communications biology

A Nature Portfolio journal

Article

https://doi.org/10.1038/s42003-025-07947-7

Reference-informed evaluation of batch correction for single-cell omics data with overcorrection awareness

Check for updates

Xiaoyue Hu[1,2,8], He Li[1,8], Ming Chen[3], Junbin Qian[4,5,6,7,9] ✉ & Hangjin Jiang[1,9] ✉

Batch effect correction (BEC) is fundamental to integrate multiple single-cell RNA sequencing datasets, and its success is critical to empower in-depth interrogation for biological insights. However, no simple metric is available to evaluate BEC performance with sensitivity to data overcorrection, which erases true biological variations and leads to false biological discoveries. Here, we propose RBET, a reference-informed statistical framework for evaluating the success of BEC. Using extensive simulations and six real data examples including scRNA-seq and scATAC-seq datasets with different numbers of batches, batch effect sizes and numbers of cell types, we demonstrate that RBET evaluates the performance of BEC methods more fairly with biologically meaningful insights from data, while other methods may lead to false results. Moreover, RBET is computationally efficient, sensitive to overcorrection and robust to large batch effect sizes. Thus, RBET provides a robust guideline on selecting case-specific BEC method, and the concept of RBET is extendable to other modalities.

Single-cell sequencing provides an unprecedented opportunity to define molecular cell type and cell state in a data-driven fashion, and has achieved great accomplishment in exploring the mechanism of different diseases at the cell level[1–3]. Researchers may take the advantage to investigate in-depth biological insights by integrating massive datasets, either publically available or generated in-house, for multiple downstream analyses, such as cell type and state annotation, differential gene expression, trajectory inference, cell-cell communication, etc. But the underlying difficulty is the large variations or batch effect in these datasets, which are collected from different labs, experiments, capturing times, handling personnel, and even technology platforms[4,5]. These differences confound the true biological variations during data integration, which need to be removed by batch effect correction (BEC) tools to ensure biologically meaningful results[6–8]. Indeed, various BEC methods have been proposed in recent years, and a recent benchmark study compared 14 BEC methods in terms of batch mixing, accuracy of cell type annotation, computational efficiency for large datasets, etc., and recommended Harmony, LIGER and Seurat as top performers[9].

However, these benchmark studies lack comparisons on key metrics that could impact routinely performed downstream analyses such as trajectory inference and cell-cell communication. First, the unwanted technical variations between samples from different batches should be effectively removed after BEC, both locally and globally. However, existing BEC evaluation tools such as kBET[10] or LISI[7], may have a lower power in cases where only parts of cells have batch effect. Second, these tools lack algorithms to estimate overcorrection, that is, the degradation of true biological information while eliminating technical bias[10]. Overcorrection makes downstream analyses highly problematic and often leads to wrong conclusion. Third, BEC methods should be evaluated for capability to cope with large batch effect sizes. Previous benchmark studies ultilized datasets with limited batch numbers (2–5) and arguably not sufficient batch effect sizes, while these are possibly several order of magnitude higher in real practice, given the overwhelming availability of scRNA-seq datasets[11]. Thus, a comprehensive and reliable test metric taking full consideration of the aforementioned scenarios is an unmet need for BEC applications.

In this paper, we propose a novel statistical method, Reference-informed Batch Effect Testing (RBET) for evaluating the performance of BEC tools. This method takes advantage of reference gene (RG) expression/variation pattern and maximum adjusted chi-squared (MAC) statistics for two-sample distribution comparison (Fig. 1A). In both simulated and real data, RBET demonstrates a superior performance in detecting batch effects,

[1]Center for Data Science, Zhejiang University, Hangzhou, China. [2]School of Mathematical Sciences, Zhejiang University, Hangzhou, China. [3]College of Life Sciences, Zhejiang University, Hangzhou, China. [4]Zhejiang Key Laboratory of Precision Diagnosis and Therapy for Major Gynecological Diseases, Women's Hospital, Zhejiang University School of Medicine, Hangzhou, China. [5]Institute of Genetics, Zhejiang University School of Medicine, Hangzhou, China. [6]Cancer Center, Zhejiang University, Hangzhou, China. [7]Zhejiang Provincial Clinical Research Center for Child Health, Hangzhou, China. [8]These authors contributed equally: Xiaoyue Hu, He Li.[9]These authors jointly supervised this work: Junbin Qian, Hangjin Jiang. ✉e-mail: dr_qian@zju.edu.cn; jianghj@zju.edu.cn

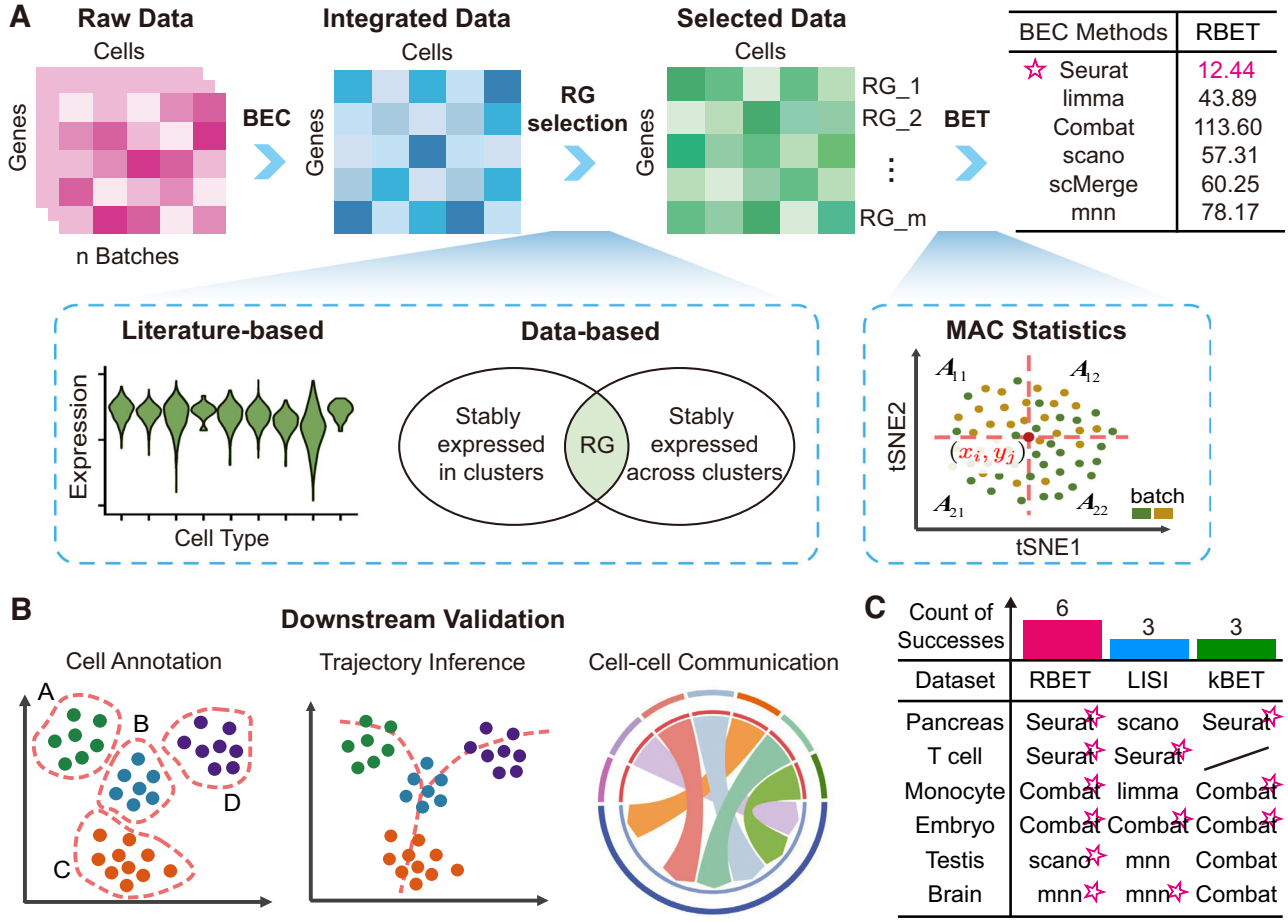

Fig. 1 | Study overview. A Overview of RBET. Single-cell sequencing data from different batches are integrated using various batch effect correction (BEC) tools. RBET first selects reference genes (RGs) either based on literature or calculated from data, and then performs batch effect testing (BET) on the selected data using MAC statistics, and finally chooses the BEC method with the smallest RBET value. **B** The choices of RBET are validated through downstream analyses including cell annotation, trajectory inference and cell-cell communication. **C** Summary of the performance of RBET and its competitors LISI and kBET in five scRNA-seq datasets and one scATAC-seq dataset (mouse brain). RBET consistently makes optimal choices under all the circumstances, while LISI and kBET only performs well in three examples.

with overcorrection awareness, large batch effect robustness and high computational efficiency, as compared to kBET and LISI. More importantly, RBET excels in prioritizing BEC tools that feed downstream analyses consistent with prior biological knowledge (Fig. 1C).

Results

Overview of RBET framework

Motivated by the consistent expression patterns of housekeeping genes across various cell types under diverse conditions[11–14] (see also Fig. S1; "Methods"), we assume that the integrated data should have no batch effect on genes with similar characteristics both locally and globally, termed reference genes (RGs) (see "Methods"). Based on this assumption, RBET framework for evaluating the performance of different BEC methods is comprised of two steps (Fig. 1A). Step 1 selects RGs specific to each dataset, with two strategies available. We collected existing validated tissue-specific housekeeping genes as RGs from published literature.For cases where validated tissue-specific housekeeping genes were not available, we directly selected RGs from the datasets, assuming that RGs should be stably expressed both within and across phenotypically different clusters (see "Methods"). By default, we used the first strategy in this step. Step 2 detects batch effect on these RGs in the integrated dataset. Noting that testing the batch effect between two samples equals to comparing their underlying distributions in a high-dimensional setting, we proposed to map the dataset into a two-dimensional space using UMAP[12], and used our previously designed MAC statistics[13] for batch effect detecting (see "Methods").

