Màster Universitari
Anàlisi de Dades Òmiques / Omics Data Analysis
FACULTAT DE CIÈNCIES I TECNOLOGIA
UVIC | UVIC·UCC

Master of Science in Omics Data Analysis

Master Thesis

An evaluation of automated methods for cell type annotation in scRNA-seq data

by

Anna Costa Garrido

Supervisor: Lara Nonell Mazelon, Bioinformatics Unit, VHIO

Academic tutor: M. Luz Calle Rosingana, Biosciences Department,

University of Vic

Biosciences Department

University of Vic – Central University of Catalonia

September 2022

Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: 12 September 2022
Manuscript Category

OXFORD

Gene expression

An evaluation of automated methods for cell type annotation in scRNA-seq data

Anna Costa-Garrido [1,2]*, Lara Nonell [1,2]

[1]Bioinformatics Unit, VHIO, Barcelona, 08035, Spain
[2]Universitat de Vic - Universitat Central de Catalunya (UVic), Vic, 08500, Spain

*To whom correspondence should be addressed.

Abstract

Single-cell RNA sequencing (scRNA-seq) is a powerful new method that makes it possible to study gene expression data at the level of individual cells. Cell type annotation, using a reference sets, is a crucial step in this analysis for obtaining insights into tissue and cell composition. However, there is a need to evaluate and objectively know which are the best annotation tools in the immunology field. In this study, we evaluated the performance of four current automatic cell type annotation methods: Support Vector Machine (SVM), SVMrejection, SingleR and scType using three test sets (MCA, PBMCs and JArribas) and two reference sets (ImmGen and Monaco). Overall, the best-performing method was SingleR based on the percentage of correctly classified cells and the weighted-average F1 score. The results also showed that the classification methods were able to correctly predict most of the cells belonging to a cell type, when there was a good representation of this cell type in the test data. Moreover, SVMrejection not only did not improve the results of SVM but it worsened them. Our findings suggest that SingleR is the best annotation tool, especially when it is fitted for each cell using immune data and the reference set is small or the cell types are imbalanced. As SVMrejection did not perform well, other options must be researched in order to annotate when there are no common cell types between test and reference sets.

1 Introduction

Single-cell RNA sequencing (scRNA-seq) is a powerful tool for characterizing individual cells and producing new insights into tissue composition and dynamic biological processes. It has revealed an unprecedented variety of cell types and subpopulations that were invisible with traditional experimental techniques. Specifically, scRNA-seq can have an important role in understanding immune cell diversity in the tumor microenvironment (TME) by generating high-resolution landscapes of different cancer types. Consequently, this method is able to identify cancer-specific states and composition biases across all major immune cell types that colocalize with cancer cells (Nieto *et al.* (2021)).

Although there has been an emergence of scRNA-seq methods, deconvolutional methods using bulk RNA-seq data are still used to estimate tissue cell proportions. Specifically, there are two types of methodologies; the first one obtains a score describing the enrichment of a cell type in a sample and, the other is a quantitative deconvolution method that estimates the relative fractions of cell types of interest using mostly a linear least square regression. See Cobos *et al.* (2020) for a benchmarking of the different cell type deconvolution pipelines for transcriptomics data.

The advances in understanding tumor composition and its evolution (Kuipers *et al.* (2017)) and also, the scalability of scRNA-seq experiments have rapidly substituted deconvolutional methods that use bulk RNA-seq data (Lafzi *et al.* (2018)). That is, having large numbers of direct single-cell measurements leads to substantially greater resolution of single-cell variation than is possible with deconvolutional methods, even with high-quality bulk data (Lei *et al.* (2022)).

A crucial step in analyzing scRNA-seq is to annotate different cell types and cellular states present in a complex cell mixture based on gene expression profiles. This step is often done through unsupervised clustering of cells based on their transcriptomic profiles, followed by cluster annotation between clusters (Ianevski *et al.* (2022)). This annotation step involves manual inspection of cluster-specific marker genes, which is often a time-consuming, error-prone task that suffers from limited reproducibility across different experiments within and across research groups (Abdelaal *et al.* (2019)). This becomes more pronounced as the number of cells and samples increases, preventing fast reproducible annotation (Ianevski *et al.* (2022)). Consequently, a growing number of classification approaches are being adapted to automatically label cells in scRNA-seq experiments. For instance, SingleR (Aran *et al.* (2019)), the most widespread method in the

bioinformatics community, correlates gene expression profiles of single cells or groups of cells from the test data with given cell types included in the reference data. Machine learning approaches such as Support Vector Machine (SVM, Pedregosa *et al.* (2011)) use a reference data set where labels (cell types) are transferred by supervised classification. Abdelaal *et al.* (2019) showed that SVM had the best overall performance across experiments involving main lineage, deep annotation level, different protocols, and with/without alignment of the datasets. Also, they showed that incorporating a rejection option in SVM (SVMrejection), to account for non represented cell types, led to better performance. Moreover, scType is a promising new marker gene database-based method as it takes into account the specificity of positive and negative marker genes across cell clusters and cells (Ianevski *et al.* (2022)).

These approaches assign each cell in an uncharacterized test dataset based on the most similar reference sample(s) or the selection markers that characterize a cell type. Any published labelled RNA-seq dataset (bulk or single-cell) or marker gene database can be used as a reference, though its reliability depends greatly on the expertise of the original authors who assigned the labels or selected the markers (Amezquita *et al.* (2022)). Another important aspect is that these approaches use different algorithmic strategies that might have an impact on the performance depending on the experiment. For this reason, there is a need to evaluate the performance of these current automatic cell type annotation methods in order to determine the best annotation tools in the immunology field.

To assess which are the best annotation tools, this work presents an evaluation of the performance of SVM, SingleR and, scType using two test sets (MCA and PBMCs) and two reference sets (ImmGen and Monaco) conditioned by species, then an evaluation of the performance of SVM and SVMrejection using both MCA and PBMCs as test and reference sets and different data processing. Finally, the SVM and SingleR annotations are compared using JArribas as the test set and ImmGen as reference.

2 Material and methods

Datasets

A total of three test datasets were used to evaluate all classification methods.

The first test dataset, **PBMCs**, was obtained from human samples and consists of UMI count data from 10x chromium seq technology. It contains peripheral blood mononuclear cells found in one sample, as described in Ding *et al.* (2019).

The second set is a Mouse Cell Atlas, MCA, which consists of UMI count data from Microwell-seq technology found in 6 samples. Han *et al.* (2018) analyzed more than 400,000 single cells covering all of the major mouse organs and constructed a basic scheme for an MCA. However, only the adult peripheral blood samples were included in the evaluation of the annotation methods.

The last one, **JArribas**, is an in house data set, comprising filtered count matrices from 10X Genomics from 3 samples obtained from a mouse model. This dataset was used exclusively to compare annotations obtained from SingleR and SVM.

Two reference sets were used to test SVM and SingleR annotations. **ImmGen** was one of these datasets used to annotate the PBMCs, MCA, and JArribas test sets with the annotation methods. It consists of microarray profiles of pure mouse immune cells from the project of the same name (Heng *et al.* (2008)).

Monaco was the other reference set used to annotate the PBMCs dataset. It consists of bulk RNA-seq samples of sorted immune cell populations from GSE107011, based on humans (Monaco *et al.* (2019)).

All test and reference datasets, except JArribas, had been previously annotated, and it is important to take into account that these annotations were considered as ground truth for the evaluation of the performance of the classification tools. Consequently, we validated that these annotations came from expert knowledge and were scientifically verified. They were not obtained from any computational method using the classification tools to be evaluated.

Table 1 summarizes previously mentioned datasets.

Data processing

A Quality Control (QC) was performed in order to check if our data was correctly distributed. Various metrics were used to assess the distribution of the data. Low values of the number of genes detected in each cell indicated dead/dying or an empty droplet, whereas high values of the total number of molecules detected within a cell were doublets (or multiplets). If these cases were detected, they were consequently removed from the dataset. Figure S1 shows how these metrics were distributed for MCA and PBMCs datasets.

Regarding the test datasets, the Seurat package (version 4.1.1) (Hao *et al.* (2021)) was used to perform the needed preprocessing steps on their counts matrices. These steps were the following:

- **Normalization**: The NormalizeData() function divides counts for each gene by the total counts in the cell and multiplies that value for each gene by the scale.factor (10,000 by default), and then, natural log transforms them.
- **Identification of highly variable features**: this step is performed using the FindVariableFeatures() function with selection.method as vst; First, it fits a line to the relationship of log(variance) and log(mean) using local polynomial regression (loess). Then, it standardizes the feature values using the observed mean and expected variance (given by the fitted line). Feature variance is then calculated on the standardized values after clipping to a maximum and selecting only N top genes (2000 in this case) with the highest variance.
- **Data scaling**: The ScaleData() function shifts the expression of each gene (all genes, not only those with the highest variance), so that the mean expression across cells is 0 and scales the expression of each gene, so that the variance across cells is 1.
- **Linear dimensional reduction**: The RunPCA() function performs a Principal Component Analysis (PCA) on the scaled data with only the previously determined variable features used as input.
- **Cluster the cells**: The FindNeighbors() function uses a KNN graph based on the Euclidean distance in the PCA space, and refines the edge weights between any two cells based on the shared overlap in their local neighborhoods (Jaccard similarity) and the FindClusters() function uses the Louvain algorithm.
- **Non-linear dimensional reduction**: The RunUMAP() function uses the Uniform Manifold Approximation and Projection (UMAP) algorithm for dimension reduction to visualize and explore these datasets, and learns the underlying manifold of the data in order to place similar cells together in low-dimensional space.

