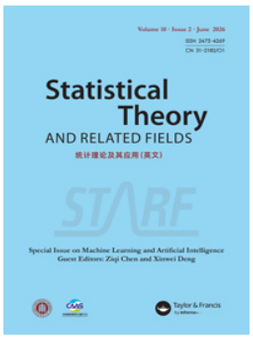

Statistical Theory and Related Fields

ISSN: 2475-4269 (Print) 2475-4277 (Online) Journal homepage: www.tandfonline.com/journals/tstf20

FIRM: an R package for large-scale flexible integration of single-cell data

Shuzhen Ding , Zhou Yu & Jingsi Ming

To cite this article: Shuzhen Ding , Zhou Yu & Jingsi Ming (08 Jun 2026): FIRM: an R package for large-scale flexible integration of single-cell data, Statistical Theory and Related Fields, DOI: 10.1080/24754269.2026.2679093

To link to this article: https://doi.org/10.1080/24754269.2026.2679093

Published online: 08 Jun 2026.

STATISTICAL THEORY AND RELATED FIELDS
https://doi.org/10.1080/24754269.2026.2679093

FIRM: an R package for large-scale flexible integration of single-cell data

Shuzhen Ding[a], Zhou Yu[a] and Jingsi Ming [a,b]

[a]KLATASDS-MOE, School of Statistics, East China Normal University, Shanghai, People's Republic of China; [b]Academy of Statistics and Interdisciplinary Sciences, East China Normal University, Shanghai, People's Republic of China

ABSTRACT

The comparative analysis of single-cell RNA sequencing (scRNA-seq) datasets across various biological conditions, technology platforms and tissue types reveals crucial insights into cellular heterogeneity and tissue architecture. Recent advances in high-resolution technologies have enabled the profiling of gene expression at the single-cell level, yet inherent heterogeneities between platforms and differences in cell type composition make data integration challenging. We present the FIRM R package, which offers a streamlined workflow for the flexible integration of scRNA-seq data using a re-scaling algorithm that accounts for the effects of cell type composition. FIRM achieves accurate mixing of shared cell type identities and superior preservation of the original structure without overcorrection, generating robust integrated datasets for downstream exploration and analysis.

ARTICLE HISTORY
Received 6 March 2026
Revised 14 April 2026
Accepted 20 May 2026

KEYWORDS
Single-cell RNA sequencing; data integration; batch correction; R package

1. Introduction

Single-cell RNA sequencing (scRNA-seq) is being used extensively to measure the mRNA expression of individual cells from deconstructed tissues, organs and even entire organisms to generate cell atlas references. These massive datasets are usually collected from many samples using different scRNA-seq technology platforms, including the popular SMART-Seq2 (SS2) and 10x platforms. Inherent heterogeneities between platforms, tissues and other batch effects make scRNA-seq data difficult to compare and integrate, especially in large-scale cell atlas efforts; yet, accurate integration is essential for gaining deeper insights into cell biology.

FIRM performs flexible integration of single-cell RNA sequencing data for large-scale multi-tissue cell atlas datasets (Ming et al., 2022). It is a re-scaling algorithm that accounts for cell type composition effects, enabling accurate integration across multiple tissues, platforms and experimental batches while preserving the original data structure and avoiding overcorrection. FIRM also supports cell label transfer and serves as a reliable tool for cell atlas research. The FIRM algorithm played a pivotal role in the construction of the first whole-organism single-cell transcriptomic atlas of the mouse lemur (*Tabula Microcebus*), serving a critical function in data integration and batch effect correction (Ezran, Liu, Chang, Ming, Botvinnik, et al., 2025a; Ezran, Liu, Chang, Ming, Guethlein, et al., 2025b). To facilitate user

CONTACT Jingsi Ming jsming@fem.ecnu.edu.cn Academy of Statistics and Interdisciplinary Sciences, East China Normal University, Shanghai 200062, People's Republic of China

access and utilization, we have packaged the `FIRM` algorithm as an R package and released it on Comprehensive R Archive Network (CRAN). Thus the `FIRM` software is readily accessible and straightforward to install.

