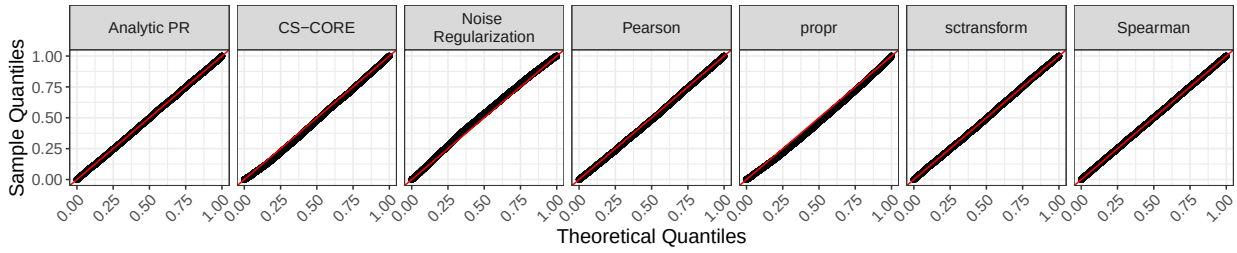

Fig. 3 Q-Q plots of empirical p-values for independent gene pairs. Empirical p-values were calculated according to the methods outlined in Section 5.3.

2.3 A lack of type I error control in method evaluation can lead to misleading results

In this section, we demonstrate the critical need for using empirical p-values to control false positives in the evaluation and comparison of different gene co-expression network estimation methods. Because we do not know the ground truth of correlated gene pairs in real data, we used simulations that mimicked real data to better understand the impact of appropriate control of false positives. For the evaluation measures, we focused on reproducibility and overlap with known biological networks. Reproducibility was assessed by counting the gene pairs identified as correlated across independent datasets. We used STRING and Reactome as prior knowledge of biological networks. All the simulated data were generated through scSimu.

An important step in the evaluation of the estimated gene co-expression networks is to decide which gene pairs are correlated. As we mentioned previously, correlation-strength-based and p-value-based are two common approaches for inferring correlated gene pairs. In the correlation-strength-based approach, selecting a larger number of top pairs leads to higher reproducibility and more overlaps with biological networks (Figures 4A and 5A). However, it may include more false positives. As shown in Figures 4B and 5B, an increasing number of reproducible pairs and a higher number of overlaps with STRING are both associated with a lower precision. We also observe higher reproducibility and more overlaps when the p-value-based approach with a less stringent cut-off is used (Figures 4E and 5E), but there is a less decline in precision (Figures 4F and 5F).

To understand the influence of these false positives in the evaluation process, we investigated the proportion of misidentified reproducible pairs (see Section 5.4). As expected, there is an increased proportion of misidentification with a growing number of gene pairs selected, especially for certain gene co-expression estimation methods (Figure 4D). This finding indicates that some gene co-expression estimation methods, when using a correlation-strength-based approach, may seem artificially more effective due to not accounting for false positives. We estimated empirical p-values through scSimu and evaluated the p-value-based approach. We find that the proportion of misidentified reproducible pairs is much lower than that based on the correlation-strength-based approach (Figure 4H). The results are similar to those comparing overlaps with STRING (Figure 5) and overlaps with Reactome (Figure **??**). Even though the correlation-strength-based approach has a much higher number of reproducible pairs and overlaps with known biological networks, the large discrepancy in precision and proportion of misidentification at different thresholds necessitates more