Once those preprocessing steps were applied to each test set, the Seurat object obtained from this processing was used in SingleR and scType to perform the cell-type annotation and the scaled data from the same object was obtained for the SVM prediction of cell types.

The reference data (ImmGen and Monaco) were already log normalized and consequently, only the ScaleData() function was applied. The corresponding object was used with SingleR and scType to perform the cell-type annotation and the scaled data from the same object was obtained for the SVM prediction of cell types, the same as the test datasets.

Apart from that, PBMCs and MCA were used both as test and reference data equally while evaluating SVM and SVMrejection. Here, the raw counts and the scaled data from Seurat processing were used. Also, the same experiments were performed trying to correct the batch effects from both datasets using the Matching Mutual Nearest Neighbors (MNN) algorithm, check Haghverdi *et al.* (2018) for more information about this algorithm.

Furthermore, the cell type's correlation matrix was computed for MCA and PBMCs dataset in order to assess the dataset complexity (Figure S2).

Classification methods

This section describes how the classification methods used in this paper work. Table 2 shows a summary of all of them.

SVM

Support Vector Machine (SVM) is a supervised learning algorithm that can be used with classification problems. The algorithm finds the optimal separating hyperplane between classes using nonlinear mapping to a sufficiently high dimension. The hyperplane is defined by the observations that lie within a margin optimized by a cost hyperparameter C that gives a trade-off between getting a large margin and classifying correctly as many examples as possible. These observations are called the support vectors (Hastie *et al.* (2017), Kuhn and Johnson (2016)).

An important part of the SVM is the use of kernels. They are able to enlarge the feature space in a specific way so as to find the optimal separating hyperplane between classes using nonlinear mapping. That is, the use of kernels reduces the amount of computation required for SVM by avoiding the math that transforms the data from low to high dimensions. The linear kernel is the simplest of all the kernels, which is the one implemented in this work. Technically, the data is not projected onto higher dimensions when this kernel is used, so it is just the inner product of observations with the constant term C. The linear kernel is typically used on data sets with large amounts of features as increasing the dimensionality on these data set does not necessarily improve separability (Hsu *et al.* (2003)).

Section B of the Supplementary Information has more details related to SVM and its formulation.

SVMrejection

Some classes present in the test data might not be in the reference. Then, a rejection option must be constructed in order to identify those cases when the prediction step of the classes in the test data is performed, like suggested in Abdelaal *et al.* (2019).

However, the output of SVM is represented by scores and this rejection step cannot be done. **Platt scaling** or **Platt calibration** is a way of transforming the output of a classification model into probability distribution over classes by fitting a logistic regression model to the classifier's score (Pedregosa *et al.* (2011)).

Specifically, it produces the posterior probability $P(y = 1|f)$ by fitting the logistic regression model considering y as arbitrarily labeled +1 and -1 with a binary classification and f as the classifier's score:

$$P(y = 1|f) = \frac{1}{1 + \exp Af + B}$$

The parameters A and B are estimated using a maximum likelihood method that optimizes the training set. A held-out calibration set or cross-validation can be used to avoid overfitting to this set, but Platt additionally suggests transforming the labels y to target probabilities: $t_+ = \frac{N_+ + 1}{N_+ + 2}$ and $t_- = \frac{1}{N_- + 2}$ where N_+ and N_- are the number of positive and negative samples, respectively. This transformation follows by applying Bayes' rule to a model of out-of-sample data that has a uniform prior over the labels (Platt (1999)).

SingleR

The annotation of cell types in SingleR is performed either for each cell independently or for each cluster already found in the *Cluster the cells* step from the *Data processing*. The steps to obtain SingleR annotations are the following:

First, a Spearman correlation coefficient is calculated between the single-cell expression or aggregated profile per clusters in the test data and each sample of the reference data set. The calculation of this correlation only uses the variable marker genes identified by pairwise comparisons between labels in the reference data, so as to improve resolution of separation between labels because these marker genes are those that drive it. Here, it is important to note that the use of Spearman's correlation provides a measure of robustness to batch effects across test and reference datasets (Aran *et al.* (2019), Lun (2022)).

Next, the correlation coefficients for each label of the reference data set are aggregated to provide a single value per cell type and per single cell/cluster. By default, SingleR aggregates these coefficients using the 80th percentile of correlation values as a score for that label and single cell/cluster, to prevent misclassification due to heterogeneity in the reference samples (Aran *et al.* (2019), Lun (2022)).

Although it is optional, a fine-tuning step is implemented in this work where SingleR reruns the correlation analysis, but only for the cell types close to the maximum score, computed from the previous step. Then, scores are recomputed using only marker genes for the subset of labels, focusing on the most relevant features. Finally, the lowest-value cell type is removed (or values more than 0.05 below the top value), and then this step is repeated until only two cell types remain. That is, the label corresponding to the top value score after the last run is assigned to the single-cell (Aran *et al.* (2019), Lun (2022)).

scType

Cell-type annotation in scType is performed using an in-built comprehensive marker database that integrates the information available in the CellMarker database and PanglaoDB.

It also uses a cell-type specificity score (S) that measures how uniquely a particular marker (i) identifies a specific cell-type of the given tissue (t).

This score S is calculated separately for each marker gene M_i within a tissue t as $S_i^t = 1 - \frac{|M_i|_t - min(|M|_t)}{max(|M|_t) - min(|M|_t)}$ where $|M_i|_t$ denotes the number of cell types of tissue t where the ith marker is enlisted and $min(|M|_t)$ and $max(|M|_t)$ are the minimum and maximum number of cell types for which any of the provided genes is enlisted as a marker in the scType database (Ianevski *et al.* (2022)).

Then, in order to assign each cell-type to a cluster (p) given the input scRNA-seq data (X) with m genes and n cells, each gene expression profile is standardized into z-scores across all cells. Considering only positive and negative marker genes corresponding to different cell types of the specified tissues, these markers are extracted from the scType database (Ianevski *et al.* (2022)).

Moreover, each gene expression level is multiplied with its cell type-specificity score (S_i^t): $X' = ((Z(X^T))^T \in M_t)\dot{S}_i^t$, resulting in a transformed expression matrix of n cells and $|M_t|$ genes. Here, M_t represents the vector of marker genes of all cell types within the tissue t and Z denotes the z-score explained previously (Ianevski *et al.* (2022)).

These transformed expression values for each cell-type are summarized into cell type-specific marker-enrichment-score as the normalized sum of all the individual genes supporting a cell-type and such transformation: $x'_c = \frac{\sum_{i=1}^{j} x'_i}{\sqrt{j}} - \frac{\sum_{k=1}^{l} x'_k}{\sqrt{j}}$. Here, c represents a specific cell-type within the tissue, $i, \ldots, j$ are the indices corresponding to cell-type-specific

marker genes, while $k, \ldots, l$ are the indices of negative marker genes that are not expected to be expressed in the cell type. x'_c results in normalized expression matrix of c-by-n dimension, where each row represents a cell type and each column an individual cell (Ianevski *et al.* (2022)).

Finally, the values of each row (cell type) are summed up across the cells corresponding to a specific cluster p, obtaining the cluster summary enrichment-score (called scType score): ScType score$_c$ $= \sum_{z \in p} x_c^z$. A cell type with the highest scType score is assigned to the cluster p (Ianevski *et al.* (2022)).

Evaluation of the methods

The performance of cell type annotation methods was evaluated through the number and global percentage of correctly classified cells and the weighted-average F1 score. For each cell type in the test dataset, the number and percentage of correctly classified cells and F1 scores was reported.

Specifically, the F1 score was computed per class in a One vs All manner as: $F_1(class = a) = 2 \cdot \frac{precision(class=a) \cdot recall(class=a)}{precision(class=a) \cdot recall(class=a)}$ where $precision(class = a) = \frac{TP(class=a)}{TP(class=a)+FP(class=a)}$ and $recall(class = a) = \frac{TP(class=a)}{TP(class=a)+FN(class=a)}$ with a as each cell type (Grandini *et al.* (2020)).

The weighted-average F1 score was calculated by taking the mean of all per-class F1 scores while considering each class's support (that is, the number of actual occurrences of that class in the dataset). The weighted-average F1 score was computed as Weighted-average F1 score $= \frac{\sum_{i=1}^{n} s_i F1_i}{\sum_{i=1}^{n} s_i}$ where s_i is the support of each cell type $i = 1, \ldots, n$ (i.e. the number of cells in each cell type included in the dataset) and $F1_i$ is the per-class F1 score $i = 1, \ldots, n$ (Grandini *et al.* (2020)).