2. Review of FIRM

`FIRM` is developed based on the open-source C++ libraries `flashpca` (Abraham & Inouye, 2014) and `Spectra` (Qiu, 2021), enabling efficient dimensionality reduction and matrix decomposition computations, and significantly enhancing the performance of large-scale single-cell data processing. Users can install `FIRM` directly from GitHub using the `devtools` package; alternatively, for enhanced convenience, the stable release is readily accessible through CRAN. Here are two alternative ways to install `FIRM`:

```
# from github
library(devtools)
install_github("mingjingsi/FIRM")

# from CRAN
install.packages("FIRM")
```

2.1. Algorithm steps

To facilitate understanding of the `FIRM` integration algorithm, we provide a clear and comprehensive overview of its core computational workflow, with detailed elaboration of the key procedural steps as follows.

(1) *Pre-processing*. We use the gene expression matrix $\mathbf{X}$ for each dataset, where X_{ij} is the gene expression for gene i in cell j. We conduct a pre-processing procedure which includes normalization, scaling and feature selection.

(2) *Dimension Reduction and Clustering*. For each dataset, we perform dimension reduction using PCA (via `Seurat::RunPCA`) and cluster cells (via `Seurat::FindClusters`) based on their PC scores.

(3) *Cluster Alignment via Subsampling*. To align clusters across different datasets while guarding against overcorrection, we employ a subsampling-based strategy. For cluster a in dataset 1, we consider its five nearest clusters in dataset 2 based on their low-dimensional representations after performing PCA on the combined scaled data. Cluster b in dataset 2 is aligned with cluster a in dataset 1 if the following criterion holds: $\|\mathbf{z}_{a\cdot} - \mathbf{z}_{b\cdot}\|^2 < Q_{0.75}(\{\|\mathbf{z}_j - \mathbf{z}_{b\cdot}\|^2\}_{j\in C_b})$, where $\mathbf{z}_{a\cdot}$ and $\mathbf{z}_{b\cdot}$ are the centres of clusters a and b, $Q_{0.75}$ is the 75% quantile and C_b is the cell set in cluster b. For clusters in dataset 1 that have not been aligned, we further perform subsampling to adjust the cluster proportions and check the alignment. If $\frac{|C_a|}{|C_1|} > \frac{|C_b|}{|C_2|}$, where C_1 and C_2 are the sets of cells in datasets 1 and 2, respectively, we subsample the cells in cluster a of dataset 1 to construct a subset $\widehat{C}_{1a}$ so that the proportion matches: $\frac{|\widehat{C}_a|}{|\widehat{C}_{1a}|} = \frac{|C_b|}{|C_2|}$, where $\widehat{C}_a$ is the set of cells in cluster a of subset $\widehat{C}_{1a}$. We then perform re-scaling by $\frac{Y_{ij}}{s_j}, i \in C_1$, where Y_{ij} is the scaled data after the preprocessing procedure and $s_j = \mathrm{sd}(\{Y_{ij}\}_{i\in\widehat{C}_{1a}})$. Based on the re-scaled data, we check the alignment again using the steps described above.

(4) *Re-scaling via Subsampling*. To calculate the scaling factor for effective re-scaling, we perform subsampling for cells in the aligned clusters to obtain subset $\widehat{C}_1$ of dataset 1 and subset $\widehat{C}_2$ of dataset 2, which contain the same types of cells and have the same cell-type proportions as well. We then compute the standard deviation on each of the subsampled datasets for each gene j: $s_{1j} = \mathrm{sd}(\{Y_{ij}\}_{i\in\widehat{C}_1})$, $s_{2j} = \mathrm{sd}(\{Y_{ij}\}_{i\in\widehat{C}_2})$. Finally, we apply global re-scaling to the complete datasets using $\frac{Y_{ij}}{s_{1j}}, i \in C_1$ and $\frac{Y_{ij}}{s_{2j}}, i \in C_2$.

(5) *Data Integration*. Finally, we merge the rescaled datasets through simple concatenation to obtain the integrated data matrix.

This integrated dataset preserves the biological variation within each original dataset while removing technical batch effects, enabling downstream joint analyses such as clustering, visualization and differential expression testing across samples.

2.2. Core components

The `FIRM` package is specifically designed for single-cell RNA sequencing data integration, offering a comprehensive and efficient collection of computational tools. The key functions are introduced as follows.