We next compared the performance of RBET with its competitors, LISI and kBET, by comprehensive simulations, and then validated the performance of RBET through real data downstream analyses, including cell annotation, trajectory inference and cell-cell communication (Fig. 1B). Note that a smaller RBET and kBET value indicates a better BEC performance, while a larger LISI value signals a better one. Since most scRNA-seq downstream analyses require full gene expression matrices, we focused on six BEC tools capable of returning full dimensional data, including Seurat[14], scanorama[15], scMerge[16], limma[17], Combat[18] and mnnCorrect[19] (Fig. 1A). Tools like Harmony[7] and fastMNN[20], which only output low-dimensional embedding, were not further explored in this study.

RBET substantially outperforms its competitors and is sensitive to overcorrection

We first compared the batch effect detection performance of RBET, kBET and LISI under two simulation strategies: (1) Gaussian examples with different means or covariance structures modeling different patterns of batch effect as a toy model (Fig. S2A–S2C); and (2) examples with simulated gene expression level mimicking real data under different cell type numbers and batch effect sizes (Fig. S3A–S3B). In our simulations, the power of each method was evaluated from 100 independent repetitions under significance level of 0.05 ("Methods").

In Gaussian examples, RBET showed comparable performance with kBET and LISI with different means or covariance structures (Fig. S2A–S2C). When turning to examples with simulated gene expression data,

RBET outperformed LISI in terms of detection power, while kBET lost control over type I error across both single and multiple cell types (Fig. 2A, B). Moreover, RBET topped the computational efficiency test, demonstrating its potential to scale up for high batch number in big dataset (Fig. 2C). Since such scaled scenario may introduce larger batch effect sizes, we tested RBET, kBET and LISI for their variability using the coefficient of variation (CV) by artificially increasing the effect size ("Methods"). Strikingly, RBET remained its variation across full size range, while the variations of LISI and kBET collapsed into zero when batch effect size was large, indicating their reduced discrimination capacity in datasets with strong batch effect (Fig. 2D). In real data, the batch effect may ocurr in only parts of cell types[21]. Thus, we also evaluated the performance of these metrics under conditions with partial batch effects, and RBET achieved higher detection power while maintaining control over type I error (Fig. S4).

Importantly, we investigated the performance of RBET, LISI and kBET when facing overcorrection of batch effects. We noted that some BEC algorithms, i.e., Seurat V5, use nearest neighbor as foundation for BEC, and increasing the number of neighbors used for correction, i.e., increasing the number of anchors (k), may lead to the loss of variation in gene expression and true cell type information, potentially causing serious overcorrection. Indeed, when k increased from 1 to 200 in Seurat, CD14+ monocytes were erroneously divided into two clusters, and pDCs were incorrectly merged with a subset of cytotoxic T cells (Fig. 2E). Additionally, RGs exhibited a loss of expression variation, which should have remained stable before and after BEC (Fig. S3C). Accordingly, we found that RBET value decreased gradually in the first phase until k reached 3 (minimal value), while a further increase of k led to gradual increase of RBET value at the second phase, which coincided with an increased level of overcorrection (Fig. 2F). Such biphasic change was not seen for kBET and LISI. This highlighted the importance of incorporating RGs in RBET.

To sum up, RBET was favored in multi-scenario simulated datasets in terms of detecting power, type I error control, computational efficiency, robustness to batch effect, and sensitivity to overcorrection.

RBET outperforms competitors in terms of cell annotations

To benchmark RBET in real data analysis, we started from the pancreas dataset which included 3 technical batches (Celseq, Celseq2 and SMART-seq2) and 13 cell types with varied numbers of cells in each cell type (7–2065)[22] (Fig. 3A). To evaluate RBET performance, we collected experimentally validated housekeeping genes specific to pancreas as candidate RGs[23], and checked whether they were differentially expressed across batches (Table S1, see "Methods"). Cells from different batches had strong batch effect, as evidenced by not well mixed clusters in UMAP (Fig. 3B), as well as the large values of RBET and kBET and a small value of LISI (Fig. 3C). After data integration by six BEC tools, both RBET and kBET selected Seurat as the best method, while LISI favored scanorama (Fig. 3C). The results of methods not selected by any evaluation tool were given in Fig. S5A.

As noted, scanorama clusters were not well mixed by batches (Fig. 3D). Thus, we further quantified the clustering quality using Silhouette Coefficient (SC; "Methods"). SC ranges form -1 to 1, where values closer to 1 indicate better-defined and well-separated clusters. Seurat received a much higher SC score than scanorama (Fig. 3E), indicating better clustering performance from Seurat. Next, we utilized cell annotation to further validate the selections. The cell types were annotated by ScType[24] with the help of marker genes[25]. We compared cell annotation results from different BEC methods with real cell tags using accuracy (ACC), Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI) ("Methods"). Both Seurat and scanorama obtained high scores exceeding 0.9 for ACC, ARI and NMI, with Seurat outperforming scanorama in all metrics (Fig. 3E, S5B). Overall, Seurat demonstrated superior annotation precision and clustering quality, confirming the selection of RBET and kBET.

RBET outperforms competitors in terms of trajectory analysis

Trajectory analysis infers dynamic process of cell differentiation and state transition, and provides great insights into the biological phenomena[26,27]. In this section, we validated the performance of RBET for multi-lineage CD8 + T cell development[28,29], single-lineage monocyte[27] and embryo dataset[30] with literature-based RGs given in Table S2-S4.

The CD8 + T cell dataset was extracted from ovarian, lung and colorectal cancers, including 40 samples and 16,133 cells (Fig. 4A, B, S6A)[11]. We found that RBET and LISI both preferred Seurat, but kBET gave a constant value of 1 to all the BEC tools (Fig. 4C), suggesting a failure in evaluating dataset with high batch effect size, consistent with our simulation results (Fig. 2D). Results for other methods were given in Fig. S6. To nail down the ideal BEC tool for downstream trajectory analysis, we took advantage of a top rated tree-based tool for trajectory inference, Slingshot[26]. This dataset consisted of two common CD8 + T cell lineages, CD8+ Terma (cytotoxic) and CD8+ Tex (exhuastic), which are both differentiated from CD8+ Tn (naïve) cells as previously validated[11]. Seurat, limma, Combat and scanorama correctly identified these two lineages using Slingshot, while lineages derived from scMerge and mnnCorrect were not consistent with CD8 + T cell biology (Fig. 4D, S6B). Since this T cell dataset was derived from different cancer types, their cell distribution along the pseudotime trajectory may vary accordingly, but they should still more or less reflect the cell fraction characteristics of T cell subtype in certain cancer (Fig. 4E, S6C). For example, colorectal cancer had relatively less Temra cells, and ovarian cancer had less Tex cells (Fig. 4A), which was correctly reflected in Seurat trajectory but not in limma and mnnCorrect (Fig. 4E, S6C, note the last peak at the end of pseudotime for each lineage). We then searched for the expression patterns of classical marker genes along the lineage-dependent pseudotime of CD8 + T cell differentiation: (1) a gradual decrease of Tn marker *CCR7* and a gradual upregulation of activation marker *NKG7* along the trajectories; (2) the terminal stage marker genes *FGFBP2* for Terma and *PDCD1*, *HAVCR2* for Tex, gradually increase along Temra and Tex lineages, respectively; (3) the Tem marker *GZMK* expresses similarly high in the middle of both trajectories, since Tem is a common progenitor state before differentiation into Temra and Tex stages[11]. According to these expected gene expression patterns, trajectories derived from Seurat generated acceptable results (Fig. 4F), and RBET's top choice Seurat outperformed other tools for this complex dataset in terms of marker gene consistency and cell distribution consistency (Fig. 4G). Interestingly, CD8+ Tex lineage is the known target for anti-PD-1/PD-L1 treatment[31], and *PDCD1* that encodes PD-1 protein expressed relatively higher in Tex lineage for lung cancer as compared to other cancers in Seurat processed trajectory (Fig. 4H), which correctly reflected current superior clinical outcome of anti-PD-1/PD-L1 treatment for lung cancer.

In addition, RBET also chose the optimal BEC methods for single-lineage monocyte and embryo datasets (Fig. S7–S10). In summary, using different examples with varying numbers of batches, sizes of batch effect and numbers of cell types, we concluded that (1) kBET, LISI and RBET selected different BEC methods as the optimal one; (2) the BEC method selected by RBET gave optimal results on trajectory analysis. Thus, if we used LISI or kBET to guide the selection, we may get misleading results on trajectory, hindering our understanding in biology system.

