CIA: a Cluster Independent Annotation method to investigate cell identities in scRNA-seq data

Ivan Ferrari[1,2#], Mattia Battistella[1,2#], Francesca Vincenti[1] , Andrea Gobbini[1], Samuele Notarbartolo[1,3], Jole Costanza[1], Stefano Biffo[1,2], Renata Grifantini[1], Sergio Abrignani[1,4] and Eugenia Galeota[1*]

1. Fondazione Istituto Nazionale di Genetica Molecolare "Romeo ed Enrica Invernizzi" (INGM), Milan, Italy.
2. Department of Biosciences, Università degli Studi di Milano, Milan, Italy.
3. Current address: Infectious Diseases Unit, Fondazione IRCCS Ca' Granda Ospedale Maggiore Policlinico, Milan, Italy.
4. Department of Clinical Sciences and Community Health, Università degli Studi di Milano, Milan, Italy.

[#] These authors equally contributed to this work.

* Corresponding author. Email: galeota@ingm.org

Abstract

Single-cell RNA sequencing (scRNA-seq) has revolutionised our ability to explore the transcriptional landscape of complex tissues and uncover novel cell types and biological functions. However, the task of identifying and classifying cells from scRNA-seq datasets remains a major challenge. To address this issue, we developed a new computational tool called CIA (Cluster Independent Annotation) that can accurately identify cell types across different datasets without the need for a training dataset or complex machine learning processes. Based on predefined cell type signatures, CIA provides a highly user-friendly and practical solution to functional annotation of single cells. Our results demonstrate that CIA outperforms other state-of-the-art approaches, while also having significantly lower computational running times. Overall, CIA simplifies the process of obtaining graphical representations of signature enrichment scores and classification results, providing researchers with a powerful tool to explore the complex transcriptional landscape of single cells.

For further details, see tutorials (https://github.com/ingmbioinfo/cia/tree/master/tutorial).

Introduction

The emergence of scRNA-seq technology has opened up unprecedented possibilities for exploring gene expression profiles at the individual cell level [1]. scRNA-seq has quickly become the preferred approach for analysing the cell transcriptional readouts, enabling researchers to identify cell transitions, discover new cell types and investigate cellular heterogeneity at various levels of granularity. Understanding cellular diversity is a significant challenge in many areas of biology and biomedical research, including immunology, developmental and stem cell biology, neurobiology, and cancer research [2,3].

Once the library preparation and sequencing are completed, count matrices representing the starting point of a scRNA-seq analysis are generated by processing and aligning raw data. The typical computational analysis workflow comprises quality control and filtering of cells and genes, normalisation of gene expression counts and the identification of groups of cells with similar transcriptional patterns. Different clustering methods can be used for the latter step, but setting the correct clustering resolution is not a straightforward process and general guidelines may not apply

equally well to all datasets. In this regard, cells and clusters must be appropriately labelled to identify cell populations, their functions and the represented biological state within a given experimental context. Cell labelling relies on prior biological knowledge from domain experts and cannot be easily delegated to computational tools, although efforts have been made to provide annotation guidelines using automated cell labelling tools [4].

These tools assign cluster identities based on known markers or gene expression patterns associated with specific cell types. However, the identification of reliable cell signatures and gene markers is complicated by the high heterogeneity characterising single-cell datasets, making it challenging to label cell types accurately. While manual annotation is still considered the gold standard for cell type identification, it is more time-consuming and labour-intensive than automatic methods, especially for large datasets.

Recently, valuable gene signature collections have been published, covering various organisms, tissues and cell types at different levels of granularity. These collections can be utilised to assign cell types in newly produced single-cell RNA-sequencing datasets. Moreover, consortia are producing large-scale single-cell datasets in the form of atlases to serve as a reliable reference for grasping the underlying differences among cell types, tissues and individuals [5]. The availability of reference atlases, coupled with the ability of machine learning approaches to exploit prior domain knowledge, represents a new way of analysing single-cell datasets.

In this context, we present CIA, a Python package that automatically assigns accurate cell type labels in scRNA-seq datasets using only gene signatures. We also demonstrate how CIA can leverage the knowledge provided by atlases to extract signatures that can be used to classify cells.

