Protocol to benchmark gene expression signature scoring techniques for single-cell RNA sequencing data in cancer

Scoring gene signatures is common for bulk and single-cell RNA sequencing (scRNAseq) data. Here, using cancer as a data model, we describe steps to benchmark signature scoring techniques for scRNAseq data in the context of uneven gene dropouts. These steps include identifying and comparing deregulated signatures, generating gold standard signatures for specificity and sensitivity tests, and simulating the impact of dropouts using down sampling. The protocol provides a framework for benchmarking scRNAseq algorithms in such context.

Publisher’s note: Undertaking any experimental protocol requires adherence to local institutional guidelines for laboratory safety and ethics.
Protocol

Protocol to benchmark gene expression signature scoring techniques for single-cell RNA sequencing data in cancer

Nighat Noureen,1,2,3,* Xiaojing Wang,1,2 and Siyuan Zheng1,2,4,*

1Greehey Children’s Cancer Research Institute, UT Health San Antonio, San Antonio, TX 78229, USA
2Department of Population Health Sciences, UT Health San Antonio, San Antonio, TX 78229, USA
3Technical contact
4Lead contact
*Correspondence: noureen@uthscsa.edu (N.N.), zhengs3@uthscsa.edu (S.Z.)
https://doi.org/10.1016/j.xpro.2022.101877

SUMMARY

Scoring gene signatures is common for bulk and single-cell RNA sequencing (scRNAseq) data. Here, using cancer as a data model, we describe steps to benchmark signature scoring techniques for scRNAseq data in the context of uneven gene dropouts. These steps include identifying and comparing deregulated signatures, generating gold standard signatures for specificity and sensitivity tests, and simulating the impact of dropouts using down sampling. The protocol provides a framework for benchmarking scRNAseq algorithms in such context. For complete details on the use and execution of this protocol, please refer to Noureen et al. (2022).1

BEFORE YOU BEGIN

Download single cell RNAseq datasets and gene signatures

© Timing: 5–10 min

1. Download scRNAseq datasets listed in the key resources table. The protocol uses human cancer datasets that consist of both tumor and normal cells. For details, refer to Supplementary file 1 from Noureen et al.1
2. Download gene signatures (or gene sets, terms used interchangeably hereafter) from Molecular Signatures Database (MSigDB)4 in GMT format. The download link is provided in the key resources table. We used C2 (curated gene sets), C3 (regulatory gene sets), and H (Hallmark gene sets) for this protocol.

Install tools/packages

© Timing: 2–3 h

3. The hardware requirements for running this protocol are provided in the key resources table.
4. This protocol utilizes the R environment (version 4.0.3) for statistical computing and graphics.
5. R packages used are listed under the “software and algorithms” section of the key resources table. Packages can be found using the links provided in the identifier column.

△ CRITICAL: This protocol uses parallel processing. It is advised to follow the protocol on a multi-thread computer; otherwise, it may take longer than as indicated.
KEY RESOURCES TABLE

| REAGENT or RESOURCE | SOURCE | IDENTIFIER |
|---------------------|--------|------------|
| Software and algorithms |        |            |
| R (v4.0.3) |        | https://www.r-project.org/ |
| JASMINE_V1 | Noureen et al. | https://github.com/NNoureen/JASMINE |
| AUCell_1.16.0 | Aibar et al. | https://bioconductor.org/packages/release/bioc/html/AUCell.html |
| SCSE | Pont et al. | https://github.com/NNoureen/BenchmarkingProtocol |
| GSVA_1.42.0 | Hänzelmann et al. | https://www.bioconductor.org/packages/release/bioc/html/GSVA.html |
| ssGSEA_1.42.0 | Hänzelmann et al. | https://www.bioconductor.org/packages/release/bioc/html/GSVA.html |
| Seurat_4.1.0 | Satija et al. | https://cran.r-project.org/web/packages/Seurat/index.html |
| MAST_1.20.0 | Finak et al. | https://www.bioconductor.org/packages/release/bioc/html/MAST.html |
| effectsizesize_0.6.0.1 | Ben-Shachar et al. | https://cran.r-project.org/web/packages/effectsize/effectsize.pdf |
| ComplexHeatmap_2.10.0 | Gu et al. | https://bioconductor.org/packages/release/bioc/html/ComplexHeatmap.html |
| scuttle_1.4.0 | McCarthy et al. | https://bioconductor.org/packages/release/bioc/html/scuttle.html |
| GSA_1.03.1 | Efron et al. | https://cran.r-project.org/web/packages/GSA/index.html |
| data_table_1.14.2 | Dowle et al. | https://CRAN.R-project.org/package=data.table |
| dplyr_1.0.8 | Wickham et al. | https://CRAN.R-project.org/package=dplyr |
| ggplot2_3.3.5 | Wickham et al. | https://ggplot2.tidyverse.org/ |
| doParallel_1.0.17 | Danie et al. | https://cran.r-project.org/web/packages/doParallel/index.html |
| ssGSEA dropouts | Noureen et al. | https://github.com/NNoureen/JASMINE/ssGSEA_scRNAseq_test.R |