RBET outperforms competitors in terms of cell-cell communication analysis

To investigate the influence of BEC on cell-cell communication analysis, we focused on the testicular data from healthy males[32,33], which contained two batches that covered all the developmental stages from spermatogonial stem cells (SSCs) to mature spermatozoa. With almost identical composition of cell types, the batch influence was mainly in late stage (Fig. 5A, B). This batch effect was still retained after limma and Combat correction as their sperm cells were still split by batches, while other four methods gave acceptable batch correction on UMAP (Fig. 5D, S11A). To facilitate RBET, 12 housekeeping genes were curated from literature and four of them were selected as RGs (Table S5). RBET, LISI and kBET selected scanorama, mnnCorrect and Combat, respectively (Fig. 5C). To make further comparison, we dived into downstream analyses including trajectory inference and cell-cell communication analysis. A single developmental trajectory

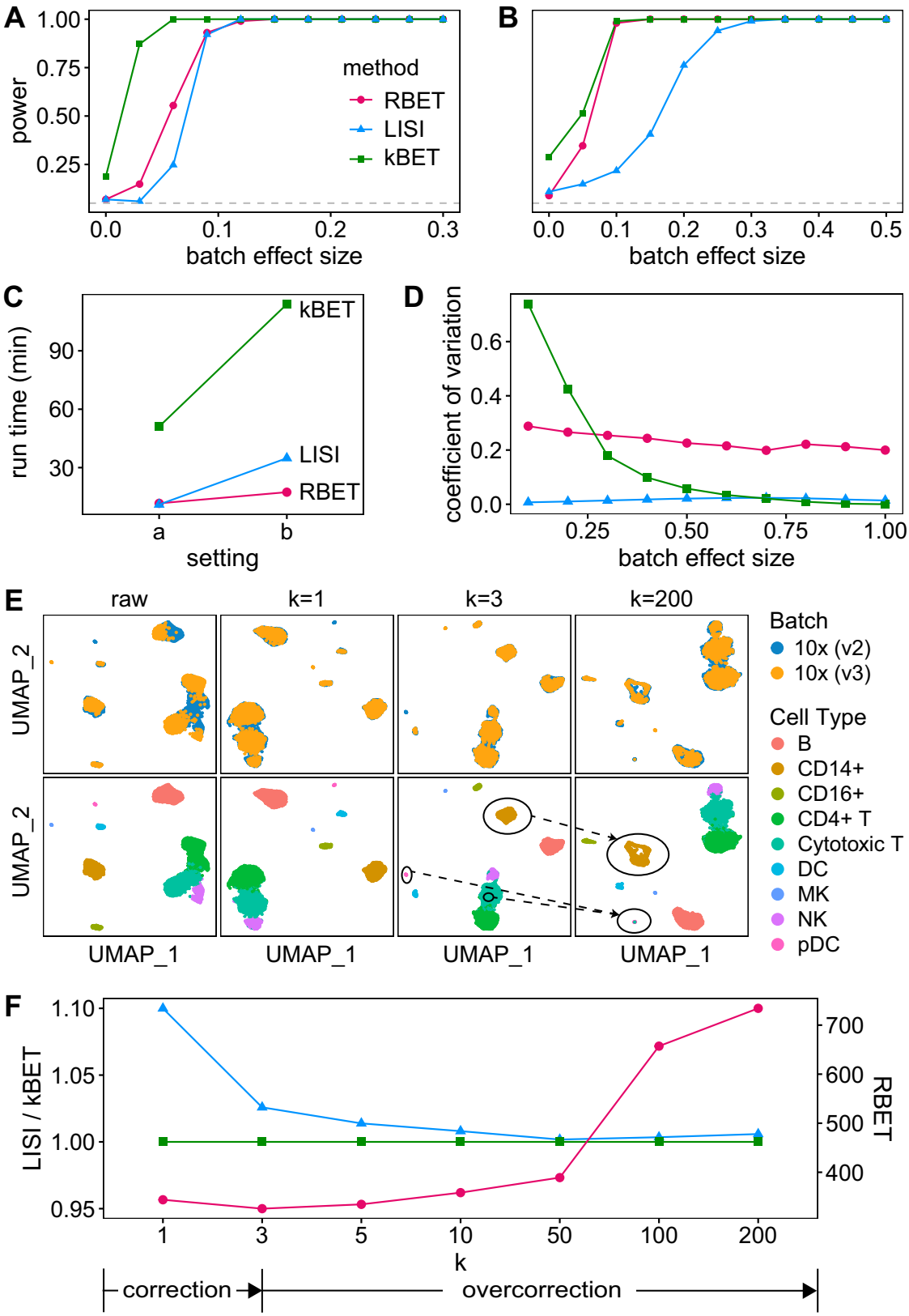

Fig. 2 | RBET substantially outperforms competitors and is sensitive to over-correction. A–D RBET outperforms its competitors kBET and LISI in the simulated gene expression example. The power of RBET, kBET and LISI under different batch effect sizes in dataset with (**A**) one cell type and (**B**) two cell types. The gray dotted line presents the theoretical power (0.05) when the null hypothesis is true, i.e., there is no batch effect. **C** The computational time used for 20 repetitions under two settings: (a) one cell type; (b) two cell types. **D** The variability of different methods measured by the coefficient of variation (CV) under different batch effect sizes. When its CV equals 0, the method has no discriminative power in BEC evaluation. **E, F** RBET is sensitive to overcorrection while LISI and kBET are not. We simulate a complex dataset containing 2 batches of cells, each with 9 cell types, which are B cell, CD14+ monocyte, CD16+ monocyte, CD4 + T cell, cytotoxic T cell, dentritic cell (DC), megakaryocyte (MK), natural killer cell (NK), and plasmacytoid dentritic cell (pDC). BEC is performed by Seurat V5 with different numbers of anchors (k). **E** UMAP plots show examples of simulated data before (k = 0) and after (k = 1, 3, 200) BEC. **F** The values of RBET, kBET and LISI with the change of k.

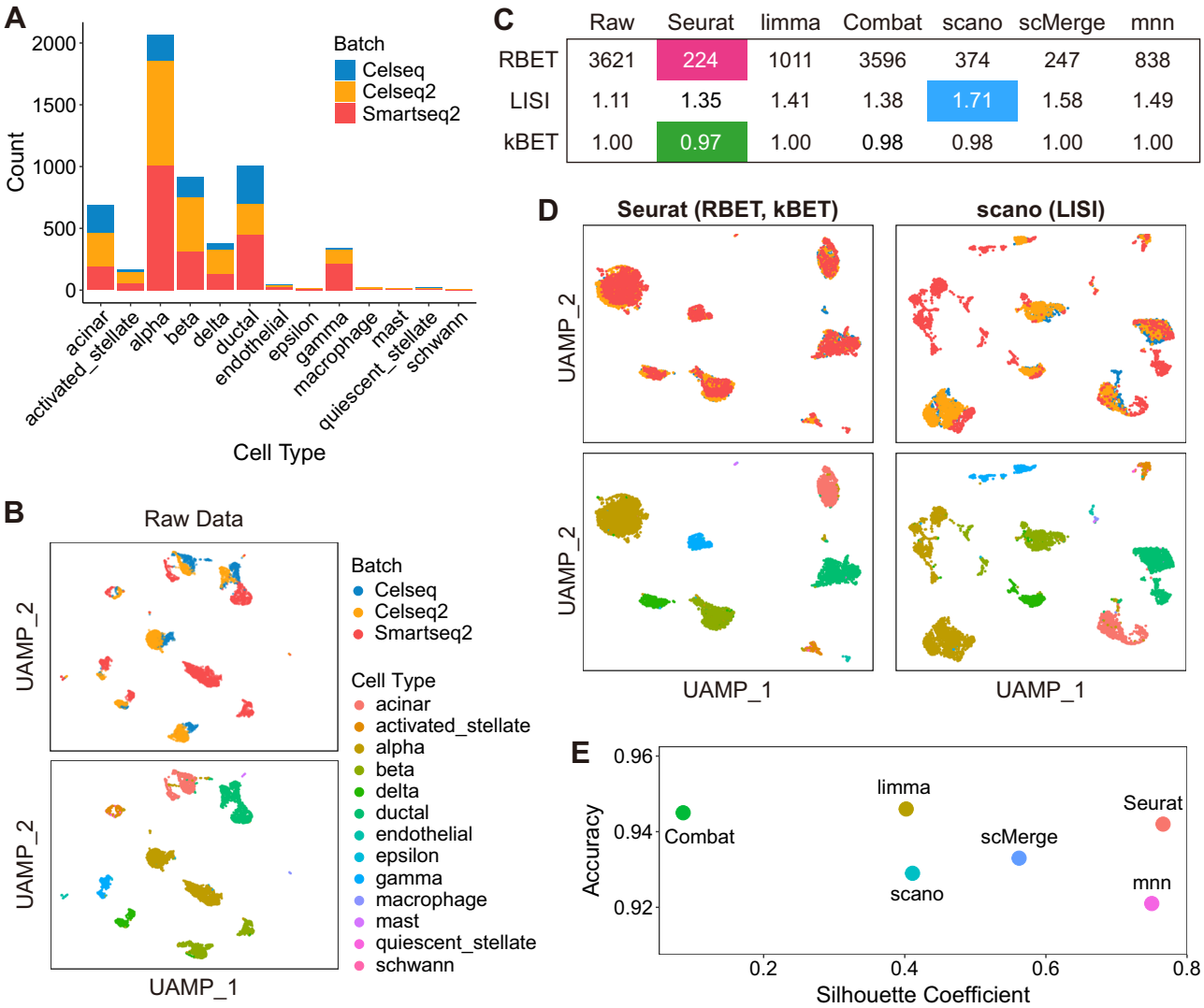

Fig. 3 | RBET selects the optimal BEC method in pancreas data validated by high cell annotation accuracy. A Histogram of cell type composition in each batch. **B** UMAP visualization of original dataset colored by batches (up) and cell types (down). **C** Evaluation scores of RBET, LISI and kBET for different BEC methods, with their optimal choices colored in red, blue and green, respectively. **D** UMAP visualization of selected integrated datasets colored by batches (up) and cell types (down). **E** The clustering quality (measured by Silhouette Coefficient) and the annotation precision (measured by accuracy between inferred cell annotations and real cell tags) of different BEC methods. Note that accuracy ranges in [0,1], while SC ranges in [−1,1], and a larger value indicates a better result.

from SSCs to sperm cells was seen from scanorama and mnnCorrect data, while the trajectory produced by Combat split into two ends, contradicted with biology (Fig. 5D, S11A).

Next, we focused on cell-cell communication from Sertoli cells to other cell types, since the former serves as germ cell ecological niches that support germ cell development[34]. These three BEC methods all gave the highest communication score to the Sertoli-SSC pair, which is consistent with previous report[32] (Fig. 5E). Further, we explored ligand-receptor (L-R) interactions and inferred their related KEGG pathway activity between cell types. For the four established Sertoli-SSC pathways that are crucial for spermatogenesis[35], only scanorama recovered all of them (Fig. 5F). Furthermore, data corrected by scanorama identified transcription factor (TF) SMAD1 and HES1 as key L-R downstream signaling effectors for Sertoli-SSC communication and spermatogenesis (Fig. 5G), which is consistent with previous findings that SMAD1/2 and HES1 are intracellular effectors of Notch signaling pathway for SSC proliferation and differentiation[32,36]. However, the same analysis based on results from other BEC tools only identified one or none of these important TFs (Fig. 5G, S11D). In summary, RBET correctly selected scanorama as the best BEC tool in the testicular dataset, which was validated by trajectory inference and cell-cell communication analysis.