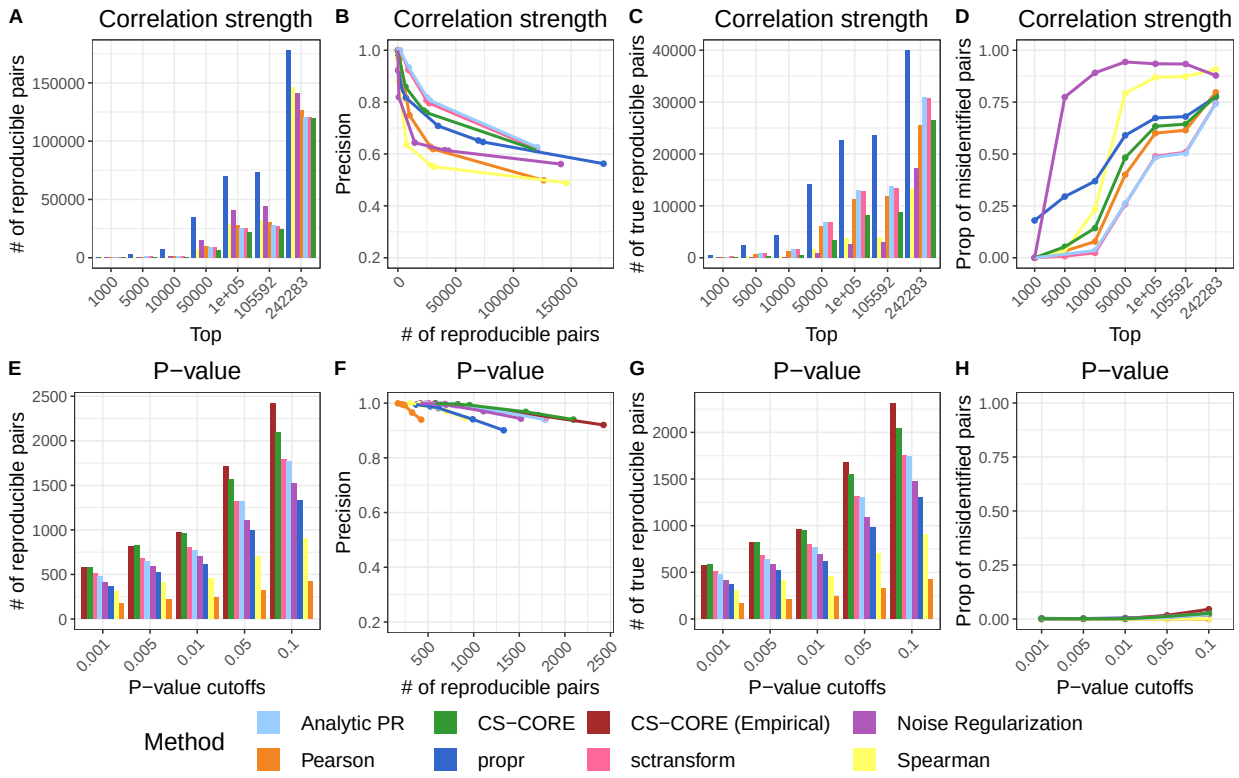

Fig. 4 Performance in terms of reproducibility. Different estimation methods are represented with different colors. For CS-CORE, we considered two different ways of calculating p-values, either through the method described in the original paper (denoted as CS-CORE) or through scSimu (denoted as CS-CORE (Empirical)). Correlated gene pairs were identified via the correlation-strength-based approach (A-D), and p-value-based approach (E-H). **A and E.** The number of identified reproducible pairs at different cutoffs. **B and F.** Relationship between precision in ROSMAP data (precision in PNAS data can be found in Figure **??**) and the number of reproducible pairs. Points in B and F correspond to different cutoffs for the correlation strength and p-value, respectively. **C and G.** The number of true reproducible pairs at different cutoffs. **D and H.** The proportion of misidentified reproducible pairs when gene pairs were selected by their estimated correlations (D) or by the empirical p-values (H).

careful threshold selection. Additionally, variations in the density and magnitude of the correlation matrix across datasets further complicate threshold selection, making applying the same threshold to different datasets questionable.

However, there is a potential trade-off between the proportion of false identification and the number of true reproducible pairs or true overlaps with known biological networks, i.e., a lower proportion of false identification may correspond to a lower number of true reproducible pairs or true overlaps. To better quantify this, we compare the number of true reproducible pairs identified by the correlation-strength-based approach and the p-value-based approach, ensuring that their proportions of false identifications are the same in Figure 6, which further supports that selecting correlated gene pairs based on p-value is better than that based on correlation strength. Similarly, we observe a higher number of true overlaps with known biological networks when using the p-value-based approach (Figures **??** and **??**).

Without considering the p-value in selecting correlated gene pairs can also affect the benchmark results across gene co-expression estimation methods. The ranks of estimation methods at the same correlation strength cutoff vary substantially in the number of reproducible pairs and the number of true reproducible pairs when we use the correlation-strength-based approach (Figures 4A and 4C), i.e., a method that exhibits the highest reproducibility may not be the best method since many of the reproducible pairs may be false