When fitting SVM and SingleR, both test and reference data were needed. However, the correspondence of cell types was not the same (Table 3 shows which cell types were found in each dataset). For this reason, the classification methods were fitted using two approaches: the first one with all cell types of both test and reference data (called the *All cell types* setting) and the other one, with their intersection (called the *Common cell types* setting) in order to see how the performance varies between the two.

When the cell type annotation methods were fitted with the *All cell types* setting, we studied how the cell types not present in the reference set were classified by giving the percentage for each annotated cell type in such cases.

Furthermore, the computation time of all experiments was obtained, showing how this time was distributed when the reference data had a different number of cells and different data processing.

As JArribas didn't have any scientifically verified annotations made by experts, we compared the annotations with those of SingleR and SVM. Using SingleR's annotations as ground truth, we computed the global and cell type-specific misclassification percentage for different settings of the methods: whether they had main or specific cell type labels, feature selection or SingleR fitted for each cell or cluster. We also constructed a *Sankey plot* to visualize and compare both annotations for each setting.

The workflow of all experiments performed in this study is described in Figure S3. The analyses were carried out with the R package version 4.1.2 (R development Core Team, GNU, GPL), Rstudio version 1.4.1106 (R Foundation for Statistical Computing, Vienna, Austria), and Python 3.8.5 (Python Software Foundation, Python Language Reference).

3 Results

This section contains the evaluation of four annotation methods (Table 2) using three test sets and two reference sets. These datasets had different numbers of cells types, cells, genes, and samples. In addition, they were obtained from different protocols and species, as can be seen in Table 1.

3.1 Evaluation of SVM, SingleR and scType using MCA and PBMCs data as test sets

We evaluated the performance of SVM, SingleR, and scType in a species specific environment. Thus, we tested MCA using the ImmGen as reference dataset for mouse and PBMCs with Monaco as reference dataset for human.

Moreover, one important aspect in the annotation of cell types was the correspondence of these cells between test and reference data which sometimes can be discordant. Table 3 shows which cell types are present in each dataset, emphasizing the importance of those that are present in the test but not in the reference set, or otherwise.

As this is a real scenario that can happen during the annotation of cell types, we evaluated each method using two settings: with all cell populations in both sets (*All cell types* setting) and with the same cell population between the test and reference datasets. (*Common cell types* setting) (See *Evaluation of the methods* in Methods Section 2).

MCA - ImmGen

If we focus on the cell type annotation where MCA was considered as test set and ImmGen as a reference set, we evaluated the performance of SVM, SingleR, and scType using the *All cell types* setting and the *Common cell types* setting (Figure 1A,C). We visually compared the cell annotations for all methods and settings through the corresponding UMAP plot (Figure 1B,D).

Table 4 shows how SingleR fitted for each cell independently was the best-performing classifier in both settings. In the *All cell types* setting, this method had 4640 (65.4%) cells correctly classified with a weighted-average F1 score of 0.70. Whereas, in the *Common cell types*) the number and percentage of these cells were 5324 (96.03%) with a weighted-average F1 score of 0.98.

In contrast, the method with the lowest metrics was SVM in both settings. It had 4346 (61.25%) cells correctly classified with a weighted-average F1 score of 0.67 in the *All cell types* setting. Whereas, it had 4801 (86.6%) cells correctly classified with a weighted-average F1 score of 0.91 in the *Common cell types* setting.

Overall, all methods performed similarly in the *All cell types* setting, where the percentage of cells correctly classified and the weighted-average F1 score were around 60% and 0.7, respectively. Moreover, in the same setting, the most represented cells types in the test data (> 1300 cells per cell type) were those that had the highest percentage and F1 score (i.e. T cells, B cells and Neutrophils). The others (< 40 cells per cell type) were not found during the annotation step, except Dendritic cells (DC). It was captured by the SVM and SingleR fitted for each cell,with a percentage of 47.37% and 18.42% and F1 score of 0.17 and 0.20, respectively.

ScType annotation showed similar results; those cell types that were more represented in the test data had the highest percentage and F1 score (i.e. T cells, B cells and Neutrophils). Regarding those cell types that were not seen in the reference data using SingleR and SVM (i.e. B cells (Plasmocytes), Erythroid cells, Lymphocytes and Myeloid cells), scType was only capable of correctly predicting Erythroid cells with 76.72% cells correctly classified. None of the B cells (Plasmocytes), Myeloid cells, or Lymphocytes were correctly captured by the method.

Inspecting Figure S4 shows how SVM, SingleR fitted for each cluster, and SingleR for each cell annotated those cell types that were not seen in the reference data. We can see how all methods agree on annotating Myeloid cells as Monocytes, Lymphocytes as Stem cells and Erythroid cells as Stem cells. However, SVM had a lower percentage in this last case, at around 20%. SVM also annotated Erythroid cells as Endothelial cells with a similar percentage. The unique case where all methods didn't agree was on B cells (Plasmocytes); both configurations of SingleR predicted them

Table 1. Description of the datasets used during this study

	TEST DATA			REFERENCE DATA	
	MCA (scRNAseq)	**PBMCs (scRNAseq)**	**JArribas (scRNAseq)**	**Monaco (bulk RNA seq)**	**ImmGen (microarray)**
Number of cell types	9	8	Unknown	8	19
Number of cells	7095	6003	18620	114	830
Number of genes	34947	33694	34285	46077	22134
Number of samples	6	1	3	17	13
Species	Mouse	Human	Mouse	Human	Mouse
Protocol	Microwell - seq	10x Genomics	10x Genomics	Illumina HiSeq 2000	Affymetrix Mouse Gene 1.0 ST Array
Source	Ding *et al.* (2019) UMI count data	Han *et al.* (2018) UMI count data	-	Heng *et al.* (2008) log-normalized expression values	Monaco *et al.* (2019) processed and normalized using the RMA on probe-level data

Table 2. Automatic methods for cell annotation included in this study

Name	Version	Lenguage	Description	Reference
SVM	0.23.2	Python	SVM with linear kernel using LinearSVC() function	Pedregosa *et al.* (2011)
SVMrejection	0.23.2	Python	SVM with linear kernel and rejection option using LinearSVC() and CalibratedClassifierCV() functions	Pedregosa *et al.* (2011)
SingleR	1.8.1	R	Correlation to training set	Aran *et al.* (2019)
scType	Release version	R	Cell type identification using specific marker combinations	Ianevski *et al.* (2022)

Table 3. Description of which cells types are included in each dataset used during this study. Cells types represented as red squares are not included in the corresponding dataset. Abbreviations: ILC, Innate lymphoid cell; NK, Natural killer; NKT, Natural killer T; Tgd, T gamma delta

	TEST DATA		REFERENCE DATA	
Cell type	**MCA (scRNA-seq)**	**PBMCs (scRNA-seq)**	**Monaco (bulk RNA-seq)**	**ImmGen (microarray)**
B cell; B cell (Plasmocyte); B cell, pro	1395; 31 ; 0	676; 0; 0	20; 0; 0	79; 0; 1
Basophil			4	6
Cytotoxic cell		2127		
Dendritic cell	38	88	8	88
Endothelial cell				20
Eosinophil				4
Epithelial cell				25
Erythroid cell	116			
Fibroblast				21
ILC				23
Lymphocyte	127			
Macrophage	1			79
Mast cell	26			20
Megakaryocytes		48		
Microglia				3
Monocyte; CD14+; CD16+		0; 967; 175	12; 0; 0	33; 0; 0
Myeloid cell	1277			
Neutrophil	2144		4	23
NK cell		429	4	38
NKT cells				22
Progenitor			4	
Plasmacytoid		38		
Stem cell				36
Stromal cells				7
T cell; Cd4+ T cell; Cd8+ T cell	2; 1; 1937	0; 1455; 0	12; 30; 16	231; 0; 0
Tgd				71