Other

Computing Platform:

- A desktop with memory 32 GB or higher is recommended for large data preprocessing and data visualization. This protocol was performed on Windows 10 system with Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20 GHz Processor and 32 GB memory
- A high-performance computing cluster – 32 GB memory or higher with 8-core processor or higher processing cores is recommended for parallel computing (for signature scoring computations)

scRNAseq data sets

Public repositories

https://www.ncbi.nlm.nih.gov/geo/
https://www.ebi.ac.uk/gxa/sc/home
https://singlecell.broadinstitute.org/single_cell
Glioma Data Link: - https://singlecell.broadinstitute.org/single_cell/study/SCP50/single-cell-ra-seq-analysis-of-astrocytoma

Gene sets

Subramanian et al. | http://www.gsea-msigdb.org/gsea/msigdb

STEP-BY-STEP METHOD DETAILS

Timing and details for each step in this protocol are based on one dataset¹⁸ used in Noureen et al.¹
The same steps were followed for the remaining nine datasets used in the study.¹
Data import, quality control and preprocessing

 Timing: 5–10 min

This step is to prepare data, including quality control and preprocessing.

1. Download gene sets (C2, C3 and hallmarks) from MSigDB.
2. Download glioma scRNAseq ‘IDH_A_processed_data_portal.txt’ and phenotype assignments ‘IDH_A_cell_type_assignment_portal_v2.txt’ data, from the Single Cell Portal by Broad Institute.
3. Load R and related packages as mentioned in the key resources table. 
4. Load gene expression and metadata into R.

Note: Here we use `read.table` and `fread` functions as the data is available in text format. Here other file reading functions like `read.csv`, `read_xlsx` or `readRDS` can be used depending on file formats.

5. The glioma data is already preprocessed, so we used it as is.

 Alternatives: In case of raw data counts, especially for 10× datasets used in Noureen et al., we preprocessed them by removing non-expressed genes and then applied regularized negative binomial regression implemented in Seurat using `SCTransform` function as follows.

```r
> GEdata = read.table('IDH_A_processed_data_portal.txt',sep='\t', head=T)  
> row.names(GEdata) = GEdata$Gene ## Assigning gene names to rows 
> GEdata = GEdata[, -1] ## Removing first column from data containing gene names 
> saveRDS(GEdata, 'IDAstrocytoma_GE_20210311.RDS') ## Saving data in RDS format 
> Metadata = fread('IDH_A_cell_type_assignment_portal_v2.txt') ## Reading Metadata 
> colnames(Metadata) =c('SampleID','Cluster','SubCluster','TIndex')
```

△ CRITICAL: Check quality control measures including mitochondrial content, number of empty cells and duplicates in your scRNAseq data before applying normalization procedure. For details of quality control, refer to Seurat toolkit.

Gene count differences between tumor and normal cells

 Timing: 2–3 min

This step is to calculate differences in gene counts between tumor and normal cells.

6. Calculate genes expressed per cell in scRNAseq data using following command. Use the GEdata from step 4.

```r
> dataset <- CreateSeuratObject(counts = GEdata, project = 'scRNA_Practice', min.cells = 1)  
#### Creating Seurat Object 
> dataset <- SCTransform(dataset) ## Binomial Regression function
```
7. Combine metadata from step 4 with gene counts from step 6 using cell type/Sample IDs.

```r
> GE_Total = merge (Metadata, GeneNumber, by='SampleID')
> GE_Total$Cluster = ifelse(GE_Total$Cluster == malignant, "malignant", "Normal")
> write.table(GE_Total, 'IDHAstrocytoma_Metadata.txt', sep = '\t', quote = FALSE, row.names = FALSE)
```

8. Average the number of expressed genes per phenotype using data from step 7 with the `mean` function.

```r
> Mean_Genes_Per_CT = tapply(GE_Total$nGenes, GE_Total$Cluster, mean)
```

Optional: Variations in the number of expressed genes among different normal cell populations can be checked based on availability of phenotypic data. In cancer scRNAseq datasets, many cancer-related cells such as cancer-associated macrophages have higher gene counts than other normal cells, an indication that these cells may be co-opted by cancer cells. For details, check Figure S1 of Main Figure 1.