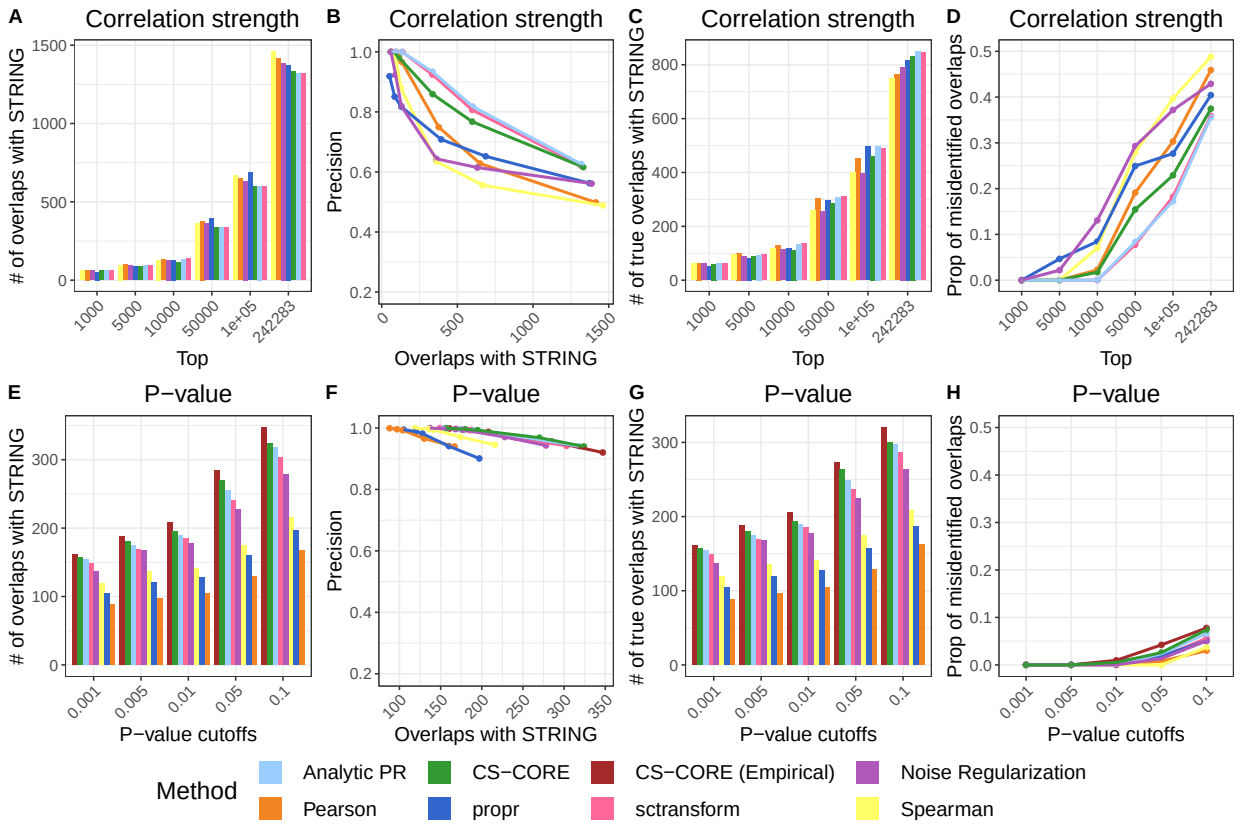

Fig. 5 Performance in terms of overlaps with STRING. A and E. The number of overlaps with STRING at different cutoffs. **B and F.** Relationship between precision and the number of overlaps with STRING. **C and G.** The number of true overlaps with STRING at different cutoffs. **D and H.** The proportion of misidentified overlaps when gene pairs were selected by their estimated correlations (D) or by the empirical p-values (H).

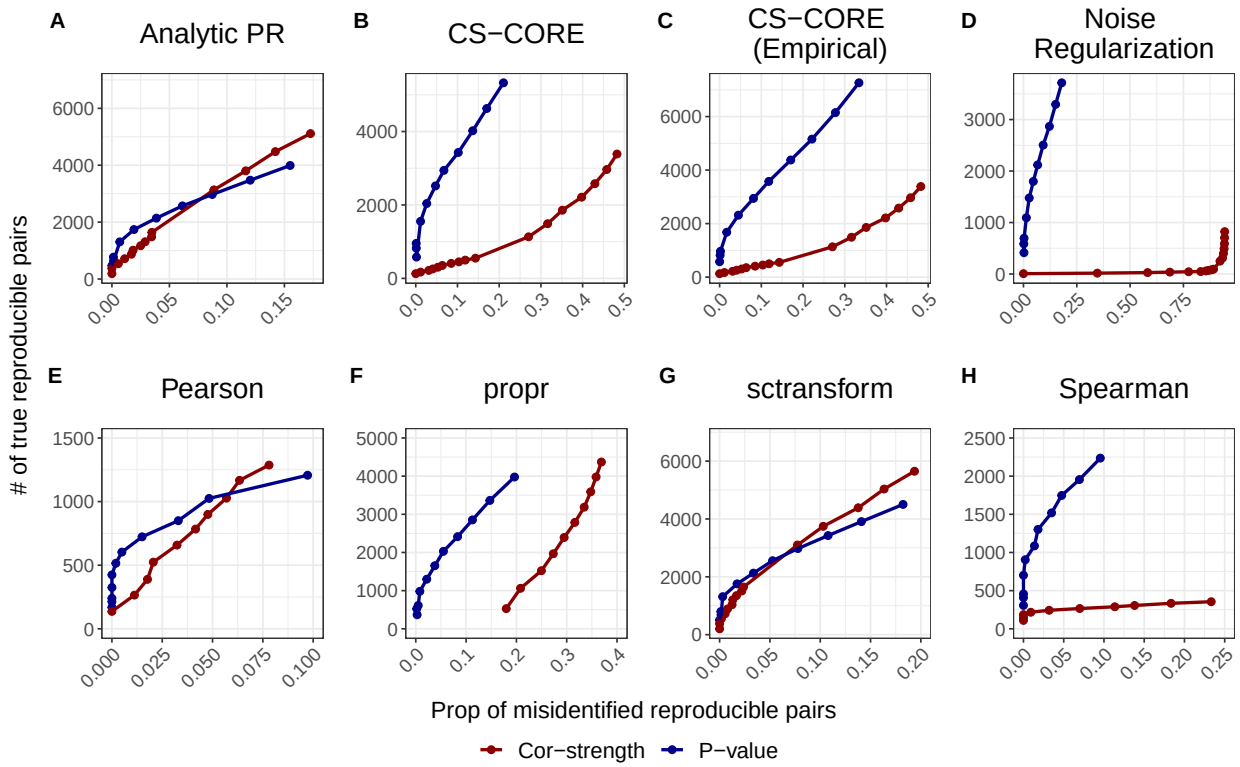

Fig. 6 Comparison of correlated gene selection approaches across different estimation methods in terms of reproducibility. The different selection approaches are distinguished by different colors.