Applying RBET to single-cell ATAC-seq data

RBET was not only effective in single-cell RNA-seq data, but also extendable to other single-cell modalities. Here, we evaluated a scATAC-seq integration task, using two batches of gene activity data for mouse brain derived from scATAC-seq peak calling[37]. Notably, the gene activity matrix presented the regulatory potential score of each gene isoform at the single-cell resolution[38], and thus scMerge which required standard gene name as input was not applicable here. Meanwhile, housekeeping gene isoform is not commonly used, and therefore data-based RBET was implemented, illustrating the broader application of data-based RG selection and its potential in other single-cell modalities. As shown in Fig. 6A, raw data contained two technical batches with 7 cell types, including varying numbers of cells in each cell type. Witnessed the large batch effect in the raw data (Fig. 6B), we performed BEC with all the methods other than scMerge, and evaluated their results (Fig. 6C). RBET and LISI both selected mnnCorrect, while kBET chose Combat as the optimal method (Fig. 6C). Obviously, Combat

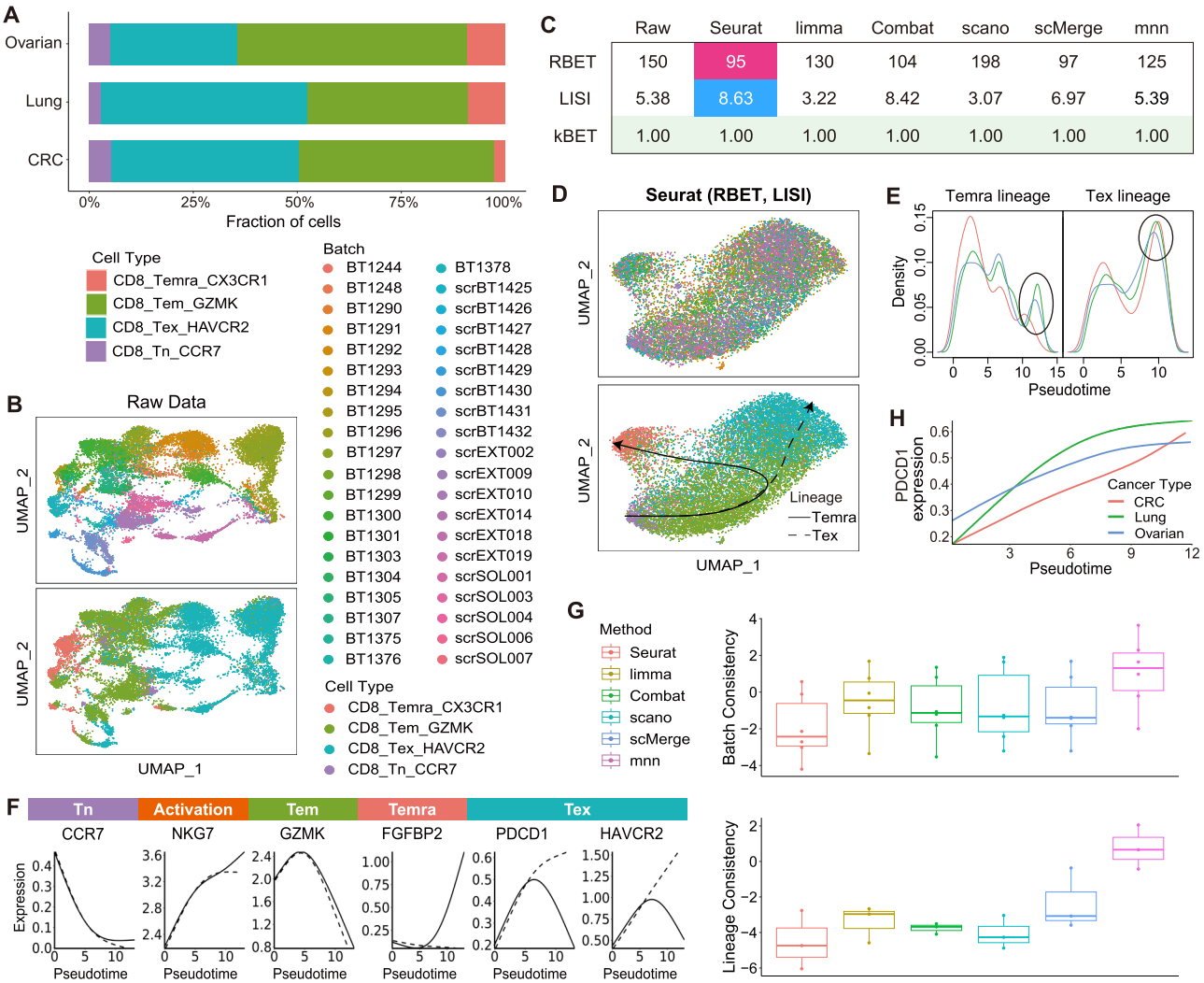

Fig. 4 | RBET chooses the optimal BEC method for CD8 + T cell dataset validated by trajectory analysis. A Histogram of cell type composition per cancer type. **B** UMAP visualization of original dataset colored by batches (up) and cell types (down). **C** Evaluation scores of RBET, LISI and kBET for different BEC methods, with their optimal choices colored in red, blue and green, respectively. kBET lose its power by giving the same value to all methods. **D** UMAP visualization of selected integrated datasets colored by batches (up) and cell types (down), together with inferred trajectories (down). **E** Density plots for colorectal cancer (CRC), lung cancer and ovarian cancer along the two trajectories. **F** The expression profiles of 6 marker genes along the pseudotime, split by lineages, solid lines for Temra lineage and dashed lines for Tex lineage. **G** The consistency of the expression profile across batches for 6 marker genes in (**F**) (up), and the consistency of the expression profile across lineages for 3 marker genes (*CCR7*, *NKG7* and *GZMK*) (down). A smaller value indicates better consistency. **H** Expression profiles of *PDCD1* for CRC, lung cancer and ovarian cancer along the Tex lineage.

underperformed in the UMAP visualization, as it left batch structure within the same cell type, while mnnCorrect fully integrated batches with well-mixed cells (Fig. 6D). The poorer clustering quality of Combat was also evidenced by its smaller SC value (Fig. 6E). Furthermore, cell annotation metrics implied that Combat had comparable accuracy performance (ARI, ACC and NMI) with mnnCorrect (Fig. 6E, S12B). To sum up, mnnCorrect outperformed Combat, implying an inferior choice of kBET.

Discussion

RBET provides a robust guidance on selecting case-specific BEC methods. It consists of two steps: (1) selecting RGs; and (2) batch effect testing on RGs in the integrated data. Different from kBET and LISI, the principle underlying RBET is that BEC should leave no batch effect only on RGs but not on all the genes. In both simulated (Fig. 2) and multi-scenario real datasets (Fig. 1C), RBET outperformed LISI and kBET in terms of batch effect detecting power, type I error control, overcorrection awareness, computational efficiency and large batch effect robustness.

The idea of RBET is inspired by the concept of housekeeping genes or RGs, which are stably and abundantly expressed with moderate expression variation across cells. There are two different strategies for constructing RGs, i.e., literature-based and data-based RGs. The first strategy is preferred in applications, and we manually curated experimentally validated housekeeping genes from literature (Table S6). When literature-based RGs are not available, we recommend data-based RGs, which may also provide reliable results (Table S7; Fig. 6). To be noted, the result of data-based RBET depends on its parameters, and thus further efforts are in need to make it more reliable.

Previous benchmark studies largely focused on metrics for cell annotation, with only a few downstream analyses including trajectory conservation. We went one step further to explore single- and multi-lineage trajectories (Fig. 4, S7–S10) according to known marker expression patterns along the differentiation pseudotime, and examined their expression consistency, as well as cell distribution consistency across batches. Moreover, we performed cell-cell communication analysis for the testicular dataset and