Results

CIA is a Python package developed for signature-based automatic annotation of cells in a clustering-free framework. Unlike most existing classifiers that require an annotated scRNA-seq dataset to train a machine learning model, CIA only relies on a set of biologically meaningful gene lists (e.g., signatures obtained experimentally, custom signatures obtained from literature from a specific field of interest, public databases).

CIA can be easily incorporated into classical scRNA-seq analyses since it is fully compatible with the Scanpy package [6]. Given a set of gene signatures in Gene Matrix Transposed (GMT) file format and a gene expression matrix in an AnnData object, CIA builds a score matrix with signature scores for each entry in the gene signature file and every cell in the expression matrix (**Fig 1**). The CIA package consists of four modules: *investigate*, *report*, *utils* and *external*.

The core module *investigate* includes two functions: *signature_score*, which calculates raw or scaled signature scores (detailed in Methods) and *signature_based_classification*, which labels each single cell of the dataset according to the best matching signature.

signature_based_classification can assign the final cell labels with two different modalities: the “fast mode” and the “fold-change (FC)-based mode”. CIA “fast mode” assigns to each cell the label associated with the highest signature score within the signature score matrix; alternatively, the FC-based mode applies a bootstrap approach, returning ratios between the calculated signature scores and median signature scores of random signatures. With the FC-based, three optional parameters

aimed at obtaining a more reliable classification can be set to apply filters on fold-change score values, fold-change score distributions and p-values.

Additionally, users can also provide signatures of negative markers (genes known to be downregulated or unexpressed for the cell types under investigation) in order to ameliorate the classification. Cell assignments that do not pass the applied filters are reclassified as "Unassigned" (see Methods).

The *report* module assists users in evaluating both the goodness of signatures and the classification performance. This module has four functions: *grouped_distributions*, *group_composition*, *classification_metrics* and *grouped_classification_metrics*. In brief, *grouped_distributions* evaluates whether the distribution of a given signature score within a group of cells (e.g. clusters) is significantly different from its distribution in the other cell groups. *group_composition* returns a contingency table of the classification. *classification_metrics* and *grouped_classification_metrics* compute sensitivity, specificity, precision, accuracy and F1-score. The first function returns the metrics per cell group, providing information about the quality of a single classification for each cell population, while the second function provides the same information for the whole dataset and can be used to rapidly compare different cell annotations at a time.

The *utils* module contains four ancillary functions: *signatures_similarity*, *filter_degs*, *read_gmt* and *save_gmt*. *signatures_similarity* returns the similarity between two gene signatures using either the Jaccard index or computing the percentage of overlap. *filter_degs* subsets differentially expressed genes (DEGs) stored in the uns layer of the AnnData object based on their expression and their significance. *read_gmt* and *save_gmt* can convert signature files in the GMT format to Python dictionaries and vice versa.

In the *external* module we included the *celltypist_majority_vote* function [7], which extends the more represented cell type label to each cell within a cluster/cell group. If reference clusters are not provided by users, the function creates groups by clustering cells at high resolution.

CIA correctly classifies PBMC3k

CIA performances were tested on a human PBMC3k single cell dataset, available on 10X Genomics website (https://support.10xgenomics.com/single-cell-gene-expression/datasets/1.1.0/pbmc3k), comprising 2700 cells from a healthy donor. To automatically annotate cells, we utilised populations' differentially expressed genes from a publicly available reference atlas of human peripheral mononuclear blood cells (PBMC atlas) [8], which was annotated through both transcript and protein levels inspections. The differential expression analysis (see Material and methods) yielded seven gene signatures that represent immune subpopulations: B lymphocytes, CD4+ and CD8+ T lymphocytes, dendritic cells (DC), monocytes (Mono), natural killer cells (NK) and platelets (**Fig 2A**).

We benchmarked our results against the manual classification of the PBMC3k dataset (**Fig 2B**). First, we applied the *signature_score* function to calculate scaled scores associated with each cell of the PBMC atlas for each of the seven subpopulation signatures. By utilising the *grouped_distributions* function we obtained median signature score values within the manually annotated clusters of cells (**Fig S1A**) and we assessed the statistical significance of each signature in representing the corresponding population. The best scores for each signature resulted consistent with the correspondent manually annotated subpopulation.