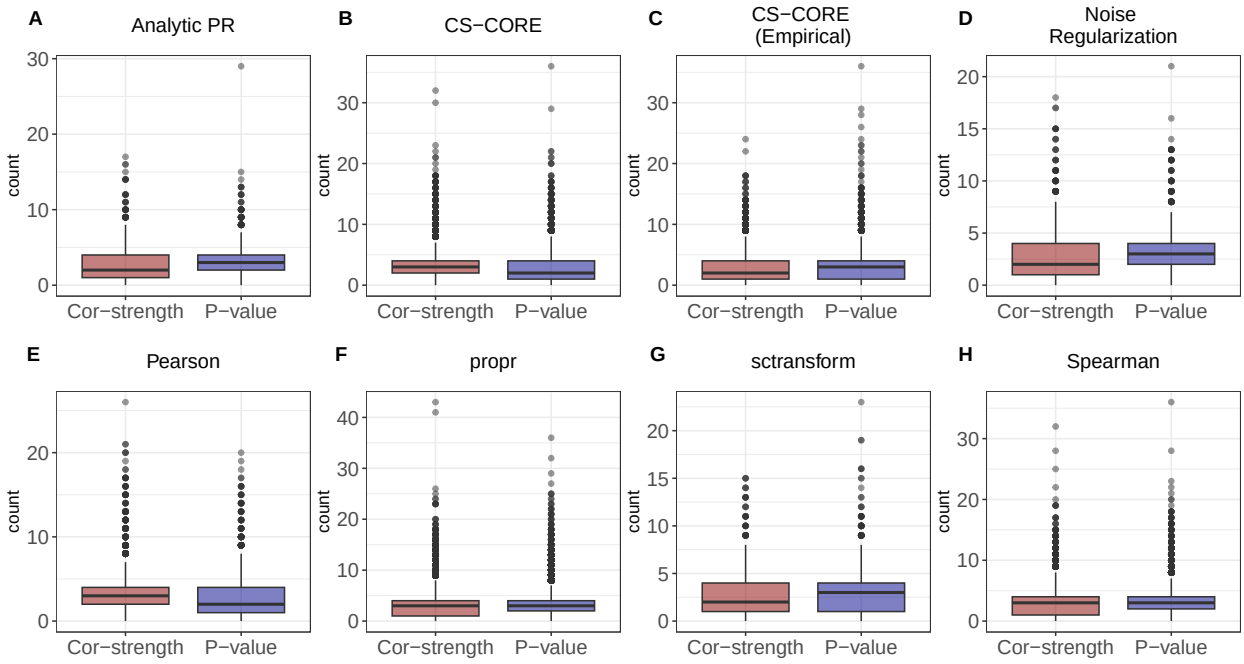

Fig. 7 Boxplots of the number of shared GO terms in excitatory neurons. Each point represents one gene pair. The y-axis is the number of shared GO terms. The threshold for the p-value-based approach was 0.05. For the correlation-strength-based approach, the threshold was set to match the number of correlated gene pairs identified by the p-value-based approach.

positives. On the contrary, the ranks are rather consistent when we use the p-value-based approach (Figures 4E and 4G). We observe the same pattern through overlaps with known biological networks (Figures 5 and **??**). Our observation also explains the inconsistent results between the benchmark study of gene association estimation methods conducted by Skinnider et al. [1] which found propr to be superior, and the study conducted by Su et al. [11] where propr did not perform well, due to the lack of controls of false positives. Our study highlights the importance of considering statistical significance in gene pair selection. By selecting correlated gene pairs based on the p-value, as we demonstrated above, more reliable evaluation results can be achieved.

2.4 P-value-based approach identifies more reliable correlated gene pairs from real data

We further compared the two approaches for selecting correlated gene pairs using real data, focusing on oligodendrocytes and excitatory neurons in the ROSMAP data. To perform the comparison, we calculated the number of shared GO terms for gene pairs identified exclusively by either the p-value-based approach or the correlation-strength-based approach. A higher number of shared GO terms indicates a stronger likelihood that two genes are correlated. Overall, gene pairs uniquely identified by the p-value-based approach exhibit a higher number of shared GO terms, suggesting that the p-value-based approach identifies more reliable correlated gene pairs (Figures 7 and **??**).

2.5 Consideration of the overlap by chance and bias in the biological networks in the evaluation process

The p-value-based approach will cause a large discrepancy in the number of identified correlated gene pairs between different estimation methods. In this case, a method with more reproducible pairs or overlaps may also have more reproducible pairs or overlaps by chance, i.e., the counts if we randomly select the same number of gene pairs with similar expression distribution as correlated. Therefore, it is advisable to consider both the observed value and the influence of randomness on reproducibility and overlap during the evaluation process.

The randomness in reproducibility can be quantified using a weighted random sampling process. To determine the sampling probability, we categorize gene pairs into expression bins based on their mean expression levels. The sampling probability for each bin is then estimated by the proportion of identified correlated gene pairs within this expression bin. To reduce the computational burden, the average number of random reproducible pairs can be estimated by $c_1 * \frac{\sum_{i \in \{\text{Gene pairs: identified as correlated in the second dataset}\}} p_i}{\sum_{j \in \text{All gene pairs}} p_j}$, where c_1 is the number of identified correlated gene pairs in the first dataset and p_i is the probability of gene pair i being sampled. Similarly, the average number of random overlaps with known biological networks can be estimated by $c * \frac{\sum_{i \in \text{Biological network}} p_i}{\sum_{j \in \text{All gene pairs}} p_j}$, where c is the number of identified correlated gene pairs. Additional attention is needed for the number of random overlaps because the known biological network represents only a small subset of all truly correlated gene pairs.

The above procedure can also reduce the influence of expression bias in known biological networks. We observe that the known biological networks are biased toward genes and gene pairs with certain expression levels (Figures 8A and 8B). To illustrate this, we calculated the average expression levels of the genes based on the ROSMAP data. We contrast the expression levels of genes within biological networks versus the other genes and find that genes in biological networks are biased toward a higher expression level (Figure 8A). We further consider pair-level bias. We established the background expression level for gene pairs by calculating the geometric mean of the expression levels for every possible combination, using the genes present in the biological network. As shown in Figure 8B, the genes in the biological network, which are already biased in expression levels, exhibit an additional tendency toward higher expression at the pair level. Although there is a general tendency for genes and gene pairs in known biological networks to have higher expression levels, the degree of such bias varies across datasets. For example, for the ROSMAP dataset, there is a clear bias for the 100 genes with the highest expression levels as well as the 100 genes with the most variation. However, the bias is negligible for the top 1,000 or 5,000 genes (Figures 8C-H) and there is an expression bias across a broader range of genes for a lung dataset [35] (Figure **??**). We also found such expression bias in the Reactome database (Figures **??** and **??**).