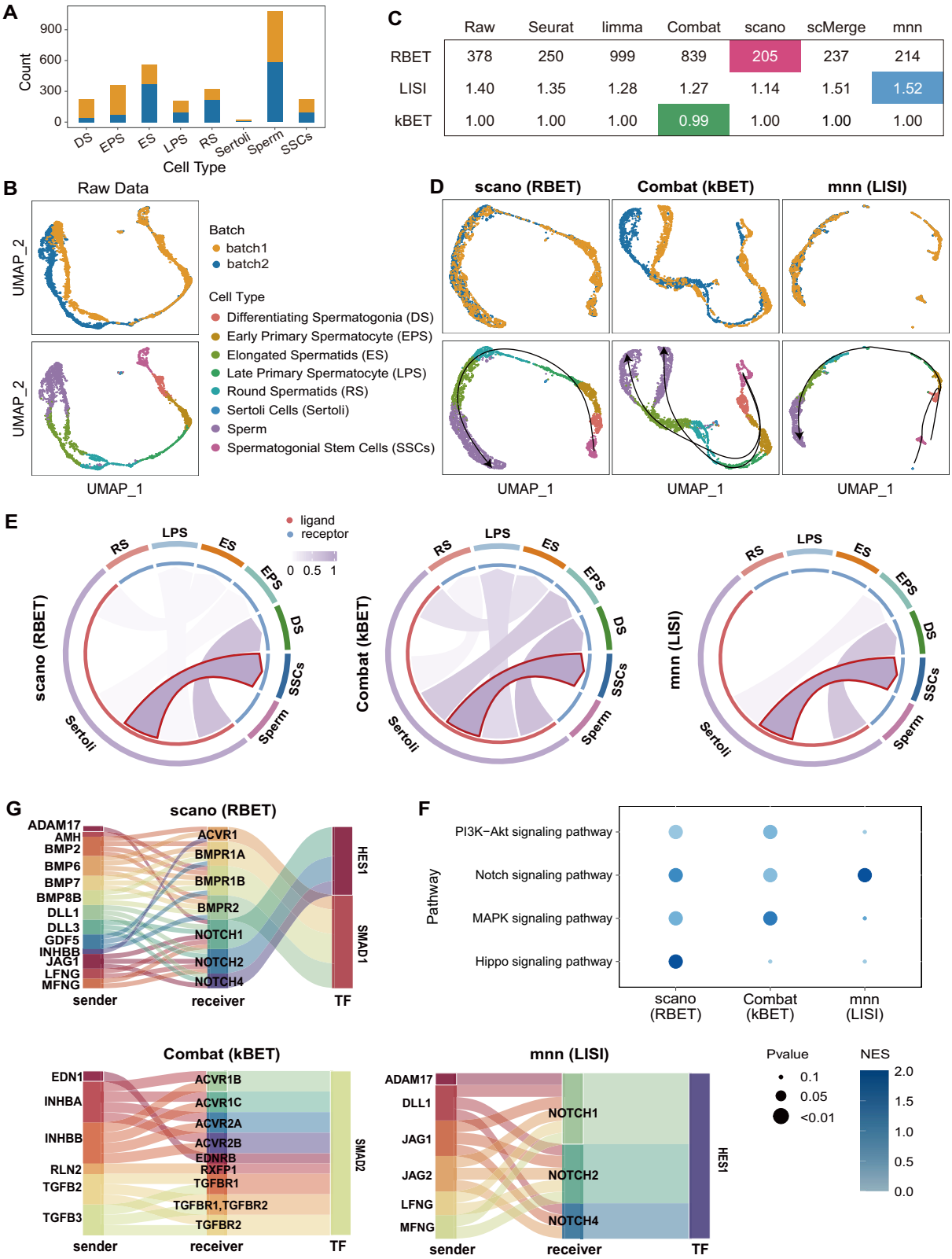

Fig. 5 | RBET achieves the best performance in testicular data in terms of cell-cell communication. **A** Histogram of cell composition in each batch. **B** UMAP visualization of original dataset colored by batches (up) and cell types (down). **C** Evaluation scores of RBET, LISI and kBET for different BEC methods, with their optimal choices colored in red, blue and green, respectively. **D** UMAP visualization of selected integrated datasets colored by batches (up) and cell types (down), together with inferred trajectories (down). **E** Strength of communications from Sertoli cells to other germ cells. The arrow outlined in red indicates the strongest communication strength. **F** Pathway activity of signaling pathways reported to be enriched by intercellular signaling from Sertoli cells to SSCs. **G** Sankey plot of the intercellular signaling from Sertoli cells to SSCs.

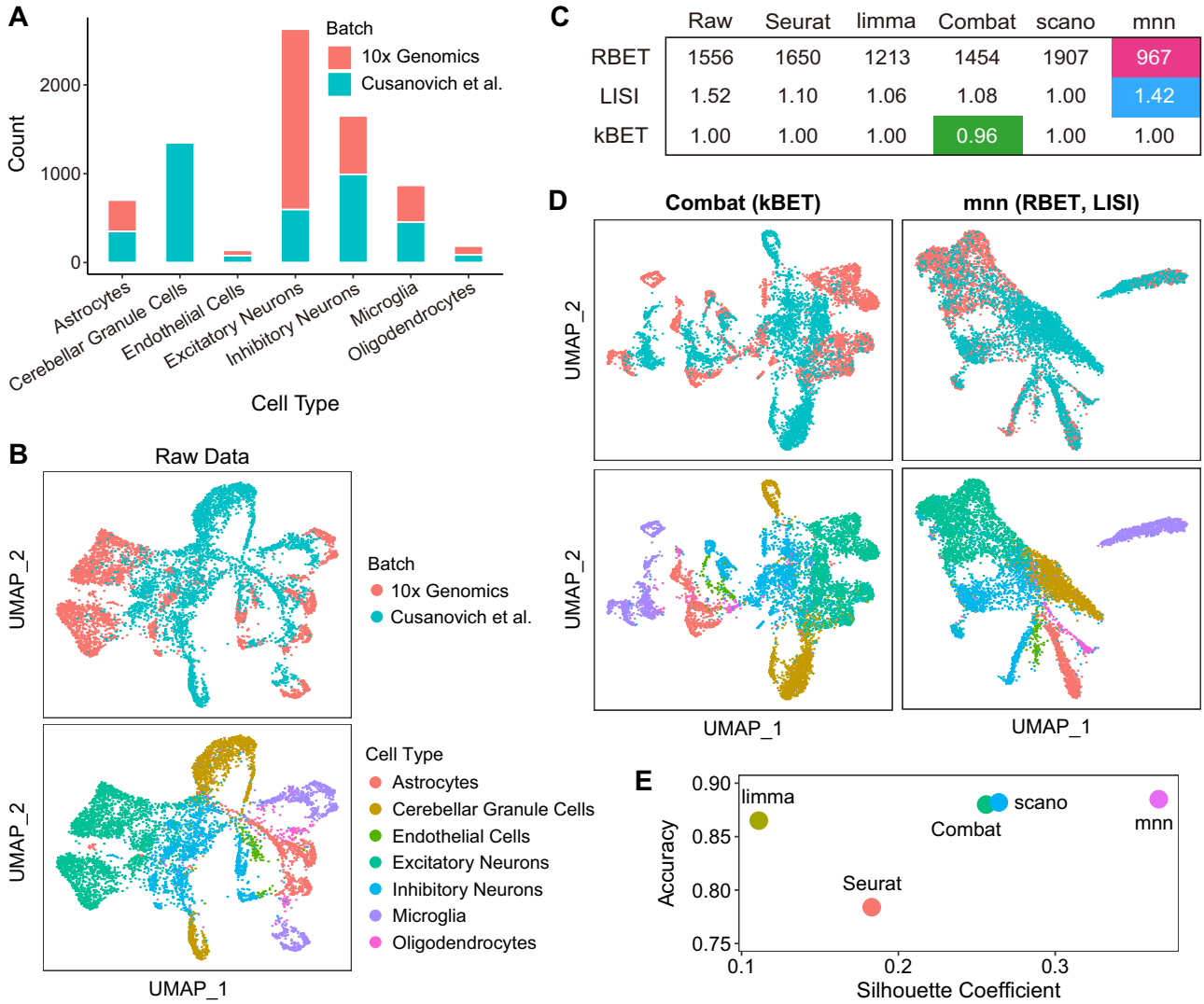

Fig. 6 | RBET performs the best in scATAC-seq data (mouse brain). **A** Histogram of cell type composition in each batch. **B** UMAP visualization of original dataset colored by batches (up) and cell types (down). **C** Evaluation scores of RBET, LISI and kBET for different BEC methods, with their optimal choices colored in red, blue and green, respectively. **D** UMAP visualization of selected integrated datasets colored by batches (up) and cell types (down). **E** The clustering quality (measured by Silhouette Coefficient) and the annotation precision (measured by accuracy between inferred cell annotations and real cell tags) of different BEC methods. Note that accuracy ranges in [0,1], while SC ranges in [−1,1], and a larger value indicates a better result.

identified known interaction paired as well as key L-R downstream signal pathways and TFs (Fig. 5), where RBET demonstrated its superiority .

Some BEC benchmark studies have comprehensively evaluated the data integration outcomes using up to 14 metrics. Different conclusions were drawn as some recommended all-weather top performers while other suggested a scenario-based guideline of choices. Our analyses favored the latter (Fig. 1C), since a simple linear regression model Combat was suitable for simple biological or technical replicates with limited cell type complexity (Fig. S7) and nonlinear Seurat and scanorama performed better in high batch effect and complex cell type composition scenarios (Figs. 4, 5). Nevertheless, the use of different metrics can lead to opposite conclusion, for example, a former top rated Harmony was discouraged to be used for complex dataset in a later study[9,10,37]. This discrepancy may stem from the use of different test datasets, as well as different multi-metric systems, which is another dilemma for common practitioners due to the lack of consensus. Different metrics only assess, mostly indirectly, limited aspects of BEC performance, and it appears tedious to perform a 14-metric benchmarking in routine practice, not to mention the expertise required to interpret and rank metrics for datasets of different research fields. Given the multi-feature and easy to use advantages of RBET, we propose to apply RBET for routine BEC optimization practice, either by comparing different BEC tools or by finetuning parameters for a favorite BEC tool. Actually, this may open an era for case-specific optimal decision making for BEC tools instead of following a general guideline.

Finally, given the concept of RGs and dimensionality reduction that can be similarly applied to other single-cell modalities, i.e., scATAC-seq (Fig. 6), CITE-seq and MERFISH, it is conceivable to expand the utility of RBET to broader BEC tasks. RBET is provided as an R package with curated lists of housekeeping genes for multiple tissue types, RG calculation tool and MAC test function. In addition, the package offers user friendly interface to common BEC tools like Seurat, which is available at https://github.com/zlyx26/RBET.

Methods

Motivation

The underlying principle is motivated by the biological features of housekeeping genes, which refer to genes that are essential for basic cellular functions and the maintenance of cellular viability[39], and show uniform expression across various cell types under diverse conditions. Given their consistency, they are often used as internal controls and standards for the

expression normalization and quantification[30,40,41]. Unlike most other genes in scRNA-seq data, housekeeping genes are stably and abundantly expressed, thus almost unaffected by dropout events. This is important because the inflated zeros in gene expression comprise both biological and technical variations, which are hard to discern. When evaluating batch correction for a human peripheral blood mononuclear cell (PBMC) dataset by Seurat, we find that the expression of housekeeping genes (i.e., RPS17 and RPS18 in Fig. S1A, S1B) is changed not only in expression level, but also in expression variation within the cell types before and after BEC, suggesting a potential usage for separating technical and biological sources of variation. Thus, we propose that a perfect batch correction should align the expression level of housekeeping genes between batches (technical batch correction), but not at the expense of losing expression variation within the batch, or in other word, loss of biological information that causes overcorrection.