We then used the signatures from the atlas to classify the PBMC3k dataset with the "fast mode" option (**Fig 2D**). The heatmaps with median signature scores clearly showed that higher median signature scores corresponded to the correct subpopulation (**Fig 2C**). The *grouped_distributions*

function returned a warning indicating that the signature score distribution of DC and Mono were not significantly different within the manually annotated dendritic cells cluster (Wilcoxon test). This reflected both the lack of specific identity markers in the Mono signature, resulting in lower median scores compared to the other signatures, and the high variability of DC signature scores within the DC cluster. However, since for each distribution, there is only a cell group in which values are significantly higher than all the other groups (Mann-Whitney U test), the simple inspection of PBMC3k UMAP showing the different signature scores is sufficient to highlight the expected cell populations (**Fig S1B**).

The "fast mode" classification inferred cell identities with high accuracy, precision, and F1 values (**Table 1**) in only 0.28 seconds. Subsequently, we classified cells with the FC-based mode, setting the number of cycles to 500 and using three different filtering strategies named "standard mode", "q" and "p-val" (**see Methods**).

The different filtering strategies always resulted in highly accurate classifications (F1 score > 0.87) obtained in less than 20 seconds (**Table 1, Figure S2A-B-C**).

Comparison of CIA with state-of-art classifiers

We compared CIA performances with that of two: Celltypist [7] and Besca [9]. Both tools require a labelled reference dataset to train their classification models.
We used the PBMC atlas as a training dataset for both tools **(Fig 2A)**. By applying a combined approach based on logistic regression and stochastic gradient descent, Celltypist can classify cells at both the single-cell and cluster resolution. We trained Celltypist model by running *celltypist.train* function and enabling feature selection. We fitted the model on the PBMC3k cells using *celltypist.annotate* function (**Fig 2E**).

Besca can be trained using one or more datasets of similar tissues and conditions and different models can be fitted to predict cell types in the test dataset. In the case of logistic regression, a threshold can be applied to specify the minimum probability to assign a cell label or not ("unknown"). The tool produces a report with metrics, confusion matrices, and comparative UMAP plots. We performed feature selection using *besca.tl.auto_annot.merge_data* function and we trained the models using *besca.tl.auto_annot.fit* function with either support vector machines (SVM) and logistic regression (LR). Finally, we obtained cell label predictions by running *besca.tl.auto_annot.adata_predict* function (**Fig 2F**).

Overall, all the tested classifiers were able to correctly predict the single cells' identity with good performances (F1 scores > 0.78) (**Table 1**). CIA achieved the best classification performance with the "fast mode" option (F1 score = 0.92), followed by the "p-value" based mode (F1 score = 0.88). Celltypist and BESCA also achieved comparable results (**Fig 2D-E-F**). Notably, CIA performance in terms of running times clearly outperformed those of the other tools.

CIA FC-based mode efficiently highlights unknown cell types

To further test the ability of our classifier to detect the presence of unanticipated cell types, we performed a "standard mode" classification on the Seurat human cord blood mononuclear cells (CBMC) from the "Using Seurat with Multimodal data" vignette (https://satijalab.org/seurat/articles/multimodal_vignette) (**Fig 3A-B**). The CBMC dataset contains human blood cell types and additional mouse cells (~5%) used as negative controls. Signatures previously extracted from the PBMC atlas were used for the classification. Although the PBMC atlas

signatures represent cell populations at lower granularity compared to the annotations provided with the CBMC dataset, CIA was able to correctly distinguish such populations. Strikingly, we correctly labelled as “Unassigned” those cells whose transcriptional profiles were not represented by the provided signatures (78.45% of the mouse cells and 90.38% of human erythrocytes).

CIA enables the fast investigation of functional signatures

In addition to cell type recognition across datasets, we also explored the ability of CIA to identify cells that exhibit a specific biological function. To this end, we calculated the scaled signature score of the "GOBP_PHAGOCYTOSIS" signature from the Molecular Signature Database (**Fig 3C**). By simply providing the URL of the gmt file to the *signature_score* function, we could chart the spatial location of cells exerting phagocytic activity within our UMAP in less than a second.

The highest score values were obtained in the clusters of Mono and DC, which are known to exhibit phagocytic activity (Mann-Whitney U test $p<0.01$) while all the other cell groups had low scores (**Fig 3D**). These results demonstrate how CIA can be leveraged for the functional characterization at single cell resolution.