Together with expression bias in the estimated correlation matrix, expression bias in biological networks can cause spurious higher overlaps. Therefore, we suggest checking for this bias before estimating the overlap with known biological networks. Even though the weighted random sampling process can mitigate the

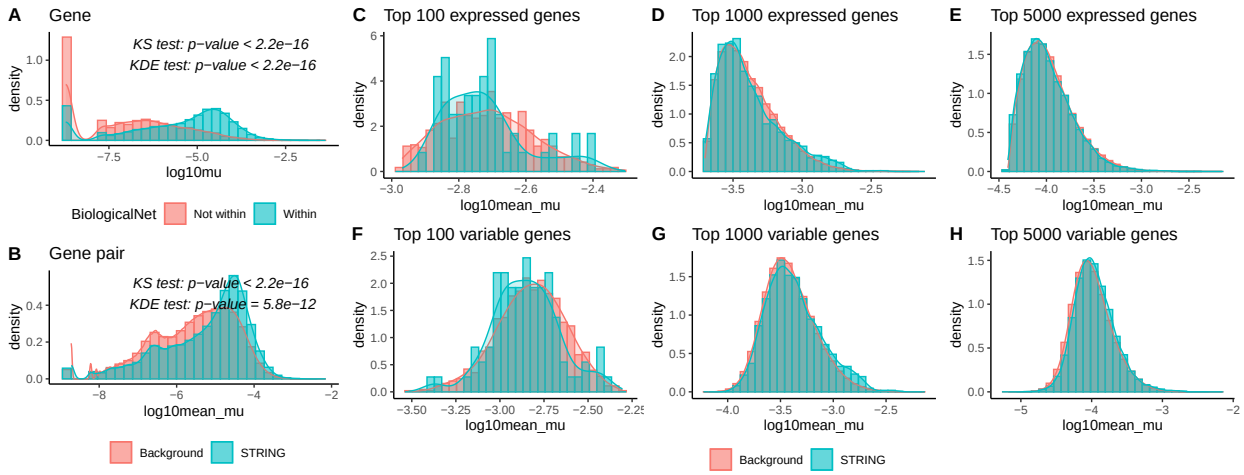

Fig. 8 Expression bias in STRING. A. Gene level bias. The color represents whether the gene is in the biological network. The KS test and KDE test were conducted to assess the similarity between the two distributions. **B.** Gene pair level bias. **C-H.** The biological network here is the sub-network of the original network obtained by filtering out gene pairs containing genes other than the top k genes. The background was generated based on this sub-network.

influence of such bias, we still recommend considering others, such as reproducibility, in the evaluation as it is unknown whether this bias is an artifact or represents real biology. Our results in the previous section are not significantly affected by this bias as there is a negligible effect for the top 1000 or 5000 genes in the ROSMAP data.

3 Discussion

In this manuscript, we discussed and addressed the issue of false positive controls in the current evaluation pipeline for inferring gene co-expression networks from single-cell data. We developed scSimu for simulating scRNA-seq data to generate empirical p-values. We benchmarked scSimu against other simulation methods and demonstrated that scSimu is capable of efficiently and accurately mimicking Drop-seq based scRNA-seq data. It also has some other desired properties. For example, scSimu offers parameter flexibility, allowing users to apply estimates from any method to the model. The simulation model accommodates discrete covariates (e.g., cell types), enabling the generation of data for each distinct category which can then be concatenated. It also preserves all gene names and does not eliminate any genes or cells from the analysis which is critical for accurate comparisons across different downstream analyses. This contrasts with methods such as SymSim, ZINB WaVE, powsimR, and muscat, which remove certain genes.

For each correlation estimation method, we considered the correlation-strength-based approach and p-value-based approach for identifying correlated gene pairs. We demonstrated the importance of using p-values to control false positives in selecting correlated gene pairs through both simulations and real data analyses. Failure to consider p-values will lead to a high proportion of false identification, and incorrectly rank the performance of different gene co-expression estimation methods. Furthermore, we discussed the necessity of considering random overlaps and showed the potential expression bias in known biological networks.

Although we provide feasible solutions to address issues in the evaluation process of gene co-expression networks, our study has several limitations. First, scSimu is restricted to simulating Drop-seq scRNA-seq data since its model assumption is negative binomial. Second, employing empirical p-values is computationally more extensive, making it impractical for gene co-expression estimation methods that already have extensive running times, for example, baredSC [36] and locCSN [9]. Third, empirical p-values cannot be used with quantile-normalization-based methods, like SpQN [6], as quantile normalization alters the data distribution which makes it incomparable between observed data and simulated independent data. Finally, we have not fully addressed the issue of expression bias in known biological networks. Further research is needed to most effectively infer gene co-expression networks.

4 Conclusions

Our study demonstrates that p-value-based approach generates more reliable results than correlation-strength-based approach. Therefore, to provide a fairer comparison across different gene co-expression network estimation methods, we suggest that researchers adopt the following practices in their evaluation: (1) estimate pairwise empirical p-values using scSimu, (2) use a p-value-based approach to select correlated gene pairs, (3) evaluate the expression bias in the known biological networks and combine with other evaluation methods if there is a bias, and (4) consider randomness if the number of identified correlated gene pairs varies substantially across methods. We believe that adopting such a unified approach can effectively reduce discrepancies across different studies.

5 Methods

5.1 scSimu

Overview of the model. For cell i and gene j, scSimu simulates the observed count, x_{ij}, from the following distributions,

$$\begin{aligned} z_{ij} &\sim Gamma(\alpha_j, \mu_j), \\ x_{ij}|z_{ij} &\sim Poisson(s_i z_{ij}), \end{aligned} \tag{2}$$

where z_{ij} is the corresponding relative underlying expression, α_j is the dispersion parameter, μ_j is the mean expression, and s_i is the sequencing depth. The correlation structure between genes is modeled by the Gaussian copula which can introduce the dependence among genes while preserving the marginal distributions [11, 19, 22, 28].