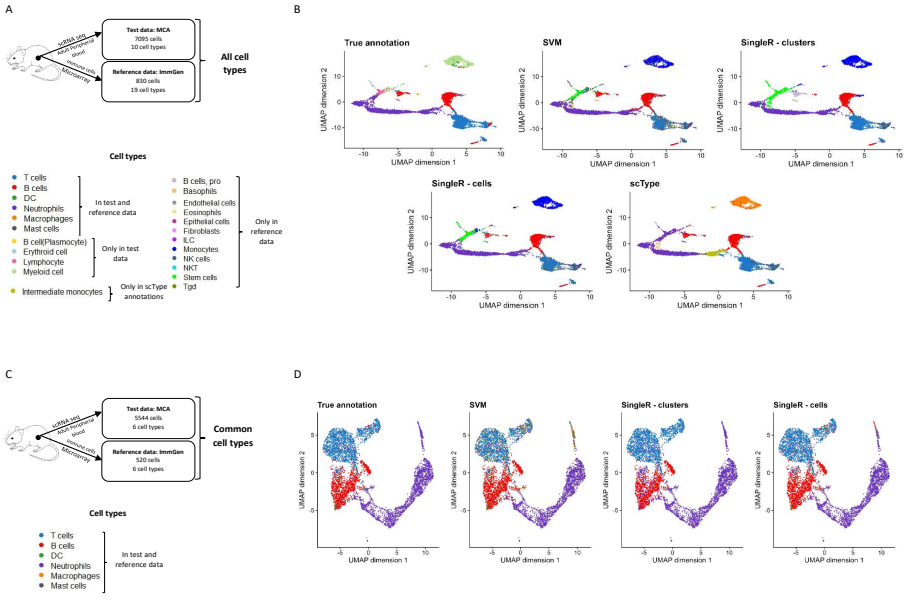


Fig. 1. Workflow and the overall performance of the automatic annotation methods (SVM, SingleR and scType) using MCA and ImmGen as test and reference set, respectively.
A) Scheme representing which test and reference sets are used for the evaluation of these methods using the All cell types setting. Also, it is represented which cell types (with different colors) are present in both sets, only in the test and reference sets and, cell types obtained from the scType predicted annotations as this method uses marker gene database and it does not rely on reference sets. B) UMAP representation for the true annotations of the test set (MCA) and the predicted annotations obtained from the different methods (SVM, SingleR fitted for each cluster and cell and, scType). Each cell type is represented with a different colors and it matches with those specified in A). C) Scheme representing which test and reference sets are used for the evaluation of SVM and SingleR using the Common cell types setting. Also, it is represented which cell types are present in both sets. Here, the scType predicted annotations are not included as this method does not rely on reference sets but in marker gene databases. D) UMAP representation for the true annotations of the test set (MCA) and the predicted annotations obtained from the different methods (SVM and SingleR fitted for each cluster and cell). Each cell type is represented with a different color and it matches with those specified in C).
Abbreviations: DC, Dendritic cell; ILC, Innate lymphoid cell; NK, Natural killer; NKT, Natural killer T; Tgd, T gamma delta.

as B cells but SVM annotates them as Tgd cells, Fibroblasts and B cells at approximately 20% in each case.

Moreover, in the *Common cell types* setting, the performance of the classification methods improved in relation to the *All cell types* setting. The percentage and the weighted-average F1 score were around 90% and 0.9, respectively in all methods, although, SingleR fitted for each cell independently performed best.

Consequently, the percentage and F1 scores for each type also increased. Some of the cells that were not highly represented in the test data and not detected in the *All cell type* setting, were now correctly predicted by some methods with a high percentage. An example of this was Dendritic cells or Mast cells where SVM had 84.21% and 100.00% cells correctly classified, respectively.

PBMCs - Monaco

Using PBMCs as test set and Monaco as reference set, we evaluated the performance of SVM, SingleR, and scType using the *All cell types* setting and the *Common cell types* setting (Figure 2A,C). We visually compared the annotations for all methods and settings through the corresponding UMAP plot (Figure 2B,D).

Table 5 suggests that scType was the best performing classifier in the *All cell types* setting, with 3481 (57.99%) cells correctly classified and a weighted-average F1 score of 0.57. In contrast, SingleR fitted for each cell independently had the highest number and percentage of cells correctly classified (3014 (79.53%)) and weighted-average F1 score (0.86).

In both settings, SVM performed worst of all (with the lowest percentage of cells correctly classified and weighted-average F1 score). We attempted to train this method with the ImmGen reference data to see if the performance improved (Table S1) In the *All cell types* setting, the percentage was reduced from 49.14% to 21.59% and weighted-average F1 score from 0.47 to 0.24 in relation to SVM trained with Monaco. In contrast, in the *Common cell types* setting the percentage improved from 74.88% to 79.31% and the weighted-average F1 score from 0.80 to 0.81.

Overall, all methods performed similarly for the *All cell types* setting, where the percentage and the weighted-average F1 score oscillated between 50 and 60% and between 0.5 and 0.6, respectively (Table 5).

Furthermore, both settings had a good representation of all cell types in the test set (> 80 cells per cell type). These cells types were detected with a percentage of more than 60% for all methods (Table 5).

ScType performed similarly to the other methods, but none of the cell types that were not seen in the reference data during the training of SVM and SingleR (i.e. Cytotoxic, Megakaryocyte and, Plasmacytoid cells) were correctly captured by the method, with 0% of cells correctly classified for each cell type.

Inspecting Figure S5 we can see how all methods agreed on annotating Plasmacytoid cells as Dendritic cells. However, Megakaryocyte cells were classified as Progenitors by SVM and SingleR fitted for each cell, but as NK cells by SingleR for each cluster. Also, both SingleRs shared the fact that Cytotoxic cells were annotated as T cells in around 50 - 75% of the cells. But there were cases where they were annotated as NK cells (around 20%) by SingleR for each cluster and as T cells (around 30%) and NK cells (around 15%) by SingleR for each cell.

Moreover, the *Common cell types* setting showed improved performance in each classification method in relation to the *All cell types*

Table 4. Performance of the different automatic methods (SVM, SingleR fitted for each cluster and cell and, scType) for the cell type annotation used in this study in two settings: All cell types and Common cell types. MCA is used as test and ImmGen as reference set. For each setting, the method's performance is evaluated through the number and global percentage of correctly classified cells and the weighted-average F1 score. For each cell type in the test dataset, it is reported the number and percentage of correctly classified cells and F1 score. The highest metrics are represented with green color, even if it represents the global metric for the annotation method or the cell type included in the test data. B cell (Plasmocyte), Erythroid cell, Lymphocyte and Myeloid cell are cell types only included in the test but not in the reference set. Consequently, methods like SVM and SingleR fitted for each cluster and cell are not able to annotate them in the All cell types and they are represented with a hyphen (-). In the Common cell types setting, these cells are not included when performing the SVM and SingleR fitted for each cluster and cell. They are also represented with a hyphen (-). ScType is able to capture those cell types that are not present in the reference set as it does not rely on this set but in a marker gene database. For this reason, this method is not included in the Common cell types setting.

	MCA - ImmGen											
	All cell types (n = 7095)							**Common cell types (n = 5544)**				
	SVM		**SingleR - clusters**		**SingleR - cells**		**scType**	**SVM**		**SingleR - clusters**		**SingleR - cells**
Num. cells correctly classified (%)	4346 (61.25%)		4592 (64.72%)		**4640 (65.4%)**		4574 (64.47%)	4801 (86.6%)		5187 (93.56%)		**5324 (96.03%)**
Weighted-average F1 score	0.67		**0.70**		**0.70**		0.67	0.91		0.93		**0.98**
Cell types (Num. cells correctly classified (%) | F1 score)												
T cell	1198 (61.75%)	0.76	**1557 (80.26%)**	**0.88**	1389 (71.60%)	0.84	1532 (79.09%)	0.86	1506 (77.63%)	0.86	**1853 (95.52%)**	0.96	1847 (95.21%)	**0.98**
B cell	1309 (93.84%)	0.93	1128 (80.86%)	0.87	1284 (92.04%)	0.93	**1326 (95.05%)**	**0.94**	1325 (94.98%)	0.95	1215 (87.10%)	0.92	**1366 (97.92%)**	**0.98**
Dendritic cell	**18 (47.37%)**	0.17	0 (0.00%)	0.00	7 (18.42%)	**0.20**	0 (0.00%)	0.00	**32 (84.21%)**	0.27	0 (0.00%)	0.00	23 (60.53%)	**0.67**
Neutrophil	1821 (84.93%)	0.91	1907 (88.95%)	0.94	**1960 (91.42%)**	**0.96**	1627 (75.89%)	0.79	1912 (89.18%)	0.94	2119 (98.83%)	0.93	**2068 (96.46%)**	**0.99**
Macrophages	0 (0.00%)	0.00	0 (0.00%)	0.00	0 (0.00%)	0.00	0 (0.00%)	0.00	0 (0.00%)	0.00	0 (0.00%)	0.00	**1 (100.00%)**	**0.04**
Mast cells	0 (0.00%)	0.00	0 (0.00%)	0.00	0 (0.00%)	0.00	0 (0.00%)	0.00	**26 (100.00%)**	0.17	0 (0.00%)	0.00	19 (73.08%)	**0.84**
B cell(Plasmocyte)	-	-	-	-	-	-	0 (0.00%)	0.00	-	-	-	-	-	-
Erythroid cell	-	-	-	-	-	-	**89 (76.72%)**	**0.87**	-	-	-	-	-	-
Lymphocyte	-	-	-	-	-	-	0 (0.00%)	0.00	-	-	-	-	-	-
Myeloid cell	-	-	-	-	-	-	0 (0.00%)	0.00	-	-	-	-	-	-