Methods

Signature score

The *signature_score* function calculates signature scores as previously described in Della Chiara et al. 2021 [10]. It takes as input a dictionary of gene signatures (or a gmt file), an AnnData object (with gene expression matrices) and an option to set the score modality. The function computes a score for each signature and every cell of the dataset; scores depend on the number of signature expressed genes and their relative abundance in the cell.

The score mode input can be used to specify the type of scores to be computed, with options such as "raw", "scaled",“log", “log2” or “log10”.

The “raw” gene signature score (GSS) for cell C is defined as:

$$\mathbf{GSS(C)} = \frac{n}{L} \times \frac{\sum_{i=1}^{n} Xi\ for\ Gi\ in\ S}{\sum_{j=1}^{m} Xj\ for\ Gj\ in\ G}$$

Where:*n* the number of genes in S that are expressed in **C**;

- G={G_1, G_2, …, G_m} is the list of length m that contains all the gene symbols of the cell **C**;
- S={G_1, G_2, …, G_L} is the list of gene symbols of length L that compose the signature;
- X={X_1, X_2, …, X_m} a vector of gene expression values for genes in G in cell **C**.

When the option score_mode is set to “scaled”, raw scores of each cell are divided by the maximum raw score in the dataset, in order to obtain values in [0-1].

Classification

The *signature_based_classification* function uses *signature_score* to compute the scores for the input gene signatures and classifies each cell accordingly. Two main classification options can be set using the parameter fast_mode. If this is set to TRUE, the function labels each cell simply selecting the signature with maximum scaled score value.

Alternatively, with the FC score-based mode, the function generates random signatures and calculates fold-changes of the provided signatures of interest against random ones. The raw signature score is divided by median raw scores of random gene signatures of the same length. In this case the function returns the label of the signature with higher fold-change. The number of random signatures, corresponding to the number of iterations, can be set by the user (n_iter). This function also allows the use of multiple processors (n_proc).

While using FC based mode, three optional filters can be applied to the score in order to keep only reliable cell labels: 1) a different threshold for fold-changes (1 by default), 2) a quantile based filter using the distribution of fold-changes to retain only the highest values and 3) a p-value (probability of being higher than random signature scores).

In our workflow we performed a FC based classification with three different strategies:

- "standard": FC higher than 1 (FC_threshold = 1), meaning where the signature score is higher than the median of random scores.
- "q": FC > 1 and higher than the median of signature score distribution (q = 0.5).
- "p-val": FC > 1 and higher than 95th percentile of random score distribution (p = 0.05).

Cells for which all the provided signatures don't pass the filters are labelled as "Unassigned".

Random signature generation

Random gene signatures are obtained by randomly sampling the genes of the dataset, with replacement. In detail, all the genes of the dataset are ranked by their mean expression and divided into n_bins (default n_bins=25). For each gene of a signature of interest, the function randomly picks a gene, preferentially from highly variable genes, within the same expression bin, generating a random signature of equal length. If there are not highly variable genes in a bin, or if they have not been defined, generic same-bin random genes are used instead. Since this step is iterated several times, the whole computation may become longer, thus the possibility to parallelize the processes has been implemented (n_proc).

Score distribution analysis

The function *grouped_distributions* can be used to determine if the distributions of scores associated with a particular signature are significantly different from others. For each annotated group of cells (e.g. clusters), considered as ground truth annotations, this function applies a Wilcoxon test to verify if the distribution of the scores associated to the signature with the highest median is significantly different from the distributions of scores associated with other signatures. The function then tests, by Mann-Whitney U test, if the distribution of scores of the best signature for each cluster is significantly different from the distribution of scores for the same signature in other clusters. The function returns a heatmap plot of the median scores for each signature in each group, and a warning message if one of the tests has a p-value < 0.01.

Classification performances

To evaluate classification performances, the *classification_metrics* function of the *report* module calculates Sensitivity, Specificity, Precision, Accuracy and F1-scores by matching CIA classification with that of the dataset provided as ground truth.