Choices of model parameters. The input for the simulation framework can be a count matrix or user-defined parameters. For count matrix input, we use the observed sequencing depths (s_i), estimate the marginal parameters for genes (α_j and μ_j) based on the Gamma-Poisson generalized linear model [37], and calculate the correlation matrix for genes that are highly expressed, highly variable, or of particular interest

by CS-CORE. Alternatively, users may opt to directly provide marginal parameters, sequencing depths, and correlation matrix based on their preferences.

Detailed procedures. For simulation, we mimic the sequencing procedure commonly adopted. We first generate a gene expression matrix under the Gamma distribution. If the correlation matrix is not an identity matrix, i.e., genes are correlated, the correlation structure inferred from real data or specified by the user will be imposed through the Gaussian copula. Compared with the simulation method used in our previous publication [11], here we updated the Gaussian copula process to allow for the case that the correlation matrix is not positive semi-definite given that some gene co-expression estimation methods sacrifice positive semi-definite to accommodate the noise and sparsity in scRNA-seq (e.g., CS-CORE, SpQN [6]). Additionally, after applying a threshold based on significance or correlation strength, a positive semi-definite correlation matrix may no longer remain positive semi-definite. Finally, the gene count matrix is simulated from the underlying gene expression matrix using the Poisson distribution with the sequencing depths from the input real data or the user.

5.2 Benchmark simulation methods

Evaluation metrics. To evaluate the performance of different simulation methods, we compared the simulated data with real data from eight data summaries:

- **Mean(logCPM)**: mean expression of genes estimated via log-normalized data.
- **Var(logCPM)**: variance of genes estimated using log-normalized data.
- **Coefficient of variation**: $\sqrt{Var(logCPM)}/Mean(logCPM)$.
- **Zero fraction (cell)**: fraction of cells with zero counts.
- **Zero fraction (gene)**: fraction of genes with zero counts.
- **Library size**: total gene counts per cell.
- **Cell correlation**: Spearman correlation between cells based on log-normalized data. The number of cells used to estimate cell correlation was capped by 500.
- **Gene correlation**: correlations estimated by CS-CORE for the top 1,000 highly expressed genes determined by Mean(logCPM).

The KS test statistics were calculated via the function `ks.test` in the R package stats version 4.3.0. The KDE test statistics were calculated by `kde.test` function in the R package ks version 1.14.1. For metrics exceeding 10,000 points (e.g., gene correlation), we only used 10,000 of them to calculate the KDE test statistics to reduce the computation time. To make the results comparable across datasets and metrics, the KDE test statistics and KS test statistics were scaled separately within each dataset and each data metric across different simulation methods using min-max normalization. We combined the KS test statistics and KDE statistics by averaging them as these two tests complement each other. For every simulation method, we calculated the average across datasets within each data metric, as displayed in Figure 2. The

overall performance of the simulation method was evaluated based on the mean across eight metrics. Gene correlation under the independent task was further evaluated by the heatmap and density for the correlation matrix. The computational time was assessed via the Intel Xeon Platinum 8358 and Intel Xeon Gold 6240 processors.

Implementation of simulation methods. We considered 13 simulation methods in addition to scSimu, of which 11 were applied to the correlated task and nine to the independent task. The descriptions of these simulation methods are listed below.

- **ESCO**: Independent genes were simulated using functions from R package ESCO version 0.99.12 by specifying `withcorr=F` and `type = "single"`.
- **hierarchicell**: Conducted using R package hierarchical version 1.0.0.
- **muscat**: Conducted through R package muscat version 1.14.0.
- **Permutation**: Followed procedures in Su et al. [11].
- **POWSC**: Simulated using R package POWSC version 0.99.12.
- **powsimR**: It simulated data containing both case and control cells based R package powsimR version 1.2.5 as originally designed for differential expression analysis.
- **scDesign2**: Simulation was conducted using R package scDesign2 version 0.1.0. Set `marginal="nb"` to reduce computational time, while default settings were retained for all other parameters. Independent genes were simulated by specifying `sim_method = "ind"`.
- **scDesign3**: Correlated genes were simulated by setting `corr_formula = "1"` and independent genes were simulated by setting `corr_formula = "ind"` utilizing R package scDesign3 version 1.0.1.
- **SCRIP**: Simulation were conducted by using R package SCRIP version 1.0.0 .
- **SPARSim**: Simulated using R package sparsim version 0.9.5.
- **Splat**: Conducted by using R package splatter version 1.26.0.
- **SymSim**: Performed using the R package SymSim version 0.0.0.9000.
- **ZINB-WaVE**: We utilized the function related to ZINB-WAVE in the splatter package to simulate data.

5.3 Empirical p-value estimation

The empirical p-values were estimated in three steps. (1) Generate k independent datasets corresponding to the real data using scSimu and estimate the correlations. For each gene pair, gather k estimates from these independent datasets alongside one from the real data. (2) Estimate the mean μ_0 and variance σ_0^2 of these k estimates for each gene pair. (3) The empirical p-value is calculated using a normal distribution with the estimated mean and variance, where the p-value $= P(|X| > |Estimate_{obv}|)$ and $X \sim N(\mu_0, \sigma_0^2)$.

Type I errors. To demonstrate that all gene co-expression estimation methods can achieve well-controlled type I errors using such empirical p-values, we first used scSimu to mimic oligodendrocytes in the PNAS data. We calculated the empirical p-values of estimated correlations for the top 1,000 highly expressed genes based on the procedure above with $k = 500$. We then compared the empirical p-values of independent gene pairs with Unif(0,1) in Figure 3.