(1) *Data preprocessing*. `prep_data()` function performs the standard `Seurat` (Butler et al., 2018) workflow preprocessing: normalization, scaling and the selection of highly variable genes.

```
prep_data(counts,           # Raw count matrix
          hvg_genes = 4000) # Target number of genes
```

(2) *Core integration function*. `FIRM()` function performs the unsupervised integration of two single-cell RNA-seq datasets by grid-searching user-specified resolution ranges (res_seq_SS2, res_seq_tenx) to identify the optimal pair that maximizes the mixing metric in the combined PCA space. Users can define a broader or narrower search range based on their computing resources to balance accuracy and computational cost.

```
FIRM(
    SS2,  # Seurat object for reference dataset
    tenx, # Seurat object for query dataset
    hvg1, # HVG gene name vector for SS2
    hvg2, # HVG gene name vector for 10X
    dims, # Number of principal components
    all_genes = FALSE, # Use all genes or not
    res_seq_SS2 = seq(0.1,2,0.1),
    # Screening resolutions for SS2 clustering
    res_seq_tenx = seq(0.1,2,0.1),
    # Screening resolutions for 10X clustering
    coreNum = 1,
    # Number of CPU cores used for parallel screening
    verbose = FALSE, # Quality metrics returned or not
    seed = NULL   # random seed
    )
```

(3) *Label transfer via k-NN (hard assignment)*. `label_trans()` function performs cell type annotation. Specifically, cell type annotation is performed by finding, for every query cell, its *k* nearest reference cells in the joint PCA embedding, followed by majority voting on the labels of these neighbours to assign the final prediction.

```
label_trans(
    integrated, # Seurat object with integrated PCA
       and meta-data columns dataset and annotation.
    ref = "10X", # Character string identifying the
       reference dataset
    query = "SS2", # Character string identifying the
       query dataset
    label_key = "annotation", # Name of the meta-data
       column that indicates celltype
    emb_key = "pca", # Name of the reduction to use
       (default "pca").
    batch_key = "dataset", # Name of the meta-data
       column that indicates dataset/batch
    k = 10, # Number of nearest neighbours to vote
       (default 10).
    seed = NULL # random seed
    )
```

(4) *Label transfer via k-NN (soft assignment / probabilities)*. `label_trans_prob()` differs from `label_trans` in that it returns a probability distribution over all possible labels for each cell, rather than a single definitive assignment. Specifically, for each query cell, the function outputs a vector of posterior probabilities indicating the likelihood of belonging to each reference label. This probabilistic representation preserves uncertainty information, enabling downstream analyses such as confidence scoring, entropy-based quality control, and soft classification of intermediate cell states.

```
# Parameter settings similar to function label_trans()
label_trans_prob(
    integrated,
    ref = "10X",
    query = "SS2",
    label_key = "annotation",
    emb_key = "pca",
    batch_key = "dataset",
    k = 10,
    seed = NULL
    )
```

For additional function information, please use the canonical form https://CRAN.R-project.org/package=FIRMto link to this page, from which comprehensive documentation, including detailed function usage, parameter specifications, and illustrative examples, can be readily accessed.

3. Example

To demonstrate the application of the `FIRM` package, we show how to integrate the SS2 and 10x scRNA-seq datasets from *Tabula Muris*. We integrate the mammary gland scRNA-seq datasets profiled using SS2 and 10x as an example. The raw scRNA-seq datasets can be downloaded from *Tabula Muris*(Schaum et al., 2018). The filtered mammary gland scRNA-seq datasets are available separately at Google Drive.

First, load the required packages for `FIRM`, including `Seurat` for single-cell data manipulation, `RANN` efficiently implements nearest neighbour search and `Rcpp` for efficient computations based on C++, along with the input expression matrices from the SS2 and 10x datasets. Subsequently, perform comprehensive pre-processing using the built-in function `prep_data()`.

```
# Load packages
library(Seurat)
library(FIRM)
library(RANN)

# Load data
load("./Mammary_Gland_counts.RData")
load("./Mammary_Gland_umi.RData")
load("./meta.RData")

SS2_obj <- CreateSeuratObject(counts)
tenx_obj <- CreateSeuratObject(umi)

# Performs a standard Seurat workflow
prep_SS2 <- prep_data(counts)
SS2 <- prep_SS2$Dataset
hvg_SS2 <- prep_SS2$hvg

prep_tenx <- prep_data(umi)
tenx <- prep_tenx$Dataset
hvg_tenx <- prep_tenx$hvg
```

