SOFTWARE TOOL ARTICLE

clustifyr: an R package for automated single-cell RNA sequencing cluster classification [version 1; peer review: 2 approved with reservations]

Rui Fu1, Austin E. Gillen1, Ryan M. Sheridan1, Chengzhe Tian2, Michelle Daya3, Yue Hao4, Jay R. Hesselberth1,5, Kent A. Riemondy1

1RNA Bioscience Initiative, University of Colorado School of Medicine, Aurora, CO, 80045, USA
2Department of Biochemistry, University of Colorado Boulder, Boulder, CO, 80303, USA
3Biomedical Informatics & Personalized Medicine, University of Colorado Anschutz Medical Campus, Aurora, CO, 80045, USA
4Bioinformatics Research Center, North Carolina State University, Raleigh, NC, 27695, USA
5Department of Biochemistry and Molecular Genetics, University of Colorado School of Medicine, Aurora, CO, 80045, USA

Abstract
Assignment of cell types from single-cell RNA sequencing (scRNA-seq) data remains a time-consuming and error-prone process. Current packages for identity assignment use limited types of reference data and often have rigid data structure requirements. We developed the clustifyr R package to leverage several external data types, including gene expression profiles to assign likely cell types using data from scRNA-seq, bulk RNA-seq, microarray expression data, or signature gene lists. We benchmark various parameters of a correlation-based approach and implement gene list enrichment methods. clustifyr is a lightweight and effective cell-type assignment tool developed for compatibility with various scRNA-seq analysis workflows. clustifyr is publicly available at https://github.com/rnabioco/clustifyr

Keywords
Single-cell RNA sequencing, cell type classification, gene expression profile, R package

Open Peer Review

Reviewer Status
Invited Reviewers

version 1
01 Apr 2020

Invited Reviewers

1 Keegan Korthauer, BC Children's Hospital Research Institute, Vancouver, Canada
University of British Columbia, Vancouver, Canada

2 Kamil Slowikowski, Massachusetts General Hospital, Boston, USA

Any reports and responses or comments on the article can be found at the end of the article.

This article is included in the Bioconductor gateway.

This article is included in the RPackage gateway.
Corresponding author: Kent A. Riemondy (kent.riemondy@cuanschutz.edu)

Author roles: Fu R: Conceptualization, Software, Writing – Original Draft Preparation, Writing – Review & Editing; Gillen AE: Conceptualization, Software, Writing – Review & Editing; Sheridan RM: Software, Writing – Review & Editing; Tian C: Software, Writing – Review & Editing; Daya M: Software, Writing – Review & Editing; Hao Y: Software, Writing – Review & Editing; Hesselberth JR: Software, Supervision, Writing – Review & Editing; Riemondy KA: Conceptualization, Software, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: RNA Bioscience Initiative at the University of Colorado School of Medicine and the National Institutes of Health [R35 GM119550 to J.R.H.]. This work was in part completed during the NIH sponsored Rocky Mountain Genomics HackCon (2018) hosted by the Biofrontiers Department at the University of Colorado at Boulder.

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2020 Fu R et al. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Fu R, Gillen AE, Sheridan RM et al. clustifyr: an R package for automated single-cell RNA sequencing cluster classification [version 1; peer review: 2 approved with reservations] F1000Research 2020, 9:223

First published: 01 Apr 2020, 9:223 https://doi.org/10.12688/f1000research.22969.1
Introduction

Single-cell mRNA sequencing (scRNA-seq) promises to deliver elevated understanding of cellular mechanisms, cell heterogeneity within tissue, and developmental transitions. A key challenge in scRNA-seq data analysis is the identification of cell types from single-cell transcriptomes. Manual inspection of the expression patterns from a small number of marker genes is still standard practice, which is cumbersome and frequently inaccurate. Unfortunately, current implementations of scRNA-seq suffer from several limitations that further compound the problem of cell type identification. First, only RNA levels are measured, which may not correlate with cell surface marker or gene expression signatures identified through other experimental techniques. Second, due to the low capture rate of RNAs, low expressing genes may face detection problems regardless of sequencing depth. Many previously established markers of disease or developmental processes suffer from this issue, such as transcription factors. On the data analysis front, over or under-clustering can generate cluster markers that are uninformative for cell type labeling. In addition, cluster markers that are unrecognizable to an investigator may indicate potentially interesting unexpected cell types, but can be very intimidating to interpret.

For these reasons, investigators struggle to integrate scRNA-seq into their studies due to the challenges of confidently identifying previously characterized or novel cell populations. Formatted data-driven approaches for assigning cell type labels to clusters greatly aid researchers in interrogating scRNA-seq experiments. Currently, multiple cell type assignment packages exist but they are specifically tailored towards input types or workflows. As more and more approaches to the classification problem are introduced, benchmarking performance and compatibility to sequencing platforms and analysis pipelines becomes increasingly important.

We developed the R package clustifyr, a lightweight and flexible tool that leverages a wide range of prior knowledge of cell types to pinpoint target cells of interest or assign general cell identities to difficult-to-annotate clusters. Here, we demonstrate its basic usage and applications with transcriptomic information of external datasets and/or signature gene profiles, to explore and quantify likely cell types. The clustifyr package is built with compatibility and ease-of-use in mind to support other popular scRNA-seq tools and formats.