5.4 Evaluation of estimated gene co-expression networks

Estimation of gene co-expression networks. We estimated the co-expression networks for the top 1,000 genes with the highest expression levels shared between the ROSMAP and PNAS datasets using seven methods. The Analytic PR inferred the gene co-expression network using Pearson correlation on the residuals obtained through the procedures in Lause et al. [14]. CS-CORE was implemented through functions in R package CS-CORE version 0.0.0.9000. Noise Regularization was based on the scrips in https://github.com/RuoyuZhang/NoiseRegularization [38]. Pearson and Spearman estimated the gene co-expression networks by the Pearson and Spearman correlation on log-normalized data, respectively. The R package propr version 4.2.6 was used to generate the gene co-expression networks for propr. For sctransform, we used R package Seurat version 5.0.1 to derive the residuals and then applied Pearson correlation to them.

Approaches for selecting correlated gene pairs. We evaluated and compared the correlation-strength-based approach and p-value-based approach for selecting correlated gene pairs.

- **Correlation-strength-based approach.** We defined correlation strength as the absolute value of the estimated correlation. The correlated gene pairs are the top k gene pairs with the largest correlation strength. The k was varied from 1,000, 5,000, 10,000, 50,000, to 100,000. For reproducibility, we included the number of true correlated gene pairs for the ROSMAP simulated data (135,244) and PNAS simulated data (59,933) as our cutoffs. For the evaluation of overlaps with known biological networks, we included the number of true correlated gene pairs for the ROSMAP simulated data (135,244) [3].
- **P-value-based approach.** The p-values were adjusted according to the Benjamini-Hochberg (BH) procedure. The correlated gene pairs are those pairs with adjusted p-values less than the threshold. The thresholds for the p-values considered here are 0.001, 0.005, 0.01, 0.05, and 0.1.

Precision. Precision is the ratio of true positives to inferred positives. For reproducibility, the precision values in the main text were based on the ROSMAP data, while those from the PNAS data were displayed in Figure **??**.

Proportion of misidentification. Let $A = \{$gene pairs: identified as correlated from the first dataset$\}$, $A_T = \{$gene pairs: true positives from the first dataset$\}$, $B = \{$gene pairs: identified as correlated from the second dataset$\}$, and $B_T = \{$gene pairs: true positives from the second dataset$\}$. Then, the total reproducible

pairs can be represented by $A \cap B$, while the true reproducible pairs are denoted as $A_T \cap B_T$. Therefore, the proportion of misidentified reproducible pairs $= \frac{\#\text{ total reproducible pairs} - \#\text{ true reproducible pairs}}{\#\text{ total reproducible pairs}}$. For overlaps with known biological networks, define C as the set of gene pairs in the biological network. Similarly, the total overlaps is $A \cap C$ and true overlaps is $A_T \cap C$. Then, the proportion of misidentified overlaps is $\frac{\#\text{ total overlaps} - \#\text{ true overlaps}}{\#\text{ total overlaps}}$.

Reproducibility. We constructed simulated data based on 17,693 oligodendrocytes in the PNAS data, which are also from the prefrontal cortex, as independent validation data. The reproducibility was evaluated based on the number of reproducible pairs which are gene pairs identified as correlated in both datasets.

Biological networks. The human protein-protein interaction network was obtained from STRING v12 [39] with combining score larger than 500. The other biological network we considered in our study was the Reactome [40], a pathway database. By combining each pair within the same pathway, we constructed an interaction network [1].

GO terms. The genes' GO terms were obtained from the R package biomaRt version 2.58.0.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data and materials

The ROSMAP dataset (syn52293417) has controlled access under the regulation of human privacy. The PNAS dataset used in our analysis is publicly available (GSE157827). The lung data we used to evaluate the expression bias in biological networks can be accessed through GSE136831. The relevant code of this study can be found in https://github.com/xs222/coexp_validation [41]. The R package scSimu is available at https://github.com/xs222/scSimu [42].

Competing interests

The authors declare that they have no competing interests.

Funding

This work was supported in part by NIH grants R01 GM134005 and U24 HG012108.

Authors' contributions

XS conducted the study. HZ supervised the project. All authors read and approved the final manuscript.

Acknowledgements

The authors would like to thank the Religious Orders Study and Memory and Aging Project (ROSMAP) Study, Dr. Shun-Fat Lau, Dr. Nancy Y. Ip, Taylor S. Adams, and Dr. Naftali Kaminski for sharing the datasets.

References

[1] Skinnider, M.A., Squair, J.W., Foster, L.J.: Evaluating measures of association for single-cell transcriptomics. Nature methods **16**(5), 381–386 (2019)

[2] Chen, S., Mar, J.C.: Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data. BMC bioinformatics **19**(1), 1–21 (2018)

[3] Pratapa, A., Jalihal, A.P., Law, J.N., Bharadwaj, A., Murali, T.: Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. Nature methods **17**(2), 147–154 (2020)

[4] Li, W.V., Li, Y.: sclink: Inferring sparse gene co-expression networks from single-cell expression data. Genomics, proteomics & bioinformatics **19**(3), 475–492 (2021)

[5] Zhang, R., Atwal, G.S., Lim, W.K.: Noise regularization removes correlation artifacts in single-cell rna-seq data preprocessing. Patterns **2**(3) (2021)

[6] Wang, Y., Hicks, S.C., Hansen, K.D.: Addressing the mean-correlation relationship in co-expression analysis. PLoS computational biology **18**(3), 1009954 (2022)