	PRECISION	ACCURACY	F1-SCORE	RUNNING TIME
CIA FAST MODE	91.58	97.59	91.58	0.23 s*
CIA FC-BASED (FC>1)	88.62	96.75	88.62	0.29 min*
CIA FC-BASED (q>0.5)	88.57	96.71	88.50	0.3 min*
CIA FC-BASED (p<0.05)	89.56	96.51	87.51	0.3 min*
CELLTYPIST	77.55	93.58	77.55	23 min
BESCA LR	84.49	95.57	84.49	4 min 38 s
BESCA SVM	86.88	96.25	86.88	41 min

Table 1 Table reporting cell-level classification performances.
* Running time of differential expression analysis (23.3s) and DEGs filtering (20.3s) are not included since signature refinement is not the intent of this package and it should be done before CIA classification.

Signature extraction from reference datasets

Cell population signatures for the classification of PBMCs were extracted from the Hao et al. 2021 [8] single-cell atlas (161764 cells). Based on the recent weighted nearest neighbour framework [8] this atlas integrates gene expression measurements with those of cell surface proteins quantified by simultaneous scRNA-seq and CITE-seq, thus providing a very specific and reliable annotation. Cell populations of the atlas include three different levels of granularity. For our analyses, the coarser level was chosen. To extract signatures from the atlas we performed a classical differential expression analysis (DEA) exploiting the *scanpy.tl.rank.genes_groups* (with default parameters) of the remaining annotated populations (B lymphocytes, CD4+ and CD8+ T lymphocytes, dendritic cells, monocytes, natural killer cells and “other” cells, mainly composed by platelets). DEG lists were subsequently filtered through the *filter_degs* to set thresholds on the fold-change (1.5), expression mean (0.25), z-score (5) and percentage of cells within each cluster which express the gene (40%).

Discussion

In this study we developed a new computational tool called CIA, that is able to accurately identify cell types across different datasets without the need for a training dataset or complex machine learning processes. Our approach is based on the idea that each cell type has a unique transcriptional signature that can be used to identify it. Inspecting the expression of a query cell with a set of pre-computed cell type signatures, CIA is able to assign a cell type label with high accuracy.

CIA is a highly user–friendly, open source and accessible package that has been designed to be easily integrated into the daily single-cell analysis workflows. Its open-source code is freely available on GitHub (https://github.com/ingmbioinfo/cia), accompanied by detailed documentation and tutorials, making it widely accessible to researchers with basic Python programming skills. Unlike most of the tools, CIA operates independently from the clustering step, allowing users to explore biologically meaningful cell groups before defining the clustering resolution. The “fast mode” option is particularly useful for this purpose, as it can classify large datasets within seconds, adding a layer of prior knowledge that clustering algorithms typically ignore and allowing confident refinement of cluster boundaries. Afterward, the “FC based mode” can be applied to obtain statistically reliable cell labelling.

CIA is based on signatures, which means it doesn’t require a training dataset of similar conditions. Using a training dataset could introduce bias towards the characteristics of that particular dataset. By contrast, with CIA, these issues are mitigated and cell labelling can be achieved accurately and efficiently. An additional key characteristic of CIA is that when the FC based mode is applied, only reliable cells are annotated. This feature is particularly useful because it highlights the cells for which signatures are not suitable or those with ambiguous transcriptional profiles. Therefore, it can prevent the misclassification of cells and reveal potential new cell types or biological conditions that might otherwise be overlooked. As a proof of concept we showed, with the CBMC dataset annotation, that our classifier is capable of accurately detecting cell populations whose transcriptional profiles were represented by the input signatures while highlighting as “Unassigned” negative controls.

Our research team has successfully implemented the use of CIA to accurately classify immune cell populations. This method outperformed other state-of-the-art approaches, while also having significantly lower computational running times. It's worth noting that when we applied the FC-based mode of CIA, we observed lower F1-scores in comparison to the “fast mode”. This was because the "Unassigned" cells were classified as false positives. Although this may seem like a decrease in performance, it actually represents an opportunity to generate new hypotheses regarding the nature of cells included in single-cell datasets.

In addition to the primary modules, CIA also offers several useful utilities that enable users to visualise signature enrichment scores on UMAPs, generate contingency matrices, compare classifications against reference datasets, evaluate classification performance and plot the classification results on UMAPs. By incorporating this information into the existing AnnData object, CIA simplifies the process of obtaining graphical representations of signature enrichment scores and classification results. In addition, the CIA algorithm has been equipped with an annotation method that can harmonise labels within a cluster, facilitating the cluster-level annotation process (see “Over-clustering and majority voting improve cell-level annotation**”,** Supplementary Materials). This feature ensures that the labelling within each cluster is consistent.