Methods

Implementation

clustifyr requires query and reference data in the form of raw or normalized expression matrices, corresponding metadata tables, and a list of variable genes (Figure 1).

```r
library(clustifyr)
pbmc_matrix_small[1:5, 1:5] # query matrix of normalized scRNA-seq counts
cbmc_ref[1:5, 1:5] # reference matrix of expression for each cell type
pbmc_meta[1:5, ] # query meta-data data.frame containing cell clusters
length(pbmc_markers_M3Drop$Gene) # vector of variable genes
```

clustifyr adopts correlation-based methods to find reference transcriptomes with the highest similarity to query cluster expression profiles, defaulting to Spearman ranked correlation, with options to use Pearson, Kendall, or Cosine correlation instead if desired. clustify() will return a matrix of correlation coefficients for each cell type and cluster, with the row names corresponding to the query cluster number and column names as the reference cell types.

```r
res <- clustify(
  input = pbmc_matrix_small,
  metadata = pbmc_meta,
  cluster_col = "seurat_clusters", # column in meta.data with clusters
  ref_mat = cbmc_ref,
  query_genes = pbmc_markers_M3Drop$Gene
)
```

```r
res[1:5, 1:5]
#>    B CD14+ Mono CD16+ Mono CD34+ CD4 T
#> 0 0.4700038 0.5033242 0.5188112 0.6012423 0.7909705
#> 1 0.4850570 0.4900953 0.5232810 0.5884319 0.7366543
#> 2 0.5814309 0.9289886 0.8927613 0.6394140 0.5258430
#> 3 0.8609621 0.4663520 0.5686564 0.6429193 0.4698687
#> 4 0.2814882 0.1888232 0.2506101 0.4140560 0.6125503
```
Query clusters are assigned cell types to the highest correlated reference cell type, with an automatic or manual cutoff threshold for query clusters dissimilar to all available reference cell types, to be labeled as “unassigned”.

```r
res2 <- cor_to_call(
  cor_mat = res, # matrix of correlation coefficients
  cluster_col = "seurat_clusters", # column in meta.data with clusters
  threshold = 0.5
)
```

To better integrate with standard workflows that involve S3/S4 R objects, methods for clustifyr are written to directly recognize Seurat\textsuperscript{14} (v2 and v3) and SingleCellExperiment\textsuperscript{15} objects, retrieve the required information, and reinsert classification results back into an output object. A more general wrapper is also included for compatibility with other common data structures, and can be easily extended to new object types. This approach also has the added benefit of forgoing certain calculations such as variable gene selection or clustering, which may already be stored within input objects.

```r
res <- clustify(
  input = sce_small, # an SCE object
  ref_mat = cbmc_ref, # matrix of expression for each cell type
  cluster_col = "cell_type1", # column in meta.data with clusters
  obj_out=TRUE # output SCE object with cell type
)
```
SingleCellExperiment::colData(res)[1:10, c("type", "r")]

#> DataFrame with 10 rows and 2 columns
#>              type                  r
#>        <character>         <numeric>
#> AZ_A1         pDCs 0.814336567702192
#> AZ_A10       Eryth 0.665800619720566
#> AZ_A11        pDCs 0.682088309107356
#> AZ_A12       Eryth 0.665800619720566
#> AZ_A2            B 0.634114583333333
#> AZ_A3         pDCs 0.814336567702192
#> AZ_A4         pDCs 0.814336567702192
#> AZ_A5           NK 0.655407634437123
#> AZ_A6         pDCs 0.682088309107356
#> AZ_A7         pDCs 0.71424223704931

res <- clustify(
  input = s_small3, # a Seurat object
  ref_mat = cbmc_ref, # matrix of expression for each cell type
  cluster_col = "RNA_snn_res.1", # name of column in meta.data containing cell clusters
  obj_out = TRUE # output Seurat object with cell type inserted as "type" column
)

res@meta.data[1:5, ]

#>                   orig.identnCount_RNA nFeature_RNA RNA_snn_res.0.8
#> ATGCCAGAACGACT SeuratProject        70           47               0
#> CATGGCCTGTGCA SeuratProject        85           52               0
#> GAACCTGATGAC SeuratProject        87           50               1
#> TGACTGATTCTCA SeuratProject       127           56               0
#> AGTCAGACTGACA SeuratProject       173           53               0

In the absence of suitable reference data (i.e. RNA-seq or microarray expression matrices), clustifyr can build scRNA-seq reference data by averaging per-cell expression data for each cluster, to generate a transcriptomic snapshot. Direct reference-building from SingleCellExperiment or Seurat objects is supported as well.

new_ref_matrix <- average_clusters(
  mat = pbmc_matrix_small,
  metadata = pbmc_meta$classified, # or use metadata = pbmc_meta, cluster_col = "classified"
  if_log=TRUE# whether the expression matrix is already log transformed
)

Data exploration plotting functions, for dimensional reduction scatter plots and heatmaps, are extended from ggplot2 and ComplexHeatmap packages, featuring colorblind-friendly default colors. Simple gene list-based methods (clustify_lists()) for sanity checks on positive and negative markers, via gene list enrichment or calculation of percentage detection by cluster, are implemented as well.
**Parameters**