Subsequently, we employ the core function `FIRM()` from the `FIRM` algorithm to execute the integration procedure, thereby achieving the harmonized integration of the SS2 and 10x datasets. This step effectively preserves intrinsic biological heterogeneity across cell types while systematically mitigating technical batch effects arising from distinct sequencing platforms, enabling downstream joint analyses such as unified clustering, visualization, and differential expression testing.

```
# FIRM integration
dims <- 15
coreNum <- 30
integrated_data <- FIRM(SS2, tenx, hvg_SS2, hvg_tenx,
                        dims = dims, coreNum = coreNum)
```

To preserve the FIRM batch correction effects and interface with the Seurat analytical pipeline, we first construct a unified cross-platform count matrix (taking the gene union and imputing missing values with zeros), create a Seurat object and subset to retain successfully aligned genes and cells, perform log-normalization, and directly inject the corrected data of FIRM into the scale.data layer, bypassing the standard scaling procedure.

```
# integrated counts
integrated_counts <- matrix(0,
                            length(union(rownames(SS2_obj),
                                         rownames(tenx_obj))),
                             ncol(SS2_obj) + ncol(tenx_obj))
rownames(integrated_counts) <- union(rownames(SS2_obj),
                                     rownames(tenx_obj))
colnames(integrated_counts) <- c(colnames(SS2_obj),
                                 colnames(tenx_obj))
integrated_counts[rownames(SS2_obj),
                  colnames(SS2_obj)] <-
                  as.matrix(SS2_obj[['RNA']]$counts)
integrated_counts[rownames(tenx_obj),
                  colnames(tenx_obj)] <-
                  as.matrix(tenx_obj[['RNA']]$counts)

# create Seurat object
integrated_FIRM <- CreateSeuratObject(integrated_counts)
integrated_FIRM <-
    integrated_FIRM[rownames(integrated_data),
                    colnames(integrated_data)]

# normalization
integrated_FIRM <- NormalizeData(integrated_FIRM,
                                 verbose = FALSE)

# put the centered integrated data in scaled data
LayerData(integrated_FIRM,
          assay = "RNA",
          layer = "scale.data") <-
          integrated_data - rowMeans(integrated_data)
```

We evaluate the FIRM integration performance through metadata management, dimensionality reduction and visualization validation: adding data source and cell type annotations, selecting common highly variable genes for PCA and UMAP dimensionality reduction, and finally generating grouped plots by platform and annotation to verify batch effect removal and biological signal preservation.

```
# add meta data
integrated_FIRM <-
    AddMetaData(integrated_FIRM, metadata =
                as.character(meta[colnames(integrated_FIRM),
                                  ]$method),
                             col.name = "dataset")
integrated_FIRM <-
    AddMetaData(integrated_FIRM, metadata =
                as.character(meta[colnames(integrated_FIRM),
                                  ]$cell_ontology_class),
                             col.name = "annotation")

# hvg for PCA and visualization
hvg <- intersect(hvg_SS2, hvg_tenx)

# PCA
integrated_FIRM <- RunPCA(integrated_FIRM,
                          features = hvg,
                          npcs = dims,
                          verbose = FALSE)

# UMAP
integrated_FIRM <- RunUMAP(integrated_FIRM,
                           reduction = "pca",
                           dims = 1:dims,
                           umap.method = "uwot",
                           metric = "correlation",
                           verbose = FALSE)
# visualization
DimPlot(integrated_FIRM, reduction = "umap",
        group.by = "dataset")
DimPlot(integrated_FIRM, reduction = "umap",
        group.by = "annotation",
        label = T, repel = T)
```