[7] Li, S., Schmid, K.T., Vries, D.H., Korshevniuk, M., Losert, C., Oelen, R., Blokland, I.V., Groot, H.E., Swertz, M.A., Harst, P., *et al.*: Identification of genetic variants that impact gene co-expression relationships using large-scale single-cell data. Genome Biology **24**(1), 1–37 (2023)

[8] Crow, M., Paul, A., Ballouz, S., Huang, Z.J., Gillis, J.: Exploiting single-cell expression to characterize co-expression replicability. Genome biology **17**, 1–19 (2016)

[9] Wang, X., Choi, D., Roeder, K.: Constructing local cell-specific networks from single-cell data. Proceedings of the National Academy of Sciences **118**(51), 2113178118 (2021)

[10] Wang, L.: Single-cell normalization and association testing unifying crispr screen and gene co-expression analyses with normalisr. Nature communications **12**(1), 6395 (2021)

[11] Su, C., Xu, Z., Shan, X., Cai, B., Zhao, H., Zhang, J.: Cell-type-specific co-expression inference from single cell rna-sequencing data. Nature Communications **14**(1), 4846 (2023)

[12] Iacono, G., Massoni-Badosa, R., Heyn, H.: Single-cell transcriptomics unveils gene regulatory network plasticity. Genome biology **20**, 1–20 (2019)

[13] Lu, S., Keleş, S.: Debiased personalized gene coexpression networks for population-scale scrna-seq data. Genome Research **33**(6), 932–947 (2023)

[14] Lause, J., Berens, P., Kobak, D.: Analytic pearson residuals for normalization of single-cell rna-seq umi data. Genome biology **22**(1), 1–20 (2021)

[15] Hafemeister, C., Satija, R.: Normalization and variance stabilization of single-cell rna-seq data using regularized negative binomial regression. Genome biology **20**(1), 296 (2019)

[16] Specht, A.T., Li, J.: Leap: constructing gene co-expression networks for single-cell rna-sequencing data using pseudotime ordering. Bioinformatics **33**(5), 764–766 (2017)

[17] Crowell, H.L., Morillo Leonardo, S.X., Soneson, C., Robinson, M.D.: The shaky foundations of simulating single-cell rna sequencing data. Genome Biology **24**(1), 1–19 (2023)

[18] 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)

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

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

[21] 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)

[22] Song, D., Wang, Q., Yan, G., Liu, T., Sun, T., Li, J.J.: scdesign3 generates realistic in silico data for multimodal single-cell and spatial omics. Nature Biotechnology, 1–6 (2023)

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

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

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

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

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

[28] Tian, J., Wang, J., Roeder, K.: Esco: single cell expression simulation incorporating gene co-expression. Bioinformatics **37**(16), 2374–2381 (2021)

[29] Zimmerman, K.D., Langefeld, C.D.: Hierarchicell: an r-package for estimating power for tests of differential expression with single-cell data. BMC genomics **22**(1), 1–8 (2021)

[30] Crowell, H.L., Soneson, C., Germain, P.-L., Calini, D., Collin, L., Raposo, C., Malhotra, D., Robinson, M.D.: Muscat detects subpopulation-specific state transitions from multi-sample multi-condition single-cell transcriptomics data. Nature communications **11**(1), 6077 (2020)

[31] Qin, F., Luo, X., Xiao, F., Cai, G.: Scrip: an accurate simulator for single-cell rna sequencing data. Bioinformatics **38**(5), 1304–1311 (2022)

[32] Mathys, H., Peng, Z., Boix, C.A., Victor, M.B., Leary, N., Babu, S., Abdelhady, G., Jiang, X., Ng, A.P., Ghafari, K., *et al.*: Single-cell atlas reveals correlates of high cognitive function, dementia, and resilience to alzheimer's disease pathology. Cell **186**(20), 4365–4385 (2023)

[33] Lau, S.-F., Cao, H., Fu, A.K., Ip, N.Y.: Single-nucleus transcriptome analysis reveals dysregulation of angiogenic endothelial cells and neuroprotective glia in alzheimer's disease. Proceedings of the National Academy of Sciences **117**(41), 25800–25809 (2020)

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

[35] Adams, T.S., Schupp, J.C., Poli, S., Ayaub, E.A., Neumark, N., Ahangari, F., Chu, S.G., Raby, B.A., DeIuliis, G., Januszyk, M., *et al.*: Single-cell rna-seq reveals ectopic and aberrant lung-resident cell populations in idiopathic pulmonary fibrosis. Science advances **6**(28), 1983 (2020)

[36] Lopez-Delisle, L., Delisle, J.-B.: baredsc: Bayesian approach to retrieve expression distribution of single-cell data. BMC bioinformatics **23**(1), 36 (2022)

[37] Townes, F.W., Hicks, S.C., Aryee, M.J., Irizarry, R.A.: Feature selection and dimension reduction for single-cell rna-seq based on a multinomial model. Genome biology **20**, 1–16 (2019)

[38] Zhang, R., Atwal, G.S., Lim, W.K.: NoiseRegularization: A tool for noise regularization in machine learning. Accessed: 2024-08-02 (2024). https://github.com/RuoyuZhang/NoiseRegularization

[39] STRING: functional protein association networks. Accessed: 2024-08-02. https://string-db.org/

[40] Reactome Pathway Database. Accessed: 2024-08-02. https://reactome.org/

[41] Shan, X., Zhao, H.: Code for coexpression validation. Accessed: 2024-08-02 (2024). https://github.com/xs222/coexp_validation

[42] Shan, X., Zhao, H.: scSimu: A simulation tool for single-cell RNA sequencing data. Accessed: 2024-08-02 (2024). https://github.com/xs222/scSimu