Signature scoring and tumor/normal comparisons

© Timing: 1 week

This step is to score gene expression signatures for scRNAseq datasets. These scores can be used to determine identities and cellular properties of single cells. Because the number of cells and gene signatures are high, we use parallel processing to accelerate the process. The timing of this step is reflected based on the glioma dataset, which comprises around 6,000 cells.

9. Load the following R packages: 1) AUCell, 2) GSA, 3) doParallel, 4) GSVA, 5) effectsize.
10. Use `doParallel` to call parallel processing for calculating signature scores for the five signature scoring tools (AUCell, JASMINE, SCSE, GSVA, ssGSEA; see key resources table).
11. For each tool, first load scRNAseq data into R using `readRDS` command. The RDS data object is saved in step 4 by removing gene names columns from the raw data.

```r
> GEdata = readRDS('IDHAstrocytoma_GE_20210311.RDS')
```

12. Load the gene sets (C2, C3 and hallmarks) into R via `GSA.read.gmt` function from GSA package.

```r
> Genesets1 <- GSA.read.gmt('h.all.v7.2.symbols.gmt')
> GSize = length(Genesets1$genesets)
> Genesets2 <- GSA.read.gmt('c2.all.v7.2.symbols.gmt')
```
13. Initiate multiple threads with the `registerDoParallel` function and implement parallel processing by using `dopar` function.

   **Note:** We used 20 threads in this step for each signature scoring tool, except GSVA for which we used 200 threads. The following codes show users how they can implement parallel processing. They have been implemented in the scripts called in step 14.

   ```r
   > registerDoParallel(20) ### initiate multiple cores
   > foreach (k=1:GSsize,.combine = rbind,.errorhandling = "remove") %dopar% {
   ## This code is implemented in the following files ```AUCell.r'', ```GSVA.r'', ```ssGSEA.r'', ```SCSE.r'', and ```JASMINE.r'''. So, it is recommended to use this code through these files
   }
   > source('AUCell.r')
   > source('GSVA.r')
   > source('ssGSEA.r')
   > source('SCSE.r')
   > source('JASMINE.r')
   ```

14. Call each tool to calculate and save the signature scores.
   a. Use `AUCell_buildRankings` and `AUCell_calcAUC` functions from the `AUCell` package for `AUCell`.
   b. Use `gsva` function from the `GSVA` package to execute `ssGSEA` and `GSVA`.
   c. Call `SCSE` function from https://github.com/NNoureen/BenchmarkingProtocol.
   d. Call `JASMINE` function from https://github.com/NNoureen/JASMINE.

   **Note:** We have implemented `SCSE` function in R based on the formula reported in the paper because the tool was originally provided as a web server.

   ▲ CRITICAL: To run each code file using the source function, place all the data files and gene set files in your current working directory, or users need to specify data file paths in the codes.

15. Combine signature scores from step 14 using the following code.

   ```r
   > source('CombineScores.r')
   ```

   **Note:** The total size of the gene sets used in step 14 is around 10,000; therefore we divided the scoring for each tool into 3 steps. The results are combined to get one file per tool.
16. Divide the signature scores compiled in step 15 into tumor and normal cells using metadata file ‘IDHAstrocytoma_Metadata.txt’ saved in step 7.

```r
> Metadata = read.table('IDHAstrocytoma_Metadata.txt',sep='\t',head=T)
> TumorCells = Metadata$SampleID[which(Metadata$Clusters == "malignant")]
> NormalCells = Metadata$SampleID[which(Metadata$Clusters == "Normal")]
```

17. Calculate effect size (ES) for each gene set using the tumor and normal cells scores from steps 15 and 16.

```r
> source('EScalculation.r')
```

**Note:** Effect size is used to measure score differences between tumor and normal cells while controlling for score variance. Effect size is calculated using `cohens_d` function from the `effectsize` package. The code file used in this step will use combined data scores saved in step 15. The code snippet in step 16 is included in this code file.