RBET framework

RBET is proposed for evaluating the success of BEC when integrating multiple single-cell sequencing datasets. It consists of two steps: (1) selecting reference genes (RGs); and (2) tesing batch effects on RGs in integrated data. There are two different strategies for selecting RGs, i.e., literature-based RGs and data-based RGs.

Literature-based reference genes selection

Although housekeeping genes show uniform expression across a panel of tissues, the expression stability of these genes differ among cell types and tissues[42]. Thus, to obtain reliable results, literature-based RGs are selected from experimentally validated housekeeping genes specific to the tissue or cell type of interest. It goes in four steps as follows. **(Step 1)** Annotate cell types in each batch by manual annotation or automatic annotation tools like ScType[24]. **(Step 2)** Select experimentally validated housekeeping genes specific to the current tissue or cell type as candidates from existing literature. To facilitate the analysis, we provide a database of experimentally validated housekeeping genes for 23 tissues or cell types collected from literature (Table S6), also available in our R package. **(Step 3)** Rank the candidate genes according to the number of cell types where they are differentially expressed across batches. First, the scRNA-seq data is log-transformed by log(1+count). Next, we use Mood's median test to test whether a candidate gene is differentially expressed in each cell type (mood.medtest function in R package *RVAideMemoire*, v0.9-83), and get the p-value p_{ij} for the i-th candidate in the j-th cell type. Then, we count the number n_i of cell types where the i-th gene is differentially expressed across batches, i.e., $n_i = \sum_j I(p_{ij}<0.05)$. Finally, we rank the candidate genes according to n_i. **(Step 4)** Those candidates with the largest n_i are taken as RGs, and each batch should express at least one RG. Note that the number of RGs should be at least 2.

Data-based reference genes selection

In case of no reported tissue-specific housekeeping genes or difficult cell annotation, data-based RG selection is also provided, which chooses RGs directly from data. Similar with literature-based method, it selects genes that express invariably in each batch and differentially across batches. It contains the following five steps. **(Step 1)** Cluster cells in each batch (FindClusters function in R package *Seurat*). **(Step 2)** Select genes that express invariably in clusters. For each batch, we subset the expression of the i-th gene in the j-th cluster as e_{ij}, and compute an index as the standard deviation (sd) of e_{ij} divided by the mean of e_{ij}, that is, $I_{ij} = \mathrm{sd}(e_{ij})/\mathrm{mean}(e_{ij})$. Then, we rank the genes according to the index from small to large in each cluster, and count the number of clusters c_i where the i-th gene is in the top r%. Next, we rank the genes according to c_i from large to small in each batch, and count the number of batches b_i where the i-th gene is in the top r%. Finally, we rank the genes according to b_i from large to small, and the genes in the top r% are taken as candidates in clusters. **(Step 3)** Select genes that express invariably across clusters. For each batch, the expression of the i-th gene is taken as e_i and the mean expression of the i-th gene in each cluster is taken as m_i. We compute an index $I_i = \mathrm{sd}(m_i)/\mathrm{mean}(e_i)$. Then we rank the genes according to the index from small to large in each batch, and count the number of batches b_i where the i-th gene is in the top r%. Finally, we rank the genes according to b_i from large to small, and the genes in the top r% are taken as candidates across clusters. **(Step 4)** We first take an intersection of candidates in clusters and across clusters. Then, we count the times n_i when the i-th candidate gene expresses differentially across batches by comparing pairwise clusters in pairwise batches (mood.medtest function in R package *RVAideMemoire*, v0.9-83), and rank the candidates according to n_i from large to small. **(Step 5)** The top l candidates are taken as RGs. Here, we take $r = 2$ and $l = 50$.

Batch effect testing based on two-sample distribution comparison

Let $\mathbf{X}\ \mathbb{R}^{np}$ and $\mathbf{Y}\ \mathbb{R}^{mp}$ be the gene expression matrices of two different batches, where n and m are cell numbers and p is the gene number. Input matrices with different genes can be inner joined together by shared genes, and thus the gene numbers become the same. We notice that samples from two different batches can be well regarded as samples from two different distributions. In such a view, the problem of batch effect detection naturally fits into the framework of two-sample distribution test, with the null hypothesis that $\mathbf{X}$ and $\mathbf{Y}$ have the same distribution:

$$H_0 : F = G \text{ versus } H_1 : F \neq G,$$

where F and G are distribution functions of $\mathbf{X}$ and $\mathbf{Y}$, respectively.

For the case $p = 1$, this problem is well studied in statistical literature[43]. However, only few methods have been proposed for the high-dimensional cases. Recently, we introduced a class of MAC statistics for this problem[13]. Theoretical analysis has demonstrated that MAC has an upper bound of order $(\log(nm))$ when there is no batch effect, which is much smaller than the lower bound of order $(n + m)$ when batch effect is present[13]. Notably, MAC outperforms other methods in cases $p < 10$, but it is almost inapplicable when p exceeds 10. Hence, due to the substantial size of genes in our datasets, i.e., p could be as large as thousands, MAC cannot be directly employed.

Motivated by the fact that dimensionality reduction tool UMAP well preserves the neighborhood characteristics of each point in the original space and is computationally efficient[12,44], we propose to map the original dataset into a two-dimensional vector space using UMAP, and then MAC statistics can be applied. Specifically, we put $\mathbf{X}$ and $\mathbf{Y}$ together to form a large $(n + m) \times p$ matrix, and reduce it into a two-dimensional observation $\mathbf{Z} \in \mathbb{R}^{(n+m)\times 2}$. $\mathbf{Z}$ can be partitioned into $\widetilde{\mathbf{X}}\ \mathbb{R}^{n\times 2}$ and $\widetilde{\mathbf{Y}}\ \mathbb{R}^{m\times 2}$, corresponding to $\mathbf{X}$ and $\mathbf{Y}$, respectively. In this way, we obtain a two-dimensional vector for each batch in the same vector space which well-preserves the original neighborhood characteristics. Next, we implement MAC with $p = 2$ to detect the batch effect among $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{Y}}$, namely that among $\mathbf{X}$ and $\mathbf{Y}$.

Let $\boldsymbol{x} = \{\boldsymbol{x_i} : i = 1, 2, \ldots, n\}$ be the collection of n rows of $\widetilde{\mathbf{X}}$, and $\boldsymbol{y} = \{\boldsymbol{y_i} : i = 1, 2, \ldots, m\}$ be the collection of m rows of $\widetilde{\mathbf{Y}}$. Thus, $\boldsymbol{x}$ and $\boldsymbol{y}$ are mapped samples from two different batches. Let $(\boldsymbol{x_i}, \boldsymbol{y_j})$ be any pair of mapped samples, where $\boldsymbol{x_i} = (z_{i,1}, z_{i,2})$ and $\boldsymbol{y_j} = (z_{j,1}, z_{j,2})$, and $d(\boldsymbol{a}, \boldsymbol{b}) = \sqrt{\sum_i (a_i - b_i)^2}$ be the Euclidean distance between two vectors. Define

$$A_{\boldsymbol{x}_i,\boldsymbol{y}_j} = \left\{x = (z_1, z_2) \in \mathbb{R}^2 : d(z_1, z_{i,1}) \le d\left(z_{i,1}, z_{j,1}\right)\right\},$$

$$B_{\boldsymbol{x}_i,\boldsymbol{y}_j} = \left\{x = (z_1, z_2) \in \mathbb{R}^2 : d(z_2, z_{i,2}) \le d\left(z_{i,2}, z_{j,2}\right)\right\},$$

Then, we divide the whole sample space $\mathbb{R}^2$ into four disjoint parts,

$$A_{11} = A_{\boldsymbol{x}_i,\boldsymbol{y}_j} \cap B_{\boldsymbol{x}_i,\boldsymbol{y}_j}, A_{12} = A^c_{\boldsymbol{x}_i,\boldsymbol{y}_j} \cap B_{\boldsymbol{x}_i,\boldsymbol{y}_j},$$

$$A_{21} = A_{\boldsymbol{x}_i,\boldsymbol{y}_j} \cap B^c_{\boldsymbol{x}_i,\boldsymbol{y}_j}, A_{12} = A^c_{\boldsymbol{x}_i,\boldsymbol{y}_j} \cap B^c_{\boldsymbol{x}_i,\boldsymbol{y}_j}.$$

Let $P_{ij} = \sum_{k=1}^{n} I(\boldsymbol{x_k} \in A_{ij})$, $Q_{ij} = \sum_{k=1}^{m} I(\boldsymbol{y_k} \in A_{ij})$, $R_{ij} = P_{ij} + Q_{ij}$, $i, j = 1, 2$, then the local statistic T_{ij} that compares the distribution of $\boldsymbol{x_i}$ and $\boldsymbol{y_j}$ is defined as

$$T_{ij} = T\left(\boldsymbol{x_i}, \boldsymbol{y_j}\right) = \sum_{i=1}^{2}\sum_{j=1}^{2} \frac{(P_{ij} - (\frac{n}{n+m})R_{ij})^2}{(\frac{n}{n+m})R_{ij}} + \frac{(Q_{ij} - (\frac{m}{n+m})R_{ij})^2}{(\frac{m}{n+m})R_{ij}}.$$

It should be noted that if any R_{ij} equals to zero, T_{ij} is set to zero. Finally, MAC statistic takes the maximum of all the local statistics T_{ij}, i.e.,

$$\mathrm{MAC} = \max_{1 \le i \le n, 1 \le j \le m} T_{ij}.$$

Sub-sampling strategy to improve computational efficiency

Since scRNA data often comprise hundreds or even thousands of cells, the calculation of MAC statistics on the whole sample is computationally expensive. Considering that similar cells have similar gene expression profiles and cluster together in the reduced vector space, we propose a sub-sampling strategy to improve the computational efficiency. To be more specific, we rank rows of $\boldsymbol{Z}$ according to its first column, and select samples whose row numbers equal to $\Delta \times j$ with $j = 1, 2, \ldots, k = (n + m)/\Delta$, where Δ is the sampling gap. Then we calculate RBET based on the k selected samples. It should be noted that this sampling strategy simply reduces the number of local statistics used for defining MAC, without changing its theoretical properties. Generally, a larger sub-sample k of RBET resulted in a higher power but also required more computational cost. We set $k = 50$ because $k \ge 50$ already showed a stable performance in both Gaussian examples (Fig. S2D) and simulated gene expression data (Fig. S3D).