**Correlation method.** We benchmarked clustifyr against a suite of comparable datasets, PBMCbench\textsuperscript{16}, generated from two PBMC samples using multiple scRNA-seq methods. Notably, for each reference dataset cross-referenced to other samples, clustifyr achieved a median F1-score of above 0.94 using Spearman ranked correlation (Figure 2A). Other correlation methods are on par or slightly worse at cross-platform classifications,

**Figure 2. Parameter considerations for clustifyr.** A) Comparison of accuracy of different correlation methods for classifying across platforms using the PBMCbench dataset. B) Heatmap showing correlation coefficients between query cell types and the reference cell types. Clusters with correlation < 0.50 are assigned as Neg.Cell by clustifyr. C) Comparison of classification power with and without feature selection. D) An assessment of the accuracy of using single or multiple averaged profiles as reference cell types was conducted using the PBMCbench test set. The number of reference expression profiles to generate for each cell type is determined by the number of cells in the cluster \( n \), and the sub-clustering power argument \( x \), with the formula \( n^x \). E) Accuracy and performance were assessed with decreasing query cluster cell numbers using the PBMCbench test.
which is expected based on the nature of ranked vs unranked methods. We therefore selected Spearman as the default method in clustifyr, with other methods also available, as well as a wrapper function to find consensus identities across available correlation methods (call_consensus()).

**Correlation minimum cutoff.** Recognition of missing reference cell types, so as to avoid misclassification, is another point of great interest in the field. From general usage of clustifyr, we find using a minimum correlation cutoff of 0.5 or 0.4 is generally satisfactory. Alternatively, the cutoff threshold can be determined heuristically using 0.8 * highest correlation coefficient among the clusters. One example is shown in Figure 2B, using PBMC rejection benchmark data modified by the SciBet package. Megakaryocytes were removed from reference data, and labeled as “neg.cells” for ground truth in test data. clustifyr analysis found the “neg.cells” to be dissimilar to all available reference cell types, and hence left as “unassigned” under the default minimum threshold cutoff. Next, we applied clustifyr to a series of increasingly challenging datasets from the scRNAseq_Benchmark13 unseen population rejection test. Without the corresponding cell type references, 57.5% of T cells were rejected and unassigned. When only CD4+ references were removed, 28.2% of test CD4+ T cells were rejected and unassigned. clustifyr was unable to reject CD4+/CD45RO+ memory T cells, mislabeling them as CD4+/CD25 T Reg instead when the exact reference was unavailable. However, these misclassifications are also observed with other classification tools benchmarked in the scRNAseq_Benchmark study.

**Variable gene selection.** As the core function of clustifyr is ranked correlation, feature selection to focus on highly variable genes is critical. In Figure 2C, we compare correlation coefficients using all detected genes (>10,000) vs feature selection by the package M3Drop. A basic level of feature selection, e.g. M3Drop, Seurat VST (default takes top 2,000), or simply 1,000 genes with highest variance in the reference data, is sufficient to classify the pancreatic cells. In the case of other cell type mixtures, especially ones without complete knowledge of the expected cell types, clustering and feature selection will be of greater importance. clustifyr does not provide novel clustering or feature selection methods on its own, but instead is built to maintain flexibility to incorporate methods from other, and future, packages. We view these questions as a fast-moving fields, and hope to benefit from new advances, while keeping the general clustifyr framework intact.

**Subclustering.** For scRNA-seq reference data, matrices are built by averaging per-cell expression data for each cluster, to generate a transcriptomic snapshot similar to bulk RNA-seq or microarray data. An additional argument to subcluster the reference dataset clusters is also available, to generate more than one expression profile per reference cell type. The number of subclusters for each reference cell type is dependent on the number of cells in the cluster (n), and the sub-clustering power argument (x), following the formula n^x. However, this approach does not improve classification in the PBMCbench data (Figure 2D). We envision its utility would greatly depend on the granularity of the clustering in the reference dataset.

**Cells per cluster.** After testing a general reference set built from the Mouse Cell Atlas to be of high accuracy in classification of the Tabula Muris data, we subsampled the query data (Figure 2E). As expected, with further downsampling of the number of cells in each query cluster, we observe decreased accuracy. Yet, even at 15 cells per tested cluster, clustifyr still performed well, with a further increase in speed. Based on these results, we set the default parameters in clustifyr to exclude or warn users of classification on clusters containing less than 10 cells. In addition, an intentional overclustering and classification function based on k-means clustering (overcluster_test()) is implemented in clustifyr for exploration of clustering quality.

**Benchmarking**

Using clustifyr, peripheral blood mononuclear cell (PBMC) clusters from the Seurat PBMC 3k tutorial are correctly labeled using either bulk-RNA seq references generated from the ImmGen database, processed microarray data of purified cell types, or previously annotated scRNA-seq results from the Seurat CBMC CITE-seq tutorial (Figure 3).

To assess the performance of clustifyr, we used the Tabula Muris dataset, which contains data generated from 12 matching tissues using both 10x 3’ end seq (“drop”) and SmartSeq2 (“facs”) platforms. Using references built from “facs” Seurat objects, we attempted to assign cell type identities to clusters in “drop” Seurat objects. In benchmarking results, clustifyr is comparably accurate versus other automated classification packages. Cross-platform comparisons are inherently more difficult, and the approach used by clustifyr is aimed at being platform- and normalization-agnostic. Mean runtime, including both reference building and test data classification, in Tabular Muris classifications was ~ 1 second if the required variable gene list is extracted from the query Seurat object. Alternatively, variable genes can be recalculated by other
Figure 3. clustifyr can utilize multiple reference data types. UMAP projections showing the ground truth cell types, or cell types called by clustifyr using different data sources (microarray or bulk RNA-seq data from purified cell types, or scRNA-seq data).