18. Identify up regulated ($ES \geq 1$) and down regulated ($ES \leq -1$) gene sets from step 17 for each signature scoring tool using the following codes.

**Note:** These codes should be repeated for each tool. The file ‘ES_AllMethods.txt’ used to read ES values have been saved in step 17 and is used here for further processing. These cutoffs are arbitrary and thus can be tailored for different datasets.

```r
> EScheck = read.table('ES_AllMethods.txt',sep='\t',head=T)
> ES_check = EScheck[which(EScheck$GSsize > 20),]
> ES_method_Up = length(which(EScheck$ES >= 1))
> ES_method_Dn = length(which(EScheck$ES <= -1))
> ESUp_Percentage = ES_method_Up/nrow(EScheck) * 100
> ESDn_Percentage = ES_method_Dn/nrow(EScheck) * 100
```

19. Associate gene set sizes with ES for each tool using `cor.test` function. We used Spearman method, but Pearson or other methods can be used. For details see Figure 1C from Noureen et al.¹

**Note:** We suggest using gene signatures as separate lists if the number of signatures is high. In this case, we used C2, C3 and hallmarks as 3 lists for all tools except GSVA where we divided them into 10 lists as they contain around 10,000 signatures in total. In our calculation we removed gene sets with less than 20 genes.

```r
> EScheck = read.table('ES_AllMethods.txt',sep='\t',head=T)
> GS_ES_Corr = cor.test(EScheck$GS, EScheck$GSsize, method= "spearman")
```
CRITICAL: GSVA is computationally less efficient than other tools used in this protocol. To obtain results within a reasonable time window, use 200 or more threads for big datasets and large lists of gene sets. In our experience, it took 5 days to complete the glioma data even when using 200 threads in parallel.

Detection sensitivity

Timing: 47 h

This step is to generate gold standard gene sets of various sizes to benchmark detection sensitivity for scRNAseq signature scoring. Gold standard gene sets consist of differentially expressed genes, either up or down, plus non-differentially expressed genes as noises. This step involves parallel processing using 10 threads, but more can be used to further reduce running time. GSVA is dropped from this and subsequent steps because of its low running speed.

20. Load Seurat and MAST packages in R.
21. Create Seurat object of scRNAseq data using `CreateSeuratObject` function.
22. Add metadata to the Seurat object using `AddMetaData` function.
23. Use `FindMarkers` function in Seurat specifying `test.use = "MAST"` to identify the DEGs. Provide tumor and normal groups information to the function.

Note: Identification of DEGs is required to generate the gold standard up and down regulated gene sets. In this step, we use tumors cells as the first group and normal cells as the second to identify the DEGs.

```r
> GEdata = readRDS('IDHAstrocytoma_GE_20210311.RDS')
> Metadata = read.table('IDHAstrocytoma_Metadata.txt',sep='	',head=T)
> mydata <- CreateSeuratObject(GEdata)
> SampleID = Metadata$SampleID
> Samples = Metadata$Clusters
> names(Samples)= SampleID
> mydata <- AddMetaData(object = mydata, metadata = Samples, col.name = "Samples")
> mydata@active.ident = Samples
> Markers = FindMarkers(mydata, ident.1 = "malignant", ident.2 = "Normal", test.use = "MAST")
> write.table(Markers, 'IDHastrocytoma_MAST_DEGS_15March2021.txt',sep=’\t’, quote=FALSE)
```

24. Generate gold standard up and downregulated gene sets using the following code files.

```r
> source('RandomGSets_DN.r') ## generating down regulated gold standard gene sets
> source('RandomGSets_Up.r') ## generating up regulated gold standard gene sets
```

Note: We used logFC>0 for generation of up regulated genes and logFC<0 for down regulated genes. We generated gene sets of 5 sizes (n=50,100,150,200,300). For each size, we set 5 noise levels to 0%, 20%, 40%, 60%, and 80%. For each noise-gene set size combination,
we randomly generated 200 gene sets. This in total produced 5000 gene sets per up and down regulated category.

25. Calculate and save signature scores for scRNAseq data using gold standard up and down regulated gene sets from step 24. Use the following files to call the signature scoring functions for random gene sets.

```r
> source('AU_RandomGeneSets.r')
> source('GSVA_RandomGeneSets.r')
> source('ssGSEA_RandomGeneSets.r')
> source('SCSE_RandomGeneSets.r')
> source('JASMINE_RandomGeneSets.r')
```

**Note:** Users can reuse this code for any other gene signatures, e.g., gold standard gene sets or gene sets of their interest. But for users’ ease we have re-implemented the code with gold standard signatures.