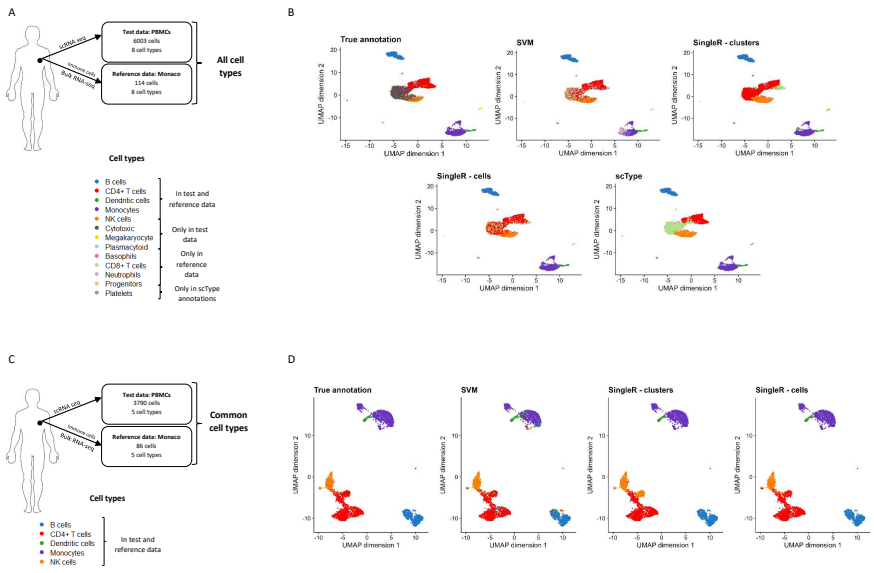


Fig. 2. Workflow and the overall performance of the automatic annotation methods (SVM, SingleR and scType) using PBMCs and ImmGen as test and reference set, respectively. A) Scheme representing which test and reference sets are used for the evaluation of these methods using the All cell types setting. Also, it is represented which cell types (with different colors) are present in both sets, only in the test and reference sets and, cell types obtained from the scType predicted annotations as this method uses marker gene database and it does not rely on reference sets. B) UMAP representation for the true annotations of the test set (PBMCs) and the predicted annotations obtained from the different methods (SVM, SingleR fitted for each cluster and cell and, scType). Each cell type is represented with a different color and it matches with those specified in A). C) Scheme representing which test and reference sets are used for the evaluation of SVM and SingleR using the Common cell types setting. Also, it is represented which cell types are present in both sets. Here, the scType predicted annotations are not included as this method does not rely on reference sets but in marker gene databases. D) UMAP representation for the true annotations of the test set (MCA) and the predicted annotations obtained from the different methods (SVM and SingleR fitted for each cluster and cell). Each cell type is represented with a different color and it matches with those specified in C). Abbreviations: NK, Natural killer.

setting. The percentage and the weighted-average F1 score were between 75% and 80% and between 0.80 and 0.86, respectively. But again, SingleR fitted for each single cell independently had the highest performance.

Computation time

All experiments performed in this Section 3.1 have an associated computation time. Figure 3 shows how this time was distributed over these experiments. In general, SVM and SingleR fitted for each cell had the longest computation time in both settings. Specifically, in the *All cell types* setting, these methods were fitted in 172.68 and 102.10 seconds using the ImmGen reference and, in 4.02 and 12.43 seconds using Monaco. Moreover, in the *Common cell types* setting, these methods were fitted in 8.71 and 27.05 seconds using the ImmGen reference and, in 1.55 and 7.21 seconds using Monaco. Furthermore, having a reference with more cells, made the fitting of the method slower as any method trained with ImmGen took more time to be fitted than one trained with Monaco.

Table 5. Performance of the different automatic methods (SVM, SingleR fitted for each cluster and cell and, scType) for the cell type annotation used in this study in two settings: All cell types and Common cell types. PBMCs is used as test and Monaco as reference set. For each setting, the method's performance is evaluated through the number and global percentage of correctly classified cells and the weighted-average F1 score. For each cell type in the test dataset, it is reported the number and percentage of correctly classified cells and F1 score. The highest metrics are represented with green color, even if it represents the global metric for the annotation method or the cell type included in the test data. Cytotoxic, Megakaryocyte, Plasmacytoid are cell types only included in the test but not in the reference set. Consequently, methods like SVM and SingleR fitted for each cluster and cell are not able to annotate them in the All cell types and they are represented with a hyphen (-). In the Common cell types setting, these cells are not included when performing the SVM and SingleR fitted for each cluster and cell. They are also represented with a hyphen (-). ScType is able to capture those cell types that are not present in the reference set as it does not rely on this set but in a marker gene database. For this reason, this method is not included in the Common cell types setting.

	PBMCs - Monaco											
	All cell types (n = 6003)							**Common cell types (n = 3790)**				
	SVM		**SingleR - clusters**		**SingleR - cells**		**scType**	**SVM**		**SingleR - clusters**		**SingleR - cells**	
Num. cells correctly classified (%)	2950 (49.14%)		3291 (54.82%)		3349 (55.79%)		**3481 (57.99%)**	2838 (74.88%)		2936 (77.47%)		**3014 (79.53%)**	
Weighted-average F1 score	0.47		0.47		0.50		**0.57**	0.80		0.82		**0.86**	
Cell types (Num. cells correctly classified (%) | F1 score)													
B cells	649 (96.01%)	0.90	**675 (99.85%)**	**0.99**	666 (98.52%)	0.99	**675 (99.85%)**	**0.99**	623 (92.16%)	0.87	**674 (99.70%)**	**1.00**	662 (97.93%)	0.99
Dendritic cell	**87 (98.86%)**	0.51	86 (97.73%)	**0.82**	81 (92.05%)	0.66	86 (97.73%)	**0.82**	**86 (97.73%)**	0.56	82 (93.18%)	**0.92**	82 (93.18%)	0.79
Monocytes	782 (68.48%)	0.81	1077 (94.31%)	1.00	1100 (96.32%)	0.98	**1141 (99.91%)**	**1.00**	943 (82.57%)	0.90	**1141 (99.91%)**	**1.00**	1108 (97.02%)	0.98
NK	376 (87.65%)	0.60	**401 (93.47%)**	0.62	387 (90.21%)	0.65	**401 (93.47%)**	**0.64**	411 (95.80%)	0.84	**416 (96.97%)**	0.78	413 (96.27%)	**0.92**
Cd4+ T cells	1056 (72.58%)	0.67	1052 (72.30%)	0.50	1115 (76.63%)	0.60	**1178 (80.96%)**	**0.87**	**775 (53.26%)**	**0.69**	623 (42.82%)	0.60	749 (51.48%)	0.68
Cytotoxic	-	-	-	-	-	-	0 (0.00%)	0	-	-	-	-	-	-
Megakaryocyte	-	-	-	-	-	-	0 (0.00%)	0	-	-	-	-	-	-
Plasmacytoid	-	-	-	-	-	-	0 (0.00%)	0	-	-	-	-	-	-

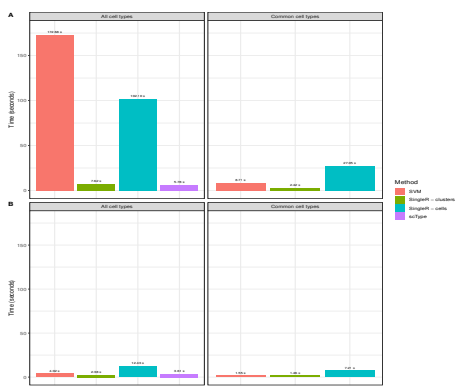


Fig. 3. Computation time of SVM, SingleR for each cell and each cluster and scType using 2 settings: All cells when all cells are used in the test and reference set and Common cell types with only concordant cells between sets are used. A) Barplots with the computation time (in seconds) for each classification method and setting using MCA as test set and ImmGen as reference set (ImmGen has 830 cells with the All cell types setting and 520 with Common cell types setting). B) Barplots with the computation time (in seconds) for each classification method and setting using PBMCs as test set and Monaco as reference set (Monaco has 114 cells with the All cell types setting and 86 with Common cell types setting).

3.2 Evaluation of SVM and SVMrejection

This section shows some insights on the implementation of SVM and SVMrejection.

Before this implementation, SVMrejection, a variation of SVM that incorporates a rejection option to account for non represented cell types, was fitted with MCA and PBMCs as tests and ImmGen and Monaco as reference sets. However, most of the cells were annotated as *T cell* for both cases and a small part of them as *Unknown*.

In an attempt to see if enlarging the size of the datasets or using two scRNA-seq datasets as both test and reference data improves the performance, we compared the annotations of SVM and SVMrejection with MCA as test data and PBMCs as reference data, and the other way around. Also, in an effort to understand the rejection option of SVM, we also included different data preprocessing methods.

Although, in general, SVMrejection had a higher global percentage and weighted-average F1 score because of the cells correctly classified as "Unknown", cell types seen in both sets (i. e. T cells, B cells and Dendritic cells) had a lower percentage and F1 score using SVMrejection than SVM (Table 6 and 7).

Also, it is important to mention that Seurat processing had the highest global percentage and weighted-average F1 score when both MCA and PBMCs were used as test data. It had 1924 (27.12%) of cells correctly classified and a weighted-average F1 score of 0.29 with SVM and, 3021 (42.58%) of cells correctly classified and a weighted-average F1 score of 0.43 with SVMrejection using MCA as test data. Moreover, it had 1931 (32.17%) of cells correctly classified and a weighted-average F1 score of 0.23 with SVM and, 3690 (61.47%) of cells correctly classified a weighted-average F1 score of 0.65 with SVMrejection using PBMCs as test data.