Methods such as M3Drop\(^2\), to reach similar results. Correlation-based clustifyr classification performed better than hypergeometric-based gene list enrichment as implemented in clustify_lists.

For scalability benchmarking, we adapted scRNAseq_Benchmark subsampling for the Tabula Muris dataset. Once again, clustifyr is accurate and efficient, compared to other developed methods (Figure 4B). We also reached similarly satisfactory results in scRNA-seq brain transcriptome data from mouse and human samples, as detailed by scRNAseq_Benchmark pipeline using data from the Allen Institute Brain Atlas\(^1\) (Figure 4C).

clustifyr was tested against scmap v1.8.0\(^8\), SingleR v1.0.1\(^1\), Seurat v3.1.1\(^1\), latest GitHub versions of ACTINN\(^1\) and scPred\(^1\), and SVM as implemented in python3 scikit-learn v0.19.1\(^\text{24}\). scRNA-seq Tabula Muris data was downloaded as seuratV2 objects. Human pancreas data was downloaded as SCE objects. In all instances, to mimic the usage case of clustifyr, clustering and dimension reduction projections are acquired from available metadata, in lieu of new analysis.

An R script was modified to benchmark clustifyr following the approach and datasets of scRNAseq_Benchmark\(^1\), using M3Drop\(^\text{23}\) variable gene selection for every test. R code used for benchmarking, and
preprocessing of other datasets, in the form of matrices and tables, are documented in R scripts available in the clustifyr and clustifyrdata GitHub repositories.

Operation
clustifyr is distributed as part of the Bioconductor R package repository and is compatible with Mac OS X, Windows, and major Linux operating systems. Package dependencies and system requirements are documented in the clustifyr Bioconductor repository.

Conclusions
We present a flexible and lightweight R package for cluster identity assignment. The tool bridges various forms of prior knowledge and scRNA-seq analysis. Reference sources can include scRNA-seq data with cell types assigned (or average expression per cell type, which can be stored at much smaller file sizes), sorted bulk RNA-seq, and microarray data. clustifyr, with minimal package dependencies, is compatible with a number of standard analysis workflows such as Seurat or Bioconductor, without requiring the user to perform the error-prone process of converting to a new scRNA-seq data structure, and can be easily extended to incorporate other data storage object types. clustifyr is designed to perform classification after previous steps of analysis by other informatics tools. Therefore, it relies on, and is agnostic to, common external packages for cell clustering and variable feature selection. We envision it to be compatible with all current and future scRNA-seq processing, clustering, and

![Figure 4. clustifyr accurately and rapidly annotates cell types. A) Accuracy and run-time of classifications generated by clustifyr or existing methods using the Tabula Muris to benchmark cell type classifications across sequencing platforms. Each point represents a different tissue comparison. B) Performance comparison of clustifyr to existing methods by subsampling the Tabula Muris dataset. Cell numbers are listed in the facet labels. C) Performance comparison of clustifyr to existing methods testing against Allen Institute Brain Atlas data containing 34 cell types.](image)
marker gene discovery workflows. Benchmarking reveals the package performs well in mapping cluster identity across different scRNA-seq platforms and experimental types. As we and others observe\(^2\), novel algorithms may not be necessary for cell type classification, at least within the current limitations of sequencing technology and our broadstroke understanding of cell “types”. Rather, the generation of community curated reference databases is likely to be critical for reproducible annotation of cell types in scRNA-seq datasets.

On the user end, clustifyr is built with simple out-of-the-box wrapper functions, sensible defaults, yet also extensive options for more experienced users. Instead of building an additional single-cell-specific data structure, or requiring specific scRNA-seq pipeline packages, it simply handles basic data.frames (tables) and matrices (Figure 1). Input query data and reference data are intentionally kept in expression matrix form for maximum flexibility, ease-of-use, and ease-of-interpretation. Also, by operating on predefined clusters, clustifyr has high scalability and minimal resource requirements on large datasets. Using per-cluster expression averages results in rapid classification. However, cell-type annotation accuracy is therefore heavily reliant on appropriate selection of the number of clusters. Users are therefore encouraged to explore cell type annotations derived from multiple clustering settings. Additionally, assigning cell types using discrete clusters may not be appropriate for datasets with continuous cellular transitions such as developmental processes, which are more suited to trajectory inference analysis methods. As an alternative, clustifyr also supports per-cell annotation, however the runtime is greatly increased and the accuracy of the cell type classifications are decreased due to the sparsity of scRNA-seq datasets, and requires a consensus aggregation step across multiple cells to obtain reliable cell type annotations.

To further improve the user experience, clustifyr provides easy-to-extend implementations to identify and extract data from established scRNA-seq object formats, such as Seurat\(^14\), SingleCellExperiment\(^15\), URD\(^4\), and CellDataSet (Monocle)\(^26\). Available in flexible wrapper functions, both reference building and new classification can be directly achieved through scRNA-seq objects at hand, without going through format conversions or manual extraction. The wrappers can also be expanded to other single cell RNA-seq object types, including the HDF5-backed loom objects, as well as other data types generated by CITE-seq and similar experiments\(^27\). Tutorials are documented online to help users integrate clustifyr into their workflows with these and other bioinformatics software.