26. Follow step 17 to calculate ES for signature scores generated in step 25.

```r
> source('EScalculation.r')
```

**Note:** To use the R file, users need to change the input file for each tool. Since this is ES calculation of random gene sets, signature scores saved in step 25 are used as input here for ES calculation.

27. Calculate the percentage of up and down regulated gene sets at each noise level for all tools using commands illustrated in step 18. The input of this step is the output of step 26.

**Note:** The percentage represents the recovery rate of up and down regulated gene sets. For details, see Figures 2A and 2B in Noureen et al.¹

**Detection specificity**

© Timing: 1.5 h

This step is to down sample scRNAseq data to calculate detection specificity. Down sampling creates nearly identical expression profiles at a lower coverage, thus allowing for specificity tests. In this step we use 100 cells from the scRNAseq dataset. We use 10 threads for parallel computing.

28. Choose desired number of cells from a scRNAseq dataset. In this demonstration, we use 100 tumor cells.

29. Down sample scRNAseq data using `downsampleMatrix` function from R package scuttle.

```r
> mydata = readRDS('IDHAstrocytoma_GE_20210311.RDS')
> dataDN_Samp = downsampleMatrix(mydata, 0.5, bycol = TRUE)
```
Note: Adjust down sampling percentage by the parameter “prop”. We used 50% in this case, but this parameter can be adjusted to simulate different coverage levels.

30. Scale back the down sampled data to ensure equal coverage as the original data.

```r
> dataDN_Samp = apply(dataDN_Samp,2, function(x) (x/sum(x)) *1000000) ### CPM normalization
```

31. Calculate signature scores for down sampled data set as described in steps 11–15.
32. Calculate ES by comparing signature scores of down sampled data from step 31 and original data from step 15 for the 100 cells.
33. Identify up and down gene sets following step 18 using ES calculated in step 32.

Consensus calling

© Timing: 20 min

Consensus calling is often used to manage outputs from multiple tools. Here, we use consensus calling to benchmark the signature scoring techniques developed for scRNAseq data. To identify consensus up or down regulated gene sets, we use gene sets identified in step 18. We limit the consensus calling to single cell-based tools because of the performance bias by bulk-based tools.

34. Identify consensus gene sets that are called by at least two tools in either direction based on the same ES criteria.

```r
> source('ConsensusCalling.r')
```

35. Use the consensus to identify true positives (TP), false positives (FP), true negatives (TN) and true positives (TP). The total number of signatures is denoted as N. Use the ConsensusSummary.r code file to compute these numbers.

```r
> source('ConsensusSummary.r')
```

36. Calculate sensitivity, specificity and accuracy of single cell tools using the following equations implemented in Sensitivity_vs_Specificity.r file.

```r
> Sensitivity = TP/ (TP + FN)
> Specificity = TN/ (TN + FP)
> Accuracy = (TN + TP)/ N
> source('Sensitivity_vs_Specificity.r')
```

Note: For details, see Figure 2D and related Supplement 4 from Noureen et al. Time complexity and memory usage are simple metrics to determine computational efficiency of a tool. In analysis of large datasets like scRNAseq, it is important to design time and space efficient tools.
Impact of dropouts on single cell scoring

© Timing: 6 h

Dropouts are the main reason for scRNAseq data sparsity. Dropouts arise from a low amount of input materials in sequencing and stochastic transcription. To check the effect of dropouts on scRNAseq signature scoring, we will simulate dropouts by down sampling scRNAseq data at different levels. The difference in signature scores between the down sampled cells and the original cells reflects the impact of dropouts on each tool.

37. Run down sampling experiment at different rates using the following code.

```r
> data = readRDS('IDHAstrocytoma_GE_20210311.RDS')
> dataDN_Samp20 = downsampleMatrix(data, 0.2, bycol = TRUE)
> dataDN_Samp40 = downsampleMatrix(data, 0.4, bycol = TRUE)
> dataDN_Samp60 = downsampleMatrix(data, 0.6, bycol = TRUE)
> dataDN_Samp80 = downsampleMatrix(data, 0.8, bycol = TRUE)
```

**Note:** The down sampling rates can be customized. We use 4 different rates (20%, 40%, 60% and 80%) for this protocol. A lower down sampling rate creates more dropouts.