Finally, cell types like B cells (Plasmocytes), Erythroid cells, Lymphocytes, Macrophages, Mast cells, Myeloid cells and Neutrophils were not present in the reference set and not seen during the training of the classification methods using MCA as s test set. Figure S6 shows how SVM and SVMrejection annotated these cell types for each data processing. Figure S7 shows how SVM and SVMrejection annotated cell types not included in the reference set and not seen during the training of these methods (i.e. CD14+, CD16+, Cytotoxics, Megakaryocyte cells, Natural Killer cells and, Plasmacytoid cells) using PBMCs as test.

Computation time

All experiments performed in this Section 3.2 have an associated computation time. Figure 4 shows how this time was distributed over these experiments.

In all of them, SVMrejection took the longest to be fitted. For example, it took 78.83 seconds using raw counts processed with MNN algorithm, whereas SVM took 19.17 seconds.

With MCA as the reference set, Seurat was the data processing with longest computation time (60.25 seconds when fitting with SVM and 191.85 seconds with SVMrejection). Whereas with PBMCs as a reference set, both Seurat processing and raw counts had the longest computation time. In this case, SVM took 36.91 seconds with Seurat processing and 35.63 seconds with raw counts; SVMrejection took 128.05 with Seurat and 114.91 seconds with raw counts.

Table 6. Performance of SVM and SVMrejection using three different data processing: raw counts, processed with Seurat and raw counts processed with Mutual Nearest Neighbor (MNN) method. MCA is used as test set and PBMCs as reference set. For each data processing, the method's performance is evaluated through the number and global percentage of correctly classified cells, the weighted-average F1 score and the number and percentage of cells unlabeled. For each cell type in the test dataset, it is reported the number and percentage of correctly classified cells and F1 score. Apart from that, there are cell types that are included in the test but not in the reference. If this cases are classified as Unknown with SVMrejection, the number, percentage and F1 score of these cases are computed. With SVM, these cases are represented as hyphens (-) as this methods does not contain a rejection option. The highest metrics for each data processing are represented with green color, even if it represents the global metric for the annotation method or the cell type included in the test data.

	MCA: test data, PBMCs: reference data					
	RAW COUNTS (not aligned)		**PROCESSED WITH Seurat**		**PROCESSED WITH MNN (aligned)**	
	SVM	**SVMrejection**	**SVM**	**SVMrejection**	**SVM**	**SVMrejection**
Num. cells correctly classified (%)	1885 (26.57%)	**1955 (27.55%)**	1924 (27.12%)	**3021 (42.58%)**	1325 (18.68%)	**2029 (28.6%)**
Weighted-average F1 score	**0.30**	0.23	0.29	**0.43**	0.23	**0.37**
Num. cells unlabeled (%)	-	5320 (74.98%)	-	4490 (63.28%)	-	2540 (35.8%)
Cell types in test data and reference (Num. cells correctly classified (%) | F1 score)						
T cell	**743 (53.26%) 0.66**	1 (0.07%) 0.00	**1216 (87.17%) 0.86**	147 (10.54%) 0.19	**973 (69.75%) 0.78**	664 (47.60%) 0.63
Dendritic cell	**1111 (57.27%) 0.62**	1 (0.05%) 0.00	**706 (36.39%) 0.44**	213 (10.98%) 0.18	**332 (17.11%) 0.26**	173 (8.92%) 0.16
B cell	**31 (81.58%) 0.08**	1 (2.63%) 0.05	**2 (5.26%) 0.06**	0 (0.00%) 0.00	**20 (52.63%) 0.39**	10 (26.32%) 0.35
Other cell types not seen in reference (n = 3722) as unknown	- -	1952 (52.44%) 0.43	- -	2661 (71.49%) 0.65	- -	1182 (31.76%) 0.38

Table 7. Performance of SVM and SVMrejection using three different data processing: raw counts, processed with Seurat and raw counts processed with Mutual Nearest Neighbor (MNN) method. PBMCs is used as test set and MCA as reference set. For each data processing, the method's performance is evaluated through the number and global percentage of correctly classified cells, the weighted-average F1 score and the number and percentage of cells unlabeled. For each cell type in the test dataset, it is reported the number and percentage of correctly classified cells and F1 score. Apart from that, there are cell types that are included in the test but not in the reference. If this cases are classified as Unknown with SVMrejection, the number, percentage and F1 score of these cases are computed. With SVM, these cases are represented as hyphens (-) as this methods does not contain a rejection option. The highest metrics for each data processing are represented with green color, even if it represents the global metric for the annotation method or the cell type included in the test data.

	PBMCs: test data, MCA: reference data					
	RAW COUNTS (not aligned)		**PROCESSED WITH Seurat**		**PROCESSED WITH MNN (aligned)**	
	SVM	**SVMrejection**	**SVM**	**SVMrejection**	**SVM**	**SVMrejection**
Num. cells correctly classified (%)	2073 (34.53%)	**2665 (44.39%)**	1931 (32.17%)	**3690 (61.47%)**	1884 (31.38%)	**2120 (35.32%)**
Weighted-average F1 score	0.24	**0.43**	0.23	**0.65**	0.21	**0.30**
Num. cells unlabeled (%)	-	1161 (19.34%)	-	3296 (54.91%)	-	706 (11.76%)
Cell types in test data and reference (Num. cells correctly classified (%) | F1 score)						
T cell	**602 (89.05%) 0.88**	443 (65.53%) 0.79	**675 (99.85%) 0.87**	623 (92.16%) 0.96	**395 (58.43%) 0.71**	257 (38.02%) 0.54
Dendritic cell	**1413 (97.11%) 0.52**	1377 (94.64%) 0.53	**1255 (86.25%) 0.54**	665 (45.70%) 0.47	**1454 (99.93%)** 0.50	1452 (99.79%) **0.52**
B cell	**58 (65.91%) 0.72**	3 (3.41%) 0.07	**1 (1.14%) 0.02**	0 (0.00%) 0.00	**35 (39.77%) 0.55**	15 (17.05%) 0.29
Other cell types not seen in reference (n = 3784) as unknown	- -	842 (22.25%) 0.34	- -	2402 (63.48%) 0.68	- -	396 (10.47%) 0.18

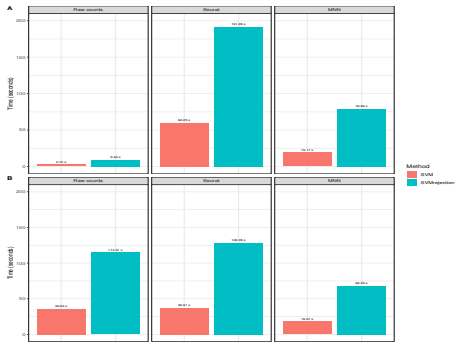


Fig. 4. Computation time of SVM and SVMrejection using 3 different data processing: Raw counts, processing with Seurat and processed with the Mutual Nearest Neighbor (MNN) method. A) Barplots with the computation time (in seconds) for each classification method and data processing using PBMCs as test set and MCA as reference set (taking into account that MCA dataset has 7095 cells). A) Barplots with the computation time (in seconds) for each classification method and data processing) using MCA as test set and PBMCs as reference set (taking into account that PBMCs dataset has 6003 cells).

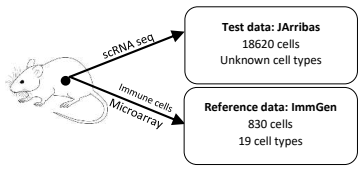
3.3 Comparison of SVM and SingleR annotations using JArribas dataset with ImmGen as reference

This section aims to compare SingleR and SVM annotations on a dataset that does not contain annotations that have been scientifically verified (the JArribas dataset). This comparison is done in a species specific environment. Thus, we tested JArribas using the ImmGen as reference dataset for mouse.

Different settings for both methods were taken into account to produce their annotations: whether they had a general (main) or specific cell type was used, feature selection or not in the test data and the configuration of SingleR: for each cell or for clusters (Figure 5A). For example, for Setting 1a) SVM was trained using specific cell type and with feature selection in the test data, then SingleR was trained with the same characteristics but it was fitted per clusters. In contrast, Setting 1b had the same SVM annotation but SingleR was fitted for cells.