The UMAP visualization of single-cell data integrated by the FIRM algorithm demonstrates that cells from SS2 and 10x are well-mixed in the reduced-dimensional space (Figure 1). Specifically, cells of the same type cluster together across batches, while distinct cell types remain clearly separated, indicating that FIRM effectively corrects batch effects while successfully preserving biological heterogeneity. FIRM harmonizes datasets through a re-scaling procedure without smoothing the expression of similar cell types across datasets towards each other, so that the relative expression patterns across cells within each dataset can be largely preserved. For this integrated dataset, FIRM achieved the highest local structure metric and ASW compared with Harmony (Korsunsky et al., 2019) and Scanorama (Hie et al., 2019). Although Harmony and Scanorama are slightly higher mixing metric compared to FIRM, considering the trade-off between the mixing metric and local structure metric, FIRM's higher local structure metric suggests that it is more robust than Harmony and Scanorama in avoiding overcorrection. These results fully validate the superior performance of FIRM in cross-platform, cross-batch single-cell data integration, providing a reliable foundation for subsequent cell type annotation and downstream analyses. Moreover, for multi-batch scenarios involving more than two datasets, FIRM adopts a sequential, hierarchical integration protocol: it first integrates the two most similar datasets (based on shared highly variable

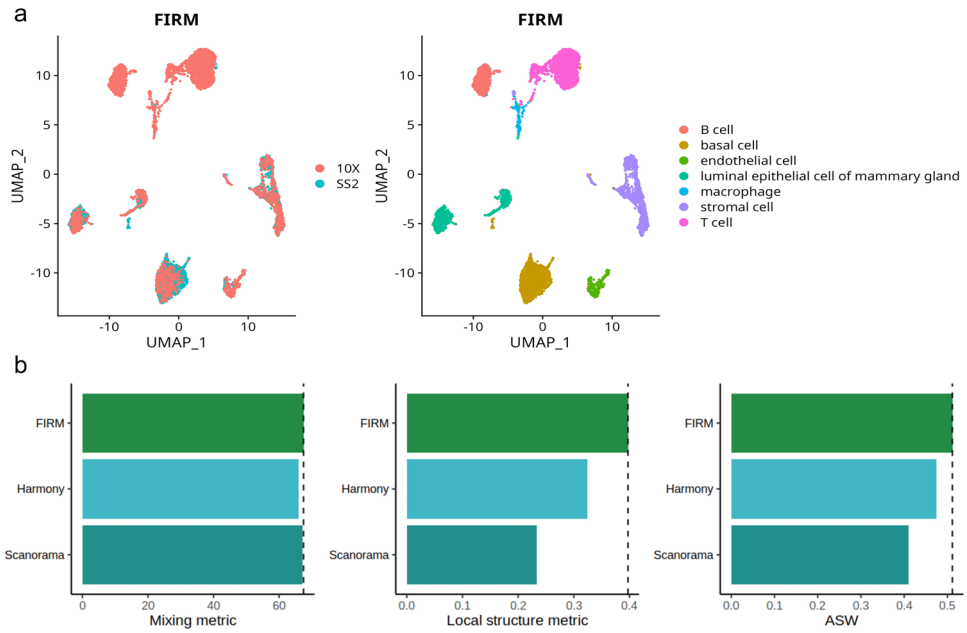

Figure 1. (a) Visualization based on integration results of `FIRM`. UMAP plots are coloured by datasets (left) and cell annotations (right). (b) Metrics for evaluating performance across the three methods on three properties: cell mixing across platforms (Mixing metric), the preservation of within-dataset local structure (Local structure metric) and average silhouette width of annotated subpopulations (ASW). The dashed lines were set at the values for `FIRM` as reference lines.

genes) to establish reliable anchor clusters, and then iteratively adds the remaining datasets to this merged reference, ensuring stable global alignment while maintaining computational efficiency.

4. Conclusion

In summary, the `FIRM` package offers a well-structured and efficient workflow for the integration of scRNA-seq data across multiple tissue types, technology platforms and experimental batches. By specifically accounting for cell type composition effects through re-scaling, `FIRM` achieves accurate mixing of shared cell type identities and superior preservation of the original structure without overcorrection. The package supports multi-dataset integration and generates robust integrated datasets for downstream exploration and analysis in large-scale cell atlas projects. More examples can be found at https://github.com/mingjingsi/FIRM.