38. Calculate signature scores for the 4 down sampled scRNAseq datasets generated in step 37 using details mentioned in steps 11–15.
39. Use signature scores of each down sampled data from step 38 and calculate ES by comparing with scores of the 100 cells derived from the original data from step 15.
40. Calculate percentages of up and down gene sets from step 39 for each down sampling rate and scoring tool following step 18.

**Note:** These percentages should be negligible regardless of the down sampling rate, because no differentially expressed signatures are expected. For details check Figure 3A and related Figures S1 and S2 from Noureen et al.

41. To evaluate the effect of dropouts on ssGSEA, check the details in Noureen et al. and related code file on our GitHub repository (ssGSEA_scRNAseq_test.R).

**EXPECTED OUTCOMES**

The step-by-step protocol describes the benchmarking of gene expression signature scoring techniques in scRNAseq data. We summarize the expected outcome format for each step in Table 1.

**LIMITATIONS**

The protocol has been tested with datasets ranging from 500 to 6000 cells. Some scRNAseq platforms can generate data with much higher coverage, and thus, results may vary when the protocol is tested in those datasets. We used 10 cancer datasets; more tests should be done in a non-cancer context such as stem cells versus differentiated cells. We tested five tools, but more could be added such as Seurat’s `AddModuleScore` function.
Table 1. Summary of expected outputs from each step in the protocol

| Key step                        | Steps number | Output format | Output file |
|---------------------------------|--------------|---------------|-------------|
| Gene count differences between tumor and normal cells | 6–8          | These steps will generate a table containing 2 columns, 1st with cell IDs and 2nd with total number of genes expressed per cell. The table will be further used to calculate the average number of genes for tumor and normal cells. | A text file. |
| Signature scoring and tumor/normal comparisons | 9–15         | These steps will generate a table with cell IDs along the columns and names of gene sets along the rows. Each entry in the table represents the signature score calculated by a tool. In total 5 tables will be generated, 1 for each tool. | An Rdata object. |
|                                  | 16–18        | These steps will generate a table containing gene sets along the rows and signature scoring tools along the columns. Each entry in the table represents ES score per gene set. Based on the ES values, percentage of up and down regulated gene sets will be calculated. | An Rdata object. |
|                                  | 19           | Correlation between gene set size and ES will be calculated in this step. In total 5 numbers will be generated per dataset. | A text file. |
| Detection Sensitivity            | 20–23        | These steps will generate a table containing differentially expressed genes along the rows. The columns will contain log2FC, p-value and FDR for each DEG. | A text file. |
|                                  | 24           | This step will generate up and down gold standard gene sets. | 2 .gmt files |
|                                  | 25           | This step will score gene sets generated in step 24. The output format is same as described in steps 9–15. | An Rdata object per tool. |
|                                  | 26–27        | These steps will generate same output as in steps 16–18 for data generated in step 25. | An Rdata file. |
| Detection Specificity            | 28–30        | These steps will generate a gene expression matrix containing genes along the rows and cell IDs along the columns. Each entry in the matrix will represent gene expression level per cell. | An Rdata object. |
|                                  | 31           | This step will generate same output as in steps 9–15 using the data generated in steps 28–30. | An Rdata object per tool. |
|                                  | 32–33        | These steps will generate same output as in steps 16–18 for the data generated in step 31. | An Rdata object. |
| Consensus calling                | 34–36        | These steps will generate a table containing names of signature scoring tools in column 1, ES direction in column 2, sensitivity, specificity and accuracy information in columns 3–5. | An Rdata object. |
| Impact of dropouts on single cell scoring | 37           | This step will generate 4 scRNAseq expression matrices with different coverages. | An Rdata object. |
|                                  | 38           | This step will generate same output as in steps 9–15 for the datasets generated in step 37. | An Rdata object per tool. |
|                                  | 39–40        | These steps will generate the same output as in steps 16–18 for the data generated in step 38. | An Rdata object. |

TROUBLESHOOTING

Problem 1
The output of FindMarkers function in some cases does not clearly separate the 2 phenotypes (step 23).
Potential solution
Over-normalization can reduce biological signals in the data. Make sure the data is normalized but not over normalized. In Seurat, normalization is implemented in function SCTransform.

Problem 2
Signature scoring tools gives an error during execution (step 14).

Potential solution
If the genes present in the set are not present in scRNAseq data, then it runs into execution error. Therefore, it is suggested to check the presence of gene set in the data before executing the function.