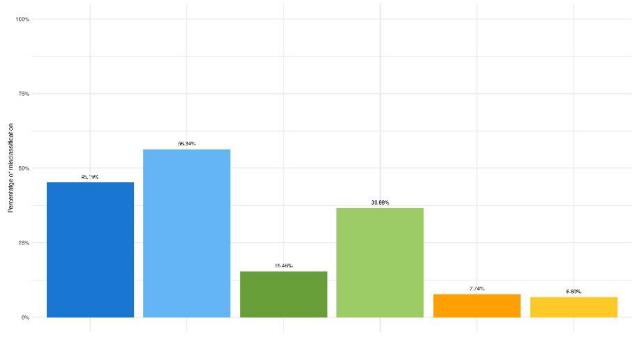
Figure 5B shows the global misclassification percentage for each setting with SingleR annotation as gold standard (true predictions). Settings 3a and 3b had the lowest misclassification percentage (7.74 % and 6.80 %, respectively). The same percentage was computed specifically for each cell type. In each case, the lowest one was obtained using the same settings 3a and 3b (Figure 5C). Figure 5D visually compares the annotations for SingleR and SVM and each setting through the corresponding Sankey plot. Both Figure 5C,D shows how B cells and Dendritic cells were better classified in Setting 2a, 3a and 3b compared to the others, with less than 4% of misclassification in each case. Fibroblasts and Neutrophils were worse classified in Setting 1a and 1b compared to the others, with more than 87% of misclassification in each case. And ILC, Macrophages, NK cells and T cells were best classified in Setting 3b, with less than 28 % of misclassification in each case.

A

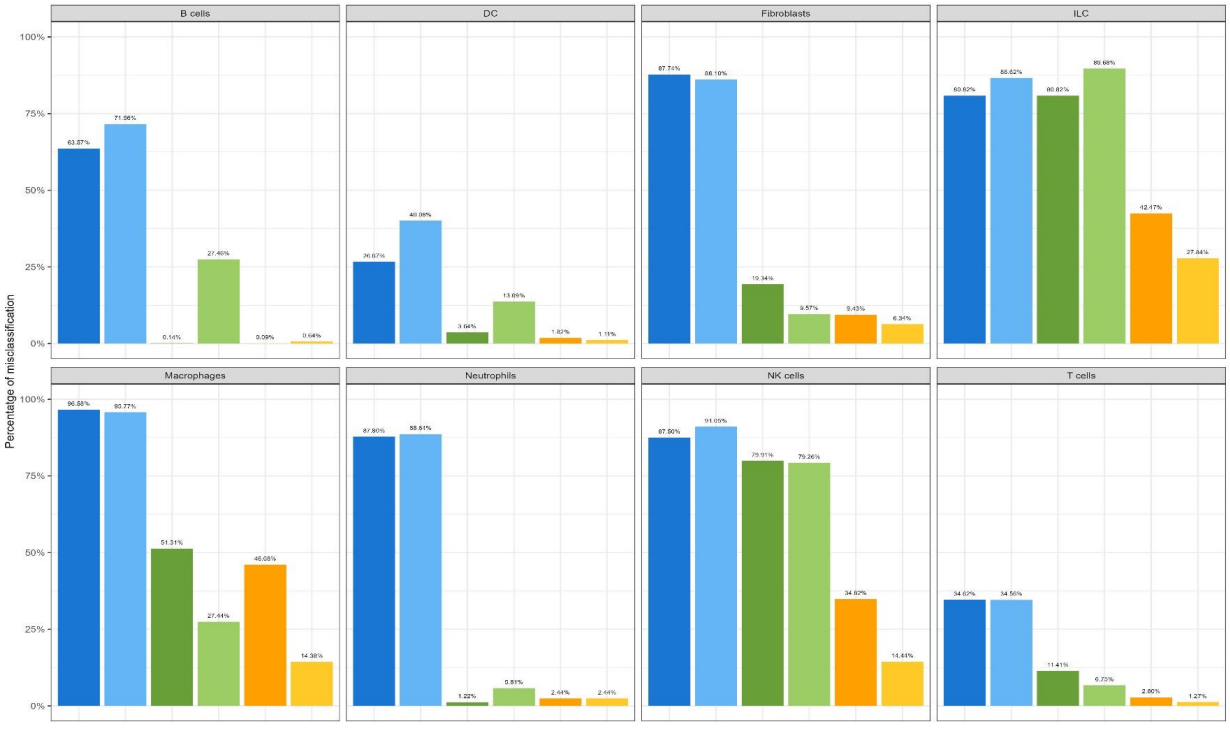


	Cell type label	Feature selection	SingleR configuration
Setting 1a	Specific	Yes	Per clusters
Setting 1b	Specific	Yes	Per cells
Setting 2a	Main	Yes	Per clusters
Setting 2b	Main	Yes	Per cells
Setting 3a	Main	No	Per clusters
Setting 3b	Main	No	Per cells

B



C



D

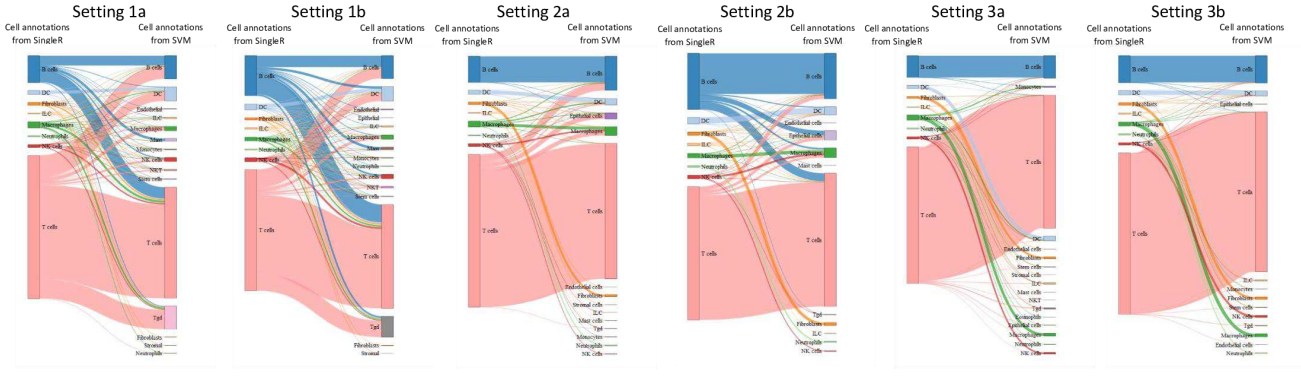


Fig. 5. Comparison of SVM and SingleR annotations using JArribas dataset. A) Scheme representing which test and reference sets are used for the comparison of these methods. Their annotations are obtained through different settings present in the table. Each Setting have an associated color as stated in the table. B) Taking SingleR's annotations as gold standard (true annotations), the barplot shows the global misclassification percentage for each setting according to the color specified in A). C) The barplots represent the misclassification for each cell type for every setting. These cell types are present in the annotation of SingleR for all settings. D) Sankey plot showing the cell annotations from SingleR (in the left) and SVM (in the right) for each setting. It is only selected those cell types are present in the annotation of SingleR for all settings.

4 Discussion

In this study, a comprehensive evaluation was conducted to assess the performance of four automatic annotation methods for scRNA-seq analysis (SVM, SVMrejection, SingleR and ScType). We evaluated these methods on three test datasets (MCA, PBMCs and JArribas) and two representative and detailed reference datasets (ImmGen and Monaco). We systematically assessed the performance through the number and global percentage of correctly classified cells and the weighted-average F1 score.

Regarding to evaluation of SVM, SingleR and scType using MCA and PBMCs data as test sets, the best-performing method was SingleR fitted for each cell. Specifically, this method had the highest number and percentage of cells correctly classified and the best weighted-average F1 score in both MCA and PBMCs test data. The scType method also performed well using PBMCs as test data in the *All cell types* setting. These results are consistent with a previous work suggesting that SingleR performs best when the size of the reference data is small or the cell types are imbalanced (Zhao *et al.* (2020)), as in our reference data. However, the performance of SVM differs from previous reports (Abdelaal *et al.* (2019)) as all analyses all analyses had the lowest percentage of cells correctly classified and weighted-average F1 score.

Moreover, it is important to take into account that the most represented cell types in the test data were those that had the highest percentage of cells correctly classified and F1 score, whereas those that had less representation, were more difficult to annotate. For example, most of these cell types in the *All cells types* setting with MCA as test data had 0% of cells correctly classified. In contrast, PBMCs had a good representation of all cell types (i.e. each cell type has more than 80 cells) and all cell types were detected with a percentage greater than 60% for all the methods.

Although scType performed similarly to SVM and SingleR using both test data, it should have been able to capture those cell types that were not seen in the reference data during the SVM and SingleR training, as it only relies on a marker gene database. However, most of these cases were not correctly captured by the method. Cytotoxic, Megakaryocyte and Plasmacytoid cells types included in the PBMCs as test had 0% of cells correctly classified.

In addition, in the *All cell types* setting, we analyzed how cell types included in the test set, but not in the reference, were annotated. For example, four cell types were present in MCA but not seen in the ImmGen dataset. In this case, all methods agreed on annotating Myeloid cells as Monocytes, which may be due to the fact that Granulocytes and Monocytes are collectively called Myeloid cells and these come from differentiated descendants with common progenitors derived from hematopoietic stem cells in bone marrow (Kawamoto and Minato (2004)). All methods also agreed on classifying Lymphocytes as Stem cells, as Lymphocytes are mature, infection-fighting cells that develop from lymphoblasts, a type of blood stem cell in bone marrow (The American Cancer Society (2022)). Moreover, in most of the cases, the methods classified Erythroid cells as Stem cells although SVM also annotated them as Endothelial cells in 20% of the cases. Erythroid cells and stem cells are related because the first one is differentiated from hematopoietic stem cells (HSCs) and resides within specific niches in adult bone marrow (Fan *et al.* (2015)). However, Erythroid and Endothelial cells do not have a specific relation to each other. Finally, the unique case where all methods did not agree was with B cells (Plasmocytes). Both configurations of SingleR predict them as B cells, given that both of them are related (Allman and Northrup (2010)) but SVM annotates them as Tgd, Fibroblasts, and B cells in the amount of approximately 20% in each case.