Compared methods for batch effect testing

RBET is compared with two other methods. (1) kBET[10]. It assumes that if there is no batch effect, the distribution of batch labels within any given neighborhood of equal size should mirror that of the entire dataset. Based on this idea, kBET applies Pearson chi-squared test to randomly selected neighborhoods of a fixed size, and then returns an overall rejection rate which reflects the degree of mixing in the dataset. A low rejection rate indicates that the two datasets are well mixed and there is no or small batch effect. R package *kBET* (v0.99.6) is used here and all parameters are set by default without using sub-sampling.

2) LISI[7] based on Inverse Simpson Index (ISI). LISI first assigns weights to adjacent cells by constructing Gaussian kernel-based distributions of neighborhoods, and then calculates ISI as the effective number of batches in each cell's neighborhood. The mean of LISI from all the neighborhoods is defined as the overall number of batches. R package *lisi* (v1.0) is used for calculation with default parameters.

Test procedure for RBET and LISI

A permutation test was performed to evaluate the null hypothesis, defined as the absence of batch effect. First, we simulated a dataset based on the given batch effect size, and used RBET (LISI) to detect the batch effect, denoted as T_0. Next, we applied the permutation procedure to calculate the p-value for the null hypothesis. To do so, (a1) we randomly reshuffled the data and calculated a RBET (LISI) value from the new dataset; (a2) we repeated step (a1) for N times to obtain N values $\{T_1, T_2, \ldots, T_N\}$, representing the observations under the null distribution; (a3) we computed the p-value, defined as $\frac{\sum_{i=1}^{N} I(T_i > T_0) + 1}{N+1}$ ($\frac{\sum_{i=1}^{N} I(T_i < T_0) + 1}{N+1}$). A p-value smaller than 0.05 indicated the presence of batch effect. In this paper, N was set to 100.

Power analysis

The power of RBET, LISI and kBET on detecting batch effects was estimated through 100 independent repetitions. Specifically, it is defined as the proportion of rejections when the alternative hypothesis is true, i.e., there is batch effect in the data.

Coefficient of variation (CV)

For any random variable, CV is defined as the ratio of its standard deviation to the mean. It serves as a statistical measure of relative variability, making it useful for comparing the degree of dispersion across methods with different units or scales.

Simulation of Gaussian examples

In these examples, we generated synthetic data following Gaussian distributions. We assumed that X followed a two-dimensional Gaussian distribution, $F = N((0, 0), I)$, and considered three different cases for the distribution of Y, G. (G1) Y followed $N(\left(\mu, \mu\right), I)$ with μ varying from 0 to 1; (G2) Y followed $N((0, 0), \sigma I)$ with σ varying from 1 to 4; (G3) Y followed $N((0, 0), \Sigma)$, where $\Sigma = (\sigma_{ij})$, $\sigma_{11} = \sigma_{22} = 1$, and $\sigma_{12} = \sigma_{21} = \rho$ with ρ varying from 0 to 1. For each case, we sampled n (= 50, 100, 200) independently and identically distributed observations from F and G, respectively.

Simulation of gene expression data

In these examples, we simulated single-cell gene expression counts by learning from the example dataset in R package *scDesign3* (v1.4.0)[45], which contained 1000 genes and 6276 cells in total. Here, we simulated two settings by taking the subsets of example data as models: (1) one cell type, containing 1915 CD4 + T cells; (2) two cell types, containing 1915 CD4 + T cells and 1656 cytotoxic T cells. Cell types were selected based on the descending order of cell count in the example dataset.

Computational efficiency

We compared the total running time of RBET, LISI, kBET and kBET with sub-sampling (see supplementary material; Fig. S3E) over 20 repetitions under two settings described in **Simulation of gene expression data**, with effect size fixed to 0.1, evaluated from 50 independent repetitions.

Simulation on overcorrection of batch effect

We simulated data for the overcorrection problem by taking the example dataset in *scDesign3* with 1000 genes and 6276 cells and the effect size fixed to 1. The data were corrected by the CCA function in Seurat (v5.1.0) using different numbers of anchors (k.anchor). The gene expression were extracted by using GetAssayData function in Seurat.

Data processing

The raw data and the integrated data were mainly analyzed with Seurat (R package, v5.1.0)[14]. After data normalization and scaling, the top 2000 variable genes were calculated by FindVariableFeatures function and used to construct principal components (PCs). PCs covering the highest variance were selected according to the elbow and Jackstraw plots. Clusters were calculated by FindClusters function and visualized using the Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP).

Tools for integrating scRNA-seq datasets

Most downstream analysis, such as differential gene expression analysis and gene set enrichment analysis, require expression matrices as input. Here, we aimed to evaluate different BEC methods based on the validity and accuracy of subsequent downstream analysis. Therefore, six commonly used BEC tools that could return gene expression matrices with full dimensionality were applied for data integration: Seurat (R package, v5.1.0), limma (R package, v3.50.3), Combat (R package, v3.42.0), scanorama (R package, vl.7.2), scMerge 2 (R package, vl.22.0), and mnnCorrect (R package, v1.12.3). These BEC tools were implemented with default parameters.

Cell annotation

Cell annotation is a fundamental step for single-cell downstream analysis. For pancreas dataset, we utilized the marker-based automatic annotation tool ScType[24] together with the marker genes specific to each tissue or cell

type. ScType matched the gene expression patterns of data to the known cell types in order to label individual cells or cell clusters[24]. For mouse brain scATAC-seq dataset, as standard gene names were not provided, ScType was no longer applicable. Fortunately, we still had true cell type labels. Thus, we calculated DEGs of each cell type with FindMarkers function in Seurat, and then labeled each cell cluster manually.

Metrics used for measuring quality of cell clustering

We compared the annotation results with real cell tags under three metrics: accuracy (ACC), adjusted rank index (ARI; R package *aricode*, v1.0.2), and normalized mutual information (NMI; R package *aricode*, v1.0.2). Additionally, we evaluated the clustering quality of each cell type using silhouette coefficient (SC; R package *cluster*, v2.1.6). The SC for cell i is defined as $S(i) = \frac{b(i)-a(i)}{\max(a(i),b(i))}$, where $a(i)$ is the average distance between cell i and all the other cells in the same cell type, and $b(i)$ is the minimum average distance between cell i and all the cells in any other cell type, and the distance is calculated based on the UMAP reduced data. In this way, SC quantifies how similar a cell is to its own cluster compared to other clusters, and thus assesses whether cells of the same cell type, but from different batches, are well clustered. SC values range from -1 to 1, where higher values indicate better-defined clusters.

Trajectory inference

Trajectory analysis aims to construct an ordered sequence of cells to describe the dynamic changes in single-cell gene expression levels. Here, Slingshot (R package *slingshot*, v2.2.1) was applied to perform pseudotime analysis on the integrated datasets, which was recommended by previous studies[26]. It took the low dimensional embedding of gene expression and a vector of cluster labels as the input. Here, the labels were predicted using k-means clustering method (kmeans function in R package *stats*, v4.1.1). Then, we utilized getLineages function in *slingshot* to fit a minimum spanning tree (MST) to identify lineage. In accordance with established biological knowledge, we set the root for CD8 + T cell dataset as CD8_Tn_CCR7 and the root for testis dataset as SSCs. However, monocyte dataset lacked explicit cell type annotation, and thus the trajectory was inferred automatically without a prespecified root. The final output was stored in a SlingshotDataSet object containing inferred lineage information.

Measuring the consistency of gene expression profiles along the pseudotime

The expression profile e of gene G could be fitted as a smoothing function of the inferred pseudotime t, denoted as $e(t)$ (smooth.spline function in R package *stats*, v4.1.1). Assuming that $e_1(t)$ and $e_2(t)$ were the fitted curves of expression profiles of gene G in two different batches or lineages, we defined the consistency between these two curves as the averaged distance between them, i.e., $C = \frac{1}{b-a}\int_a^b |e_1(t) - e_2(t)|^2 dt$. It was actually the mean of $|e_1(t) - e_2(t)|^2$ by taking t as a uniformly distributed random variable on $[a, b]$. This provided us a simple measurement by first sampling n instances, $\{t_1, \ldots, t_n\}$, from $U[a, b]$, and then estimating the averaged distance as $\hat{C} = \frac{1}{n}\sum_{i=1}^{n} |e_1(t_i) - e_2(t_i)|^2$. In this paper, we took $n = 1000$, which was large enough to get an accurate estimation. To mitigate the effect of gene expression size, we normalized $\hat{C}$ by the square of the mean expression (μ_G), i.e., $\widetilde{C} = \hat{C}/\mu_G^2$. In case of multiple batches, we calculated their pairwise consistency and took the average.

Cell communication analysis

The biological behavior of cells is regulated by the intercellular and intracellular signaling network, which is known as cellular communication. Here, we performed cell-cell communication analysis on the integrated datasets using CellCall (R package *cellcall*, v1.0.7)[32]. First, we inferred the overall communication scores between different cell types using TransCommuProfile function. Next, we identified crucial pathways involved in communication by performing a pathway activity analysis with getHyperPathway function. Finally, we visualized the intercellular signaling from one cell type to another in sankey plot (LR2TF and sankey_graph functions).

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

All the datasets used for analyses in this work are publicly available online, deposited in the NCBI's Gene Expression Omnibus database and ArrayExpress. In particular, we used a pancreas dataset[22] containing three batches of single-cell expression data obtained by different sequencing technologies, CelSeq (GSE81076)[46], CelSeq2 (GSE85241)[25] and SMART-Seq2 (E-MTAB-5061)[47], a human monocyte dataset[48] under accession code GSE146974, an embryo development dataset containing germ cells in two batches, one from Li et al. (GSE86146)[25] and the other from Guo et al. (GSE63818)[26], a CD8 + T cell dataset extracted from Qian et al.[11] under accession number E-MTAB-8107, E-MTAB-6149 and E-MTAB-6653, taking each sample as a batch, a testis dataset[21] of testicular transcriptome sequencing data from healthy males under accession code GSE112013, and a mouse brain scATAC-seq dataset[37] containing two batches of gene activity data (small_atac_gene_activity).