Problem 3
Random sampling by sample function generates a new list each time it is used for different noise levels. This creates inconsistency in results during comparisons.

Potential solution
In order to reuse the same list of up and down regulated genes generated by random sampling, it is suggested to generate the list once and save it. Reuse the saved list across all noise level comparisons. Alternatively, use set.seed function before sample. This would help avoid inconsistency in comparative analysis.

Problem 4
Parallel processing using dopar function does not save the results properly.

Potential solution
Saving the results from dopar function in a file after every iteration does not append the file correctly. It is therefore suggested to save the results in a dataframe for a gene set list and write the dataframe at the end in a file.

Problem 5
GSVA or ssGSEA signature scores are not saved properly.

Potential solution
If the scRNAseq data size is large and number of gene sets to be scored are more than 5000, then GSVA and ssGSEA signature scores are not saved properly during parallel processing. It is therefore, suggested to divide the gene set lists into smaller subsets and re-submit multiple jobs for execution. This would make execution better and would avoid loss of results for some gene sets.

RESOURCE AVAILABILITY

Lead contact
Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Siyuan Zheng (zhengs3@uthscsa.edu).

Materials availability
This study did not generate new materials.

Data and code availability
This paper refers to the existing publicly available datasets. Details are mentioned in the step-by-step procedure. Sample codes to run each tool and calculate effect size are provided on GitHub at https://github.com/NNoureen/BenchmarkingProtocol,2 https://doi.org/10.5281/ZENODO.7230690. JASMINE source code is available at https://github.com/NNoureen/JASMINE,3 https://doi.org/10.5281/ZENODO.7245400.
Acknowledgments

This work was supported by CPRIT (RR170055 to Z.S.). N.N. was supported by a CPRIT postdoctoral fellowship award (RP170345).

Author Contributions

N.N. prepared materials and codes and wrote the protocol. X.W. prepared materials and wrote the protocol. S.Z. supervised the project and wrote the protocol. All authors reviewed and approved the manuscript.

Declaration of Interests

The authors declare no competing interests.

References

1. Noureen, N., Ye, Z., Chen, Y., Wang, X., and Zheng, S. (2022). Single-cell gene expression signatures across scRNAseq datasets. Nucleic Acids Res. 50, 7199–7209. https://doi.org/10.1093/nar/gkab850.

2. Noureen, N. and Zheng, S. (2022). Benchmarking gene expression signature scoring methods for single cell RNA sequencing data. GitHub. https://doi.org/10.5281/ZENODO.7230690.

3. Subramanian, A., Tamayo, P., Mootha, V.K., Mukherjee, S., Ebert, B.L., Gillette, M.A., Paulovich, A., Pomeroy, S.L., Golub, T.R., Lander, E.S., and Mesirov, J.P. (2005). Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proc. Natl. Acad. Sci. USA 102, 15545–15550. https://doi.org/10.1073/pnas.0506580102.

4. Aabar, S., Gonzalez-Blas, C.B., Moerman, T., Huynh-Thu, V.A., Imrichova, H., Hulselmans, G., Rambow, F., Marine, J.-C., Geurts, P., Aerts, J., et al. (2017). SCENIC: single-cell regulatory network inference and clustering. Nat. Methods 14, 1083–1086. https://doi.org/10.1038/nmeth.4463.

5. Pont, F., Tosolini, M., and Fournié, J.J. (2019). Single-Cell Signature Explorer for comprehensive visualization of single cell signatures across scRNAseq datasets. Nucleic Acids Res. 47, e133. https://doi.org/10.1093/nar/gkz601.

6. Hänzelmann, S., Castelo, R., and Guinney, J. (2013). GSEA: gene set variation analysis for microarray and RNA-Seq data. BMC Bioinf. 14, 7. https://doi.org/10.1186/1471-2105-14-7.

7. Satija, R., Farrell, J.A., Gennert, D., Schier, A.F., and Regev, A. (2015). Spatial reconstruction of single-cell gene expression data. Nat. Biotechnol. 33, 495–502. https://doi.org/10.1038/nbt.3192.

8. Finak, G., McDavid, A., Yajima, M., Deng, J., Gershuk, V., Shalek, A.K., Sligher, C.K., Miller, H.W., McInraith, M.J., Pric, M., et al. (2015). MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. Genome Biol. 16, 278. https://doi.org/10.1186/s13059-015-0844-5.