**Software availability**

clustifyr is available from Bioconductor: [https://bioconductor.org/packages/devel/bioc/html/clustifyr.html](https://bioconductor.org/packages/devel/bioc/html/clustifyr.html)

Up-to-date source code, tutorials, and prebuilt references available from: [https://github.com/rnabioco/clustifyr](https://github.com/rnabioco/clustifyr)

Archived source code as at time of publication: [https://doi.org/10.5281/zenodo.3718588](https://doi.org/10.5281/zenodo.3718588)

Data used in examples and additional prebuilt references available from: [https://github.com/rnabioco/clustifyrdata](https://github.com/rnabioco/clustifyrdata)

License: MIT

**Data availability**

Original raw data used in benchmarking is available from the following sources:

| Dataset | Source |
|---------|--------|
| PBMC 3k Seurat V3 object | https://www.dropbox.com/s/63gnlw45j7cje8/pbmc3k_final.rds?dl=0 |
| CBMC CITE-seq | Accession number, GSE100866: ftp://ftp.ncbi.nlm.nih.gov/geo/series/GSE100nnn/GSE100866/suppl/GSE100866_CBMC_8K_13AB_10X-RNA_umi.csv.gz |
| Hematopoiesis microarray data | Accession number, GSE24759: https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE24759 |
| Tabula Muris as Seurat V2 objects | https://figshare.com/projects/Tabula_Muris_Transcriptomic_characterization_of_20_organisms_and_tissues_from_Mus_musculus_at_single_cell_resolution/27733 |
| Mouse Cell Atlas | https://doi.org/10.6084/m9.figshare.5435866.v8 |
| Pancreatic scRNA-seq as SingleCellExperiment objects | https://hemberg-lab.github.io/scRNA_seq.datasets/ |
A step-by-step workflow for M3Drop: dropout-based feature selection for scRNA-seq data: a tutorial. Mol Syst Biol. 2019; 15(6): e8746.

Pubmed Abstract | Publisher Full Text | Free Full Text

3. Luecken MD, Theis FJ: Current best practices in single-cell RNA-seq analysis: a tutorial. Mol Syst Biol. 2019; 15(6): e8746.

Pubmed Abstract | Publisher Full Text | Free Full Text

4. Farrell JA, Wang Y, Riesenfeld SJ, et al.: Single-cell reconstruction of developmental trajectories during zebrafish embryogenesis. Science. 2018; 360(6392): pii: eaa3131.

Pubmed Abstract | Publisher Full Text | Free Full Text

5. Tabula Muris Consortium; Overall coordination; Logistical coordination: et al.: Single-cell transcriptomics of 20 mouse organs creates a Tabula Muris. Nature. 2018; 552(7687): 367–72.

Pubmed Abstract | Publisher Full Text | Free Full Text

6. Kiselev VY, Andrews TB, Hemberg M: Challenges in unsupervised clustering of single-cell RNA-seq data. Nat Rev Genet. 2015; 20(5): 273–82.

Pubmed Abstract | Publisher Full Text | Free Full Text

7. Vallejo CA, Riesso D, Schaalje A, et al.: Normalizing single-cell RNA sequencing data: challenges and opportunities. Nat Methods. 2017; 14(6): 565–71.

Pubmed Abstract | Publisher Full Text | Free Full Text

8. Kiselev VY, Yu A, Hemberg M: scmap: projection of single-cell RNA-seq data across data sets. Nat Methods. 2018; 15(5): 359–62.

Pubmed Abstract | Publisher Full Text | Free Full Text

9. Aran D, Looney AP, Liu L, et al.: Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. Nat Immunol. 2019; 20(2): 163–72.

Pubmed Abstract | Publisher Full Text | Free Full Text

10. Pliner HA, Shendure J, Trapnell C: Supervised classification enables rapid annotation of cell atlases. Nat Methods. 2019; 16(10): 983–6.

Pubmed Abstract | Publisher Full Text | Free Full Text

11. Ma F, Pellegrini M: ACTINN: automated identification of cell types in single cell RNA sequencing. Bioinformatics. 2020; 36(2): 533–8.

Pubmed Abstract | Publisher Full Text | Free Full Text

12. Alquicir-Hernandez J, Sathe A, Ji HP, et al.: scPred: accurate supervised method for cell-type classification from single-cell RNA-seq data. Genome Biol. 2019; 20(1): 264.

Pubmed Abstract | Publisher Full Text | Free Full Text

13. Abdolali T, Michielsen L, Catts D, et al.: A comparison of automatic cell identification methods for single-cell RNA sequencing data. Genome Biol. 2019; 20(1): 194.

Pubmed Abstract | Publisher Full Text | Free Full Text

14. Butler A, Hoffman P, Smyth P, et al.: Integrating single-cell transcriptomic data across different conditions, technologies, and species. Nat Biotechnol. 2018; 36(5): 411–20.

Pubmed Abstract | Publisher Full Text | Free Full Text

15. Lun AT, McCarthy DJ, Marioni JC: A step-by-step workflow for low-level analysis of single-cell RNA-seq data with Bioconductor [version 2]: peer review: 3 approved, 2 approved with reservations]. F1000Res. 2016; 5: 2122.