While `FIRM` is robust for multi-platform integration, its performance may be constrained in the following scenarios: (1) Integration order: For more than two datasets, `FIRM` uses a hierarchical approach by integrating datasets with high similarity first. This integration order may not be optimal and the most effective. (2) Extreme platform variations: Although optimized for SS2 and 10x, `FIRM`'s re-scaling model might require further validation when applied to emerging technologies with fundamentally different sequencing principles or extreme batch effects not covered by current benchmarks.

Author contributions

CRediT: **Shuzhen Ding:** Software, Visualization, Writing – original draft; **Zhou Yu:** Resources, Supervision; **Jingsi Ming:** Funding acquisition, Software, Supervision, Writing – review & editing

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work is supported in part by funds from the Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China [grant number JYB2025XDXM904], the National Natural Science Foundation of China [grant numbers 12201219, 12371289], Shanghai Key Program of Computational Biology [grant numbers 23JS1400500, 23JS1400800] and the Shanghai Pilot Program for Basic Research [grant number TQ20220105].

ORCID

Jingsi Ming http://orcid.org/0000-0001-7059-4156

References

Abraham, G., & Inouye, M. (2014). Fast principal component analysis of large-scale genome-wide data. *PloS One*, *9*(4), e93766. https://doi.org/10.1371/journal.pone.0093766

Butler, A., Hoffman, P., Smibert, P., Papalexi, E., & Satija, R. (2018). Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nature Biotechnology*, *36*(5), 411–420. https://doi.org/10.1038/nbt.4096

Ezran, C., Liu, S., Chang, S., Ming, J., Botvinnik, O., Penland, L., Tarashansky, A., de Morree, A., Travaglini, K. J., Zhao, J., Wang, G., Hasegawa, K., Sin, H., Sit, R., Okamoto, J., Sinha, R., Zhang, Y., Karanewsky, C. J., Pendleton, J. L., … Krasnow, M. A. (2025a). A molecular cell atlas of mouse lemur, an emerging model primate. *Nature*, *644*(8075), 173–184. https://doi.org/10.1038/s41586-025-09114-8

Ezran, C., Liu, S., Chang, S., Ming, J., Guethlein, L. A., Wang, M. F., Dehghannasiri, R., Olivieri, J., Frank, H. K., Tarashansky, A., Koh, W., Jing, Q., Botvinnik, O., Antony, J., The Tabula Microcebus Consortium, Pisco, A. O., Karkanias, J., Yang, C., Ferrell Jr, J., … Krasnow, M. A. (2025b). Mouse lemur cell atlas informs primate genes, physiology and disease. *Nature*, *644*(8075), 185–196. https://doi.org/10.1038/s41586-025-09114-8

Hie, B., Bryson, B., & Berger, B. (2019). Efficient integration of heterogeneous single-cell transcriptomes using Scanorama. *Nature Biotechnology*, *37*(6), 685–691. https://doi.org/10.1038/s41587-019-0113-3

Korsunsky, I., Millard, N., Fan, J., Slowikowski, K., Zhang, F., Wei, K., Baglaenko, Y., Brenner, M., Loh, P.-R., & Raychaudhuri, S. (2019). Fast, sensitive and accurate integration of single-cell data with Harmony. *Nature Methods*, *16*(12), 1289–1296. https://doi.org/10.1038/s41592-019-0619-0

Ming, J., Lin, Z., Zhao, J., Wan, X., Consortium, T. T. M., Ezran, C., Liu, S., Yang, C., & Wu, A. R. (2022). FIRM: Flexible integration of single-cell RNA-sequencing data for large-scale multi-tissue cell atlas datasets. *Briefings in Bioinformatics*, *23*(5), Article bbac167. https://doi.org/10.1093/bib/bbac167

Qiu, Y. (2021). Sparse eigenvalue computation Toolkit as a redesigned ARPACK. https://spectralib.org/.

Schaum, N., Karkanias, J., Neff, N. F., May, A. P., Quake, S. R., Wyss-Coray, T., Darmanis, S., Batson, J., Botvinnik, O., Chen, M. B., Chen, S., Green, F., Jones, R., Maynard, A., Penland, L., Pisco, A. O., Sit, R. V., Stanley, G. M., Webber, J. T., … Wyss-Coray, T. (2018). Single-cell transcriptomics of 20 mouse organs creates a *Tabula muris*. *Nature*, *562*(7727), 367–372. https://doi.org/10.1038/s41586-018-0590-4