Code availability

The code for RBET is written in R and Rcpp, and can be obtained at https://github.com/zlyx26/RBET. The code for analyses and plots is available on Zenodo: https://zenodo.org/records/14898612.

Received: 6 October 2024; Accepted: 18 March 2025;
Published online: 30 March 2025

References

1. Feng, W. et al. Single-cell transcriptomic analysis identifies murine heart molecular features at embryonic and neonatal stages. *Nat. Commun.* **13**, 7960 (2022).
2. Zhang, P. et al. Dissecting the single-cell transcriptome network underlying gastric premalignant lesions and early gastric cancer. *Cell Rep.* **27**, 1934–1947 (2019).
3. Reyes, M. et al. An immune-cell signature of bacterial sepsis. *Nat. Med.* **26**, 333–340 (2020).
4. Butler, A., Hoffman, P., Smibert, P., Papalexi, E. & Satija, R. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat. Biotechnol.* **36**, 411–420 (2018).
5. Barkas, N. et al. Joint analysis of heterogeneous single-cell RNA-seq dataset collections. *Nat. Methods* **16**, 695–698 (2019).
6. Leek, J. T. et al. Tackling the widespread and critical impact of batch effects in high-throughput data. *Nat. Rev. Genet.* **11**, 733–739 (2010).
7. Korsunsky, I. et al. Fast, sensitive and accurate integration of single-cell data with Harmony. *Nat. Methods* **16**, 1289–1296 (2019).
8. Lyu, Y., Lin, S. H., Wu, H. & Li, Z. SCIntRuler: guiding the integration of multiple single-cell RNA-seq datasets with a novel statistical metric. *Bioinformatics* **40**, btae537 (2024).
9. Tran, H. T. N. et al. A benchmark of batch-effect correction methods for single-cell RNA sequencing data. *Genome Biol.* **21**, 1–32 (2020).
10. Büttner, M., Miao, Z., Wolf, F. A., Teichmann, S. A. & Theis, F. J. A test metric for assessing single-cell RNA-seq batch correction. *Nat. Methods* **16**, 43–49 (2019).
11. Qian, J. et al. A pan-cancer blueprint of the heterogeneous tumor microenvironment revealed by single-cell profiling. *Cell Res.* **30**, 745–762 (2020).
12. Becht, E. et al. Dimensionality reduction for visualizing single-cell data using UMAP. *Nat. Biotechnol.* **37**, 38–44 (2019).

13. Jiang, H., Zhao, X., Ma, R. C. & Fan, X. Consistent screening procedures in high-dimensional binary classification. *Stat. Sin.* **32**, 109–130 (2022).
14. Hao, Y. et al. Integrated analysis of multimodal single-cell data. *Cell* **184**, 3573–3587 (2021).
15. Hie, B., Bryson, B. & Berger, B. Efficient integration of heterogeneous single-cell transcriptomes using Scanorama. *Nat. Biotechnol.* **37**, 685–691 (2019).
16. Lin, Y. et al. scMerge leverages factor analysis, stable expression, and pseudoreplication to merge multiple single-cell RNA-seq datasets. *Proc. Natl Acad. Sci.* **116**, 9775–9784 (2019).
17. Ritchie, M. E. et al. limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic acids Res.* **43**, e47–e47 (2015).
18. Zhang, Y., Parmigiani, G. & Johnson, W. E. ComBat-seq: batch effect adjustment for RNA-seq count data. *NAR Genomics Bioinforma.* **2**, lqaa078 (2020).
19. Haghverdi, L., Lun, A. T., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nat. Biotechnol.* **36**, 421–427 (2018).
20. Lun, A. Further MNN algorithm development. https://marionilab.github.io/FurtherMNN2018/theory/description.html (2019).
21. Guo, J. et al. The adult human testis transcriptional cell atlas. *Cell Res.* **28**, 1141–1157 (2018).
22. Stuart, T. et al. Comprehensive integration of single-cell data. *Cell* **177**, 1888–1902.e21 (2019).
23. Cherubini, A., Rusconi, F. & Lazzari, L. Identification of the best housekeeping gene for RT-qPCR analysis of human pancreatic organoids. *PloS one* **16**, e0260902 (2021).
24. Ianevski, A., Giri, A. K. & Aittokallio, T. Fully-automated and ultra-fast cell-type identification using specific marker combinations from single-cell transcriptomic data. *Nat. Commun.* **13**, 1246 (2022).
25. Muraro, M. et al. A single-cell transcriptome atlas of the human pancreas. *Cell Syst.* **3**, 385–394.e3 (2016).
26. Saelens, W., Cannoodt, R., Todorov, H. & Saeys, Y. A comparison of single-cell trajectory inference methods. *Nat. Biotechnol.* **37**, 547–554 (2019).
27. Street, K. et al. Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC Genomics* **19**, 477 (2018).
28. Li, L. et al. Single-cell RNA-seq analysis maps development of human germline cells and gonadal niche interactions. *Cell Stem Cell* **20**, 858–873 (2017).
29. Guo, F. et al. The transcriptome and DNA methylome landscapes of human primordial germ cells. *Cell* **161**, 1437–1452 (2015).
30. Banda, M., Bommineni, A., Thomas, R. A., Luckinbill, L. S. & Tucker, J. D. Evaluation and validation of housekeeping genes in response to ionizing radiation and chemical exposure for normalizing RNA expression in real-time PCR. *Mutat. Res.* **649**, 126–134 (2008).
31. Bassez, A. et al. A single-cell map of intratumoral changes during anti-PD1 treatment of patients with breast cancer. *Nat. Med.* **27**, 820–832 (2021).
32. Zhang, Y. et al. CellCall: Fronter paired ligand–receptor and transcription factor activities for cell–cell communication. *Nucleic Acids Res.* **49**, 8520–8534 (2021).
33. Jin, S. et al. Inference and analysis of cell-cell communication using CellChat. *Nat. Commun.* **12**, 1088 (2021).
34. Shinohara, T., Orwig, K. E., Avarbock, M. R. & Brinster, R. L. Restoration of spermatogenesis in infertile mice by Sertoli cell transplantation. *Biol. Reprod.* **68**, 1064–1071 (2003).
35. Ni, F.-D., Hao, S.-L. & Yang, W.-X. Multiple signaling pathways in Sertoli cells: recent findings in spermatogenesis. *Cell Death Dis.* **10**, 541 (2019).
36. Garcia, T. X., Parekh, P., Gandhi, P., Sinha, K. & Hofmann, M.-C. The NOTCH ligand JAG1 regulates GDNF expression in Sertoli cells. *Stem Cells Dev.* **26**, 585–598 (2017).
37. Luecken, M. D. et al. Benchmarking atlas-level data integration in single-cell genomics. *Nat. Methods* **19**, 41–50 (2022).
38. Wang, C. et al. Integrative analyses of single-cell transcriptome and regulome using MAESTRO. *Genome Biol.* **21**, 198 (2020).
39. Hounkpe, B. W., Chenou, F., de Lima, F. & De Paula, E. V. HRT Atlas v1.0 database: redefining human and mouse housekeeping genes and candidate reference transcripts by mining massive RNA-seq datasets. *Nucleic Acids Res.* **49**, D947–D955 (2021).
40. Al-Bader, M. D. & Al-Sarraf, H. A. Housekeeping gene expression during fetal brain development in the rat-validation by semi-quantitative RT-PCR. *Brain Res. Dev. Brain Res.* **156**, 38–45 (2005).
41. Thellin, O. et al. Housekeeping genes as internal standards: use and limits. *J. Biotechnol.* **75**, 291–295 (1999).
42. Molina, C. E. et al. Identification of optimal reference genes for transcriptomic analyses in normal and diseased human heart. *Cardiovascular Res.* **114**, 247–258 (2018).
43. Thas, O. *Comparing Distributions*. 233 (Springer, 2010).
44. Wang, Y., Huang, H., Rudin, C. & Shaposhnik, Y. Understanding how dimension reduction tools work: An empirical approach to deciphering t-SNE, UMAP, TriMAP, and PaCMAP for data visualization. *J. Mach. Learn. Res.* **22**, 1–73 (2021).
45. Song, D. et al. scDesign3 generates realistic in silico data for multimodal single-cell and spatial omics. *Nat. Biotechnol.* **42**, 247–252 (2024).
46. Grün, D. et al. De novo prediction of stem cell identity using single-cell transcriptome data. *Cell Stem Cell* **19**, 266–277 (2016).
47. Segerstolpe, Å. et al. Single-cell transcriptome profiling of human pancreatic islets in health and type 2 diabetes. *Cell Metab.* **24**, 593–607 (2016).
48. Li, X. et al. Deep learning enables accurate clustering with batch effect removal in single-cell RNA-seq analysis. *Nat. Commun.* **11**, 2338 (2020).

Acknowledgements

H.J. is partially supported by Key R&D Program of Zhejiang Province (2021C03G2013079), and High-level Talent Special Support Program of Zhejiang Province. J.Q. is partially supported by National Natural Science Foundation of China (82173150), and Zhejiang Province Natural Science Fund for Excellent Young Scholars (LR22H160004).

Author contributions

H.J. and J.Q. supervised the project; H.J. and J.Q. conceived of and designed the study; X.H. and H.L. collected data and performed data analysis; H.J., J.Q., X.H., H.L., and M.C. wrote the paper.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42003-025-07947-7.

Correspondence and requests for materials should be addressed to Junbin Qian or Hangjin Jiang.

Peer review information *Communications Biology* thanks the anonymous reviewers for their contribution to the peer review of this work. Primary Handling Editors: Ani Manichaikul and Mengtan Xing.

Reprints and permissions information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2025