Pubmed Abstract | Publisher Full Text | Free Full Text

16. Ding J, Adiconis X, Simmons SK, et al.: Systematic comparative analysis of single cell RNA-sequencing methods. bioRxiv: 2019; 632216.

Publisher Full Text

17. Li C, Liu B, Kang B, et al.: SciBet: a fast classifier for cell type identification using single cell RNA sequencing data. bioRxiv: 2019; 645258.

Publisher Full Text

18. Düö A, Robinson MD, Soneson C: A systematic performance evaluation of clustering methods for single-cell RNA-seq data [version 2]: peer review: 2 approved]. F1000Res. 2018; 7: 1141.

Pubmed Abstract | Publisher Full Text | Free Full Text

19. Soneson C, Robinson MD: Bias, robustness and scalability in single-cell differential expression analysis. Nat Methods. 2018; 15(4): 255–61.

Pubmed Abstract | Publisher Full Text | Free Full Text

20. Han X, Wang P, Zhou Y, et al.: Mapping the Mouse Cell Atlas by Microwell-Seq. Cell. 2018; 172(5): 1091–7.

Pubmed Abstract | Publisher Full Text | Free Full Text

21. Heng TS, Painter MW, Immunological Genome Project Consortium: The immunological Genome Project: networks of gene expression in immune cells. Nat Immunol. 2008; 9(10): 1091–4.

Pubmed Abstract | Publisher Full Text | Free Full Text

22. Novenstern N, Subramanian A, Lawton LN, et al.: Densely interconnected transcriptional circuits control cell states in human hematopoiesis. Cell. 2011; 144(2): 296–309.

Pubmed Abstract | Publisher Full Text | Free Full Text

23. Andrews TS, Hemberg M: M3Drop: dropout-based feature selection for scRNASeq. Bioinformatics. 2019; 35(16): 2865–7.

Pubmed Abstract | Publisher Full Text | Free Full Text

24. Pedregosa F, Varoquaux G, Gramfort A, et al.: scikit-learn: Machine Learning in Python. J Mach Learn Res. 2011; 12: 2825–30.

Reference Source

25. Köhler ND, Bültem M, Theis FJ: Deep learning does not outperform classical machine learning for cell-type annotation. bioRxiv: 2019 [cited 2020 Jan 28]; 653907.

Publisher Full Text

26. Cao J, Spielmann M, Oiu X, et al.: The single-cell transcriptional landscape of mammalian organogenesis. Nature. 2019; 566(7745): 496–502.

Pubmed Abstract | Publisher Full Text | Free Full Text

27. Richer AL, Riemondy KA, Hardie L, et al.: Simultaneous measurement of biochemical phenotypes and gene expression in single cells. bioRxiv: 2019; 820233.

Publisher Full Text

28. Fu R, Gillen A, Sheridan R, et al.: mabioclusterify 0.99.7 (Version 0.99.7). Zenodo. 2020. http://www.doi.org/10.5281/zenodo.3718588

Acknowledgements

A previous version of this article is available on bioRxiv: https://doi.org/10.1101/855064.
Open Peer Review

Current Peer Review Status:  

© 2020 Slowikowski K. This is an open access peer review report distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Kamil Slowikowski  
Massachusetts General Hospital, Boston, MA, USA

The authors describe an R package for annotating cell clusters in scRNA-seq datasets. Specifically, the package implements code for computing correlations between the columns of two data matrices. They show that high correlations between unknown cell clusters in the first data matrix and annotated cell types in the second matrix can be used to label the unknown cell clusters. They try varying parameters and show the effects on the results, and they also benchmark the time and accuracy compared to other packages designed to annotate scRNA-seq data.

Details of the code, methods, and analyses are partly provided. Some details seem to be missing (e.g. the functionality for gene lists).

The conclusions about the tool and its performance are partly supported by the findings presented in the article. Some terms such as "medF1-score" and "accuracy" are left undefined, and some results omit some methods (Figure 4A has different methods than B or C). Readers may have difficulty understanding the specific questions that were asked and what results are shown.

Main comments:
1. The clarity of the manuscript can be increased by adding more verbose details about all analyses. Please consider expanding details about each question, the approach, the datasets used, and the results.

2. Please consider adding a table describing the reference datasets used in this article, just like the one shown on one of your GitHub repositories. This should help to summarize which datasets were used for the analyses in this article.

Comments about specific parts of the manuscript are below. Excerpts from the article are shown in "quoted italics" after a bullet point, and my comments are shown directly below the bullet point.
"A key challenge in scRNA-seq data analysis is the identification of cell types from single-cell transcriptomes. Manual inspection of the expression patterns from a small number of marker genes is still standard practice, which is cumbersome and frequently inaccurate."

Do we know the accuracy by manual inspection? Is there a reference for this? In the absence of evidence, you might consider weakening the statement to say "may be inaccurate" rather than "is cumbersome and frequently inaccurate". You might consider that many scRNA-seq experiments are done for the purpose of discovering new cell types that have not been well-described in previous published datasets. In this setting, manual inspection is necessary and automated analyses could be inaccurate or misleading.

"Currently, multiple cell type assignment packages exist but they are specifically tailored towards input types or workflows."