9. Ben-Shachar, M., Lüdecke, D., and Makowski, D. (2020). Effectsize: estimation of effect size Indices and standardized parameters. J. Open Source Softw. 5, 2815. https://doi.org/10.21105/joss.02815.

10. Gu, Z., Eils, R., and Schlesner, M. (2016). Complex heatmaps reveal patterns and correlations in multidimensional genomic data. Bioinformatics 32, 2847–2849. https://doi.org/10.1093/bioinformatics/btw313.

11. McCarthy, D.J., Campbell, K.R., Lunt, A.T.L., and Wills, Q.F. (2017). Scater: pre-processing, quality control, normalization and visualization of single-cell RNA-seq data in R. Bioinformatics 33, 1179–1186. https://doi.org/10.1093/bioinformatics/btw777.

12. Efron, B., and Tibshirani, R. (2007). On testing the significance of sets of genes. Ann. Appl. Stat. 1. https://doi.org/10.1214/07-AOAS101.

13. Dowle, M., and Srinivasan, A. (2022). dplyr: A Grammar of Data Manipulation. https://dplyr.tidyverse.org.

14. Wickham, H., Francois, R., Henry, L., and Muller, K. (2022). ggplot2: Elegant Graphics for Data Analysis (Springer-Verlag New York).

15. Wickham, H. (2016). ggplot2: Elegant Graphics for Data Analysis. https://ggplot2.org.

16. Weston, S. (2022). Foreach Parallel Adaptor for the “Parallel” Package. https://github.com/RevolutionAnalytics/doparallel.

17. Finak, G., McDavid, A., Yajima, M., Deng, J., Gershuk, V., Shalek, A.K., Sligher, C.K., Miller, H.W., McInraith, M.J., Pric, M., et al. (2015). MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. Genome Biol. 16, 278. https://doi.org/10.1186/s13059-015-0844-5.

18. Venteicher, A.S., Tirosh, I., Hebert, C., Yuhak, K., Neftel, C., Fiblin, M.G., Hovestadt, V., Escalante, L.E., Shaw, M.L., Rodman, C., et al. (2017). Decoupling genetics, lineages, and microenvironment in IDH-mutant gliomas by single-cell RNA-seq. Science 355, eaai8478. https://doi.org/10.1126/science.aai8478.

19. Hao, Y., Hao, S., Andersen-Nissen, E., Mauck, W.M., Zheng, S., Butler, A., Lee, M.J., Wilk, A.J., Darby, C., Zager, M., et al. (2021). Integrated analysis of multicomplex single-cell data. Cell 184, 3573–3587.e29. https://doi.org/10.1016/j.cell.2021.04.048.

20. Noureen, N., Wu, S., Lv, Y., Yang, J., Alfred, Yung, W.K., Geffond, J., Wang, X., Koul, D., Ludlow, A., and Zheng, S. (2021). Integrated analysis of telomerase enzymatic activity unravels an association with cancer stemness and proliferation. Nat. Commun. 12, 139. https://doi.org/10.1038/s41467-020-20474-9.

21. Bao, R., Huang, L., Andrade, J., Tan, W., Kibbe, W.A., Jiang, H., and Feng, G. (2014). Review of current methods, applications, and data management for the bioinformatics analysis of whole exome sequencing. Cancer Inform. 13, 67–82. https://doi.org/10.4137/CIN.S13779.

22. Elliott, K., Bailey, M.H., Saksena, G., Covington, K.R., Kandot, C., Stewart, C., Hess, J., Ma, S., Chioti, K.E., McLellan, M., et al. (2018). Scalable open science approach for mutation calling of tumor exomes using multiple genomic pipelines. Cell Syst. 6, 271–281.e7. https://doi.org/10.1016/j.cels.2018.03.002.

23. Zheng, S. (2017). Benchmarking: contexts and details matter. Genome Biol. 18, 129. https://doi.org/10.1186/s13059-017-1258-3.

24. Hicks, S.C., Townes, F.W., Teng, M., and Irizarry, R.A. (2018). Missing data and technical variability in single-cell RNA-sequencing experiments. Biostatistics 19, 562–578. https://doi.org/10.1093/biostatistics/kxx053.

25. Bao, R., Huang, L., Andrade, J., Tan, W., Kibbe, W.A., Jiang, H., and Feng, G. (2014). Review of current methods, applications, and data management for the bioinformatics analysis of whole exome sequencing. Cancer Inform. 13, 67–82. https://doi.org/10.4137/CIN.S13779.