A relevant aspect that is worth mentioning is that enlarging the size of the reference data (i.e. the number of cells), makes the fitting of the method slower as any method trained with ImmGen took more time to be fitted than one trained with Monaco (Abdelaal *et al.* (2019)). The SVM and SingleR fitted for for each cell methods had the longest computation times.

Furthermore, the *Common cell types* setting performed better than the *All cell types* setting, but this is not a real scenario and there is a need to include an "Unknown" option when a cell in the test data is not seen during the training of the method (i.e. not in the reference dataset). Including a rejection option in SVM (SVMrejection) could be an possibility. It is created from a calibration step that transforms the score obtained from the model to probabilities, as Abdelaal *et al.* (2019) proposed).

In the evaluation of SVM and SVMrejection, we compared both methods on the MCA - ImmGen and PBMC - Monaco configurations, but unexpectedly the results were not as promising as Abdelaal *et al.* (2019) stated; all cells were classified as T cells or as *Unknown*. Section C of the Supplementary Information has more details related to experiments performed on the MCA dataset. We speculate that implementing a rejection option remains a challenging task as it relies on posterior probabilities to assign labels but ignores the actual similarity between each cell and the assigned population, as Abdelaal *et al.* (2019) had postulated. This option was additionally tried with MCA as test data and PBMCs as reference data, and the other way around, in an attempt to see if enlarging the size of the datasets or using two scRNA-seq data as both test and reference data improves the performance when training SVM. However, adding this rejection option did not improve the performance of SVM, as the three cell types seen in both sets, T cells, B cells and Dendritic cells, were better classified using SVM without the rejection option. Apart from performing badly, SVMrejection takes the longest time to be fitted, especially with Seurat data processing.

According to the comparison of SVM and SingleR annotations using JArribas dataset with ImmGen as reference, we found that SingleR and SVM produced similar annotations when they were fitted using main cell types labels, without feature selection and SingleR was fitted for each cell, because it had the lowest global and specific-cell type misclassification percentage when SingleR annotation was considered as gold standard.

Some limitations of this study are the limited number of datasets and methods used. Analysing more of them could lead to more robust results and reliable conclusions. Other limitations were inherited from the datasets. For example, reference data (ImmGen and Monaco) were small and the cell types were imbalanced, making it difficult for classification methods to do a correct annotation.

Future studies improving SVMrejection could help in those cases where cell types are discordant between test and reference datasets. Research into other methods may be needed. One option could be using correlations between each cell in the test data and cell types present in the reference, and discarding those that have all correlations below a threshold. Also, research into the possibility of building an ensemble voting of different tools could be a solution in order to improve the performance of the SVM(Zhao *et al.* (2020)).

We suggest the use of SingleR since it performs better compared to the other classifiers, especially when it is fitted for each cell and the reference data is small or the cell types are imbalanced. The results also suggest that any classification method is able to correctly predict most of the cells belonging to a cell type when there is a good representation of this cell type in the test data. We can further speculate that the performance of classification methods is dependent on the reference dataset (as Abdelaal *et al.* (2019), Ding *et al.* (2019), Huang *et al.* (2021) suggested). Finally, we see that SingleR and SVM have similar annotations when using main cell types, without feature selection on the test data and SingleR is fitted for each cell.

5 Conclusions

This project presents a comprehensive evaluation of automatic annotation methods for single-cell RNA sequencing data. We recommend the use of SingleR fitted for each cell type as it had performed best overall, especially when using immune data, when the size of the reference data is small, and when the cell types are imbalanced. SVM performed worse in those cases. Finally, incorporating a rejection option is vital when the cell types are discordant between test and reference datasets, but it remains a challenging task, as all analyses performed in this study using SVMrejection did not perform well. Other options must be researched in order to annotate when there are no common cell types between test and reference sets.

Acknowledgements

I would like to thank the supervisor of this project, Lara Nonell, and also to Mercè Alemany and Pau Marc Muñoz for their help during all these months. Also, I want to thank Toni Lozano for all his advices. Finally, I would like to thank my family for all the unconditional support during this year.

Funding

This work has been supported by BBVA Foundation as a part of the Cancer Immunotherapy and Immunology Program (CI1617-1071/FCT2100).

Availability of data and materials

The source code and all datasets (except JArribas) is available in the GitHub repository, at: `https://github.com/annacostagarrido/thesis_tfm`.

Supplementary information

Supplementary information.pdf.

References

Abdelaal, T. *et al.* (2019). A comparison of automatic cell identification methods for single-cell RNA sequencing data. *Genome Biol*, **20**, 194.

Allman, D. and Northrup, D. (2010). 5.02 - b-cell development*. In C. A. McQueen, editor, *Comprehensive Toxicology (Second Edition)*, pages 35–52. Elsevier, Oxford, second edition edition.

Amezquita, R. *et al.* (2022). Basics of Single-Cell Analysis with Bioconductor. `http://bioconductor.org/books/3.13/OSCA.basic/index.html`.

Aran, D. *et al.* (2019). Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nat Immunol*, **20**, 163–172.

Cobos, F. A. *et al.* (2020). Benchmarking of cell type deconvolution pipelines for transcriptomics data. *Nat Commun*, **11**, 5650.

Ding, J. *et al.* (2019). Systematic comparative analysis of single cell rna-sequencing methods. *bioRxiv*.

Fan, A. X. *et al.* (2015). Chapter 11 - regulation of erythroid cell differentiation by transcription factors, chromatin structure alterations, and noncoding rna. In S. Huang, M. D. Litt, and C. A. Blakey, editors, *Epigenetic Gene Expression and Regulation*, pages 237–264. Academic Press, Oxford.

Grandini, M. *et al.* (2020). Metrics for Multi-Class Classification: an Overview. *arXiv*.

Haghverdi, L. *et al.* (2018). Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nat Biotechnol*, **36**.

Han, X. *et al.* (2018). Mapping the mouse cell atlas by microwell-seq. *Cell*, **172**(5), 1091–1107.e17.

Hao, Y. *et al.* (2021). Integrated analysis of multimodal single-cell data. *Cell*.

Hastie, T. *et al.* (2017). *The Elements of Statistical Learning*. Springer, New York, NY.

Heng, T. *et al.* (2008). The Immunological Genome Project: networks of gene expression in immune cells. *Nature Immunology*, **9**, 1091–1094.

Hsu, C. *et al.* (2003). A practical guide to support vector classification. **11**.

Huang, Q. *et al.* (2021). Evaluation of cell type annotation r packages on single-cell rna-seq data. *Genomics, Proteomics and Bioinformatics*, **19**(2), 267–281. Single-cell Omics Analysis.

Ianevski, A. *et al.* (2022). Fully-automated and ultra-fast cell-type identification using specific marker combinations from single-cell transcriptomic data. *Nat Commun*, **13**, 1246.

Kawamoto, H. and Minato, N. (2004). Myeloid cells. *The international journal of biochemistry and cell biology*, **36**, 1374–9.

Kuhn, M. and Johnson, K. (2016). *Applied Predictive Modeling*. Springer, New York, NY.

Kuipers, J. *et al.* (2017). Advances in understanding tumour evolution through single-cell sequencing. *Biochimica et Biophysica Acta (BBA) - Reviews on Cancer*, **1867**, 127–138.

Lafzi, A. *et al.* (2018). Tutorial: guidelines for the experimental design of single-cell RNA sequencing studies. *Nature Protocols*, **13**, 2742–2757.

Lei, H. *et al.* (2022). Semi-deconvolution of bulk and single-cell RNA-seq data with application to metastatic progression in breast cancer. *Bioinformatic*, **38**, 386–394.

Lun, A. (2022). Assigning cell types with SingleR. `http://bioconductor.org/books/release/SingleRBook/`.

Monaco, G. *et al.* (2019). RNA-Seq Signatures Normalized by mRNA Abundance Allow Absolute Deconvolution of Human Immune Cell Types. *Cell reports*, **26**.

Nieto, P. *et al.* (2021). A single-cell tumor immune atlas for precision oncology. *Genome research*, **31**, 1913–1926.

Pedregosa, F. *et al.* (2011). Scikit-learn: Machine Learning in Python. *JMLRl*, **12**, 2825–30.

Platt, J. (1999). Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. *MIT Press*, pages 61–74.

The American Cancer Society (2022). Normal Bone Marrow, Blood, and Lymphoid Tissue.

Zhao, X. *et al.* (2020). Evaluation of single-cell classifiers for single-cell rna sequencing data sets. *Briefings in Bioinformatics*, **21**(5), 1581–1595.