Please consider naming and describing each method that will be compared to clustifyr in this manuscript, so the reader can assess how the methodology of clustifyr compares to other methods. Which methods are "specifically tailored towards input types or workflows"? Could you give an example to help the reader understand this claim?

**Suggested improvements for Figure 1:**
- In Figure 1, you might consider showing the dimensions of the inputs and outputs. This might help the reader to understand how they relate to each other.
- Should the query and reference data be counts? CPM? Or Log2(CPM + 1)? You might consider elaborating on this.

**Suggested improvements for Figure 2:**
- Please consider rotating Figure 2A, D, and E 90 degrees clockwise to improve legibility.
- Please consider limiting the axes ranges to the data instead of using the range [0, 1].
- Please consider increasing all font sizes in all panels in all figures, including titles, legends, axis text, etc. Some readers might need larger sizes to see clearly.
- Please consider changing the title to "All genes (n = 10,000)" and "M3Drop variable genes (n = 1,000)" in Figure 2C, so we have some sense of the number of genes used to generate each heatmap.
- Please consider showing a graphical representation of the experiment setup for this figure. What is the reference? What is the query? What are their dimensions? What is the main question in this analysis?
- One way to enhance clarity is to add descriptive titles to every figure in every panel (e.g. "Testing different correlation statistics", etc.).
- Please consider adding more details to the legend text for Figure 2 to help readers understand exactly what experiment has been done, what data was used, and what result is shown.
- In Figure 2C, it seems that the y-axis and x-axis have been swapped by mistake. I see that the y-axis is labeled "ground truth cell type" but it includes "unclassified". I would expect the category "unclassified" to appear in the "called cell type" axis, but not in the "ground truth cell type" axis. Are
the axes swapped or are they correct? Could you please clarify?

• In Figure 2E, what does the color indicate? Is it the power argument "n^x" or something else?

• The reader may be wondering:
  • How many query cells did you use?
  • How many clusters were in the query dataset? How many cells per cluster?
  • How many reference datasets were used?
  • How many clusters were in the reference dataset?
  • Were the query and reference datasets acquired from the same tissue sample or were they completely independent and unrelated?

In the section “Subclustering”, please consider adding more details to help the reader avoid misunderstandings. What exactly is the "sub-clustering power argument (x)"? Please consider giving a concrete example to help the reader understand this section. Please consider creating a new figure that helps the reader to understand the "subcluster()" functionality.

What is the PBMCbench data? Is this the same data as mentioned in the section "Correlation minimum cutoff"?

In the section "Cells per cluster", you might consider introducing the dataset, then introducing the question that is being addressed, and finally reporting the results. What is the number (15, 8, 4)? Is the "Mouse Cell Atlas" the same as the "Tabula Muris"? Were these mouse datasets used in the previous sections? The reader might benefit from an introduction of these datasets.

Suggested improvements for Figure 3:
• Please consider adding labels "A", "B", "C", "D" to mark each of the four panels, so they can be referenced clearly.

• Please consider using the same name consistently in the text and the figure titles. For example, the figure says "Bulk RNA-seq reference data" but the text says "ImmGen database". The reader might better understand the results if the same label were used in both places instead of using two different labels for the same thing.

• Please consider including the identifiers for readers who wish to find these datasets and download them. For example, if the datasets are available on NCBI GEO, please consider including the accession numbers directly in the legend text, or in a table. Check to see if any other database provides an accession number. If an accession number is not available, please consider providing the DOI for a publication or a URL for a website that provides the data. By the way, if any data you are using is not deposited to a permanent repository, please consider uploading this data to a permanent repository (e.g. Figshare).

In the section describing Figure 4A, please consider these suggested changes:
• Please explain what is "clustifyr", "clustifyr_lists", and "clustifyr_m3drop".
• How was feature selection performed for each analysis in Figure 4A?
• What is the strategy used by scmap?
• What is the strategy used by "Seurat"?
• What is the strategy used by "SingleR"?
How is clustifyr similar or different?

This section says "Correlation-based clustifyr classification performed better than hypergeometric-based
gene list enrichment as implemented in clustify_lists." Please consider explaining the "clustify_lists"
algorithm in detail and also consider sharing the quantification of the performance of each approach so
the reader can interpret the claim "performed better". Also consider elaborating on "performed better".

What is "scRNAseq_Benchmark subsampling"? Could you elaborate on what this is and why it was used?

**Suggested improvements for Figure 4:**

- Please consider including an overview schematic to help the reader understand which datasets
  were used for each result.
- Please define "accuracy". What is the algorithm for computing this number?
- Please define "medF1-score". What is the algorithm for computing this number?
- For the lower half of panel B, please consider using a format similar to the one in Figure 2B from
  Kiselev et al. (2018). For example, please use a log10 axis for time, so readers can see the
  difference between methods.
- Why is "medF1-score" used for Figure 4C and "accuracy" for Figure 4B?

Why does Figure 4A have 6 methods, Figure 4B have 5 methods, and Figure 4C have 3 methods? Is it
possible to include all 6 methods for all panels? Could you please comment on the reasons for excluding
or including methods in each analysis?

- "As we and others observe, novel algorithms may not be necessary for cell type classification, at
  least within the current limitations of sequencing technology and our broadstroke understanding of
  cell "types". Rather, the generation of community curated reference databases is likely to be critical
  for reproducible annotation of cell types in scRNA-seq datasets."

I agree that a community curated reference database would be a valuable contribution to the field. You
might consider creating a table or other type of descriptive listing that helps the reader to understand all of
the references that were used in this article. Consider including tissue source, healthy or disease status,
number of cells and genes, technology used for the assay, DOI, data URL, NCBI GEO accession, or any
other details that the reader might find helpful.

Thank you for providing a GitHub repository with data files! Please also consider sharing the same data in
compressed plain text format (e.g. "file.tsv.gz"). In addition to GitHub, please consider using a specialty
service that is funded for the purpose of permanently archiving research data such as NIH Figshare (https://nih.figshare.com). There are other options (Zenodo, Open Science Framework OSF, etc.).

- "As an alternative, clustifyr also supports per-cell annotation, however the runtime is greatly
  increased and the accuracy of the cell type classifications are decreased due to the sparsity of
  scRNA-seq datasets, and requires a consensus aggregation step across multiple cells to obtain
  reliable cell type annotations."

You might consider offering another alternative option. One extreme is to use the cluster averages, while
the other extreme is to use single cells. Perhaps there might be a middle ground where clustifyr could
automatically use k-means or some other algorithm to form clusters within the user-defined clusters. This
would give the user even more flexibility.

After reviewing the code, I can see that there is an "overcluster()" function that seems to do exactly what I
suggested. Please consider describing this in the article and showing an example of how it works. In
retrospect, I can see that the section titled "Subclustering" was supposed to describe this topic — I
misunderstood this section on the first read.

You may want to double-check all of the links in all of your HTML pages. I see three URLs:

- https://github.com/rnabioco/clustifyrdatahub/
- https://github.com/rnabioco/clustifyr
- https://github.com/rnabioco/clustifyrdata

I can see that the "clustifyrdatahub" repo has code for creating ".rda" files from the reference datasets.

I also see similar scripts at https://github.com/rnabioco/clustifyrdata/tree/master/data-raw

Readers might be confused when they see two different repos with similar scripts. You might consider deleting the "clustifyrdatahub" repo if it is not necessary.

I'm happy to see that the data is organized and annotated in the GitHub repo. Specifically, in the GitHub "clustifyrdata" repo, in the "README.md" file, the table shows the name of the reference, the number of cell types, the number of genes, the organism, and a link to the publication. Please consider adding some version of this table to the article, so the reader can understand the scope of this article.

After reviewing the code, I was able to resolve some of my misunderstandings caused by lack of clarity in the terse descriptions in this article. To reduce the chance of misunderstanding by other readers, you might consider clarifying or adding details to the descriptions of functions and results. For example, the article does not mention that GSEA is used to work with gene lists.

References
1. Kiselev VY, Yiu A, Hemberg M: scmap: projection of single-cell RNA-seq data across data sets. Nat Methods. 15 (5): 359-362 PubMed Abstract I Publisher Full Text

Is the rationale for developing the new software tool clearly explained?
Yes

Is the description of the software tool technically sound?
Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?
Partly

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?
Yes

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?
Partly

**Competing Interests:** No competing interests were disclosed.

**Reviewer Expertise:** Bioinformatics, computational biology, immunogenomics, scRNA-seq.
The article introduces a user-friendly and inter-operable R package for cell-type assignment of single-cell RNA-sequencing data. As clearly stated by the authors, the method heavily relies on the (1) results of and (2) any assumptions made by the clustering algorithm applied to the query dataset. The method has potential to be widely useful given its flexibility to take input and give output from many different existing (and future) algorithms. Although the methods proposed are not novel (simple correlation metrics), the software serves to streamline one of the most common procedures in single-cell RNA-sequencing analysis. As detailed below I have some questions regarding the evaluation of the method compared to existing approaches, and a suggestion to more widely distribute the prebuilt references curated as part of the study.

Major comments:
1. The 'unseen population rejection test' is an informative measure. However, it is not clear without going back to the scRNAseq_Benchmark (Abdelaal et al., 2019) how clustifyr's performance compares to other tools. It would be useful to give some quantitative or visualization that conveys this comparison.

2. The approach is aimed at being "normalization-agnostic" as stated in 'Benchmarking' section. However, it's not clear whether this refers to clustifyr in general, or just using the rank correlation setting. If in general, this property should be demonstrated.

3. The benchmarking results provided are very helpful, but it's not clear why only a (differing) subset of the methods was applied to each evaluation (i.e. panels of Figure 4 in particular).

Minor comments:
1. From the description of the method, it seems that if the query dataset is 'over-clustered', meaning a cell-type is incorrectly split into two clusters, clustifyr can return the same cell type assignment for both clusters (provided the correct reference had the highest correlation, and that correlation was above the threshold). Is this correct? If not, please clarify.
2. The prebuilt references in the clustifyrdata github repository has potential utility to researchers who don't already have a reference dataset. It might be a good fit to build these reference datasets as a Bioconductor ExperimentHub package.

References
1. Abdelaal T, Michielsen L, Cats D, Hoogduin D, et al.: A comparison of automatic cell identification methods for single-cell RNA sequencing data. *Genome Biology*. 2019; 20 (1). Publisher Full Text

Is the rationale for developing the new software tool clearly explained?
Yes

Is the description of the software tool technically sound?
Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?
Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?
Partly

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?
Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Statistical genomics, bioinformatics

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
• Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com