Finally, we also showed that CIA can be used to identify cells that express a specific biological function by calculating the signature score of a gene set of interest. Indeed, the cells that are enriched for the GOBP_PHAGOCYTOSIS, are among the Mono and DC clusters, all associated with phagocytic activity.

Overall, our results suggest that CIA is a useful tool for scRNA-seq analysis that can help researchers to identify cell types and biological functions in their datasets without the need for extensive training or optimization, in a relatively low amount of time.

References

[1] Chen G., Ning B., Shi T. (2019). Single-Cell RNA-Seq Technologies and Related Computational Data Analysis. Frontiers in Genetics. DOI:10.3389/fgene.2019.00317

[2] Nguyen Q. H., Pervolarakis N., Nee K., Kessenbrock K. (2018). Experimental Considerations for Single-Cell RNA Sequencing Approaches. Frontiers in Cell and Developmental Biology. DOI:10.3389/fcell.2018.00108

[3] Wagner A., Regev A., Yosef N. (2016). Revealing the vectors of cellular identity with single-cell genomics. Nature Biotechnology. DOI: 10.1038/nbt.3711.

[4] Clarke Z. A., Andrews T. S., Atif J., Pouyabahar D., Innes B. T., MacParland S. A., Bader G. D. (2021). Tutorial: guidelines for annotating single-cell transcriptomic maps using automated and manual methods. Nature Protocols. DOI:10.1038/s41596-021-00534-0.

[5] Lotfollahi M., Naghipourfar M., Luecken M.D., Khajavi M., Büttner M., Wagenstetter M., Avsec Ž., Gayoso A., Yosef N., Interlandi M., Rybakov S., Misharin A. V., Theis F. J. (2022). Mapping single-cell data to reference atlases by transfer learning. Nature Biotechnology. DOI:10.1038/s41587-021-01001-7.

[6] Wolf F., Angerer P., Theis, F. (2018). SCANPY: large-scale single-cell gene expression data analysis. Genome Biology. DOI:10.1186/s13059-017-1382-0

[7] Domínguez C. C., Xu C., Jarvis L.B., Rainbow D. B., Wells S. B., Gomes T.0, Howlett S. K., Suchanek O., Polanski K., King H. W., Mamanova L., Huang N., Szabo P. A., Richardson L., Bolt L., Fasouli E.S., Mahbubani K.T., Prete M., Tuck L., Richoz N., Tuong Z.K., Campos L., Mousa H.S., Needham E.J., Pritchard S., Li T., Elmentaite R., Park J., Rahmani E., Chen D., Menon D. K., Bayraktar O. A., James L. K., Meyer K. B., Yosef N., Clatworthy M. R., Sims P. A., Farber D. L., Saeb-Parsy K., Jones J. L., Teichmann S .A. (2022). Cross-tissue immune cell analysis reveals tissue-specific features in humans. Science. DOI:10.1126/science.abl5197.

[8] Hao Y., Hao S., Andersen-Nissen E., Mauck W. M. 3rd, Zheng S., Butler A., Lee M. J., Wilk A. J., Darby C., Zager M., Hoffman P., Stoeckius M., Papalexi E., Mimitou E. P., Jain J., Srivastava A., Stuart T., Fleming L. M., Yeung B., Rogers A. J., McElrath J. M., Blish C. A., Gottardo R., Smibert P., Satija R. (2021). Integrated analysis of multimodal single-cell data. Cell. DOI:10.1016/j.cell.2021.04.048.

[9] Mädler S. C., Julien-Laferriere A., Wyss L., Phan M., Sonrel A., Kang A. S. W., Ulrich E., Schmucki R., Zhang J. D., Ebeling M., Badi L., Kam-Thong T., Schwalie P. C., Hatje K. (2021). Besca, a single-cell transcriptomics analysis toolkit to accelerate translational research. NAR Genomics and Bioinformatics. DOI:10.1093/nargab/lqab102

[10] Della Chiara G., Gervasoni F., Fakiola M., Godano C., D'Oria C., Azzolin L., Bonnal R. J. P., Moreni G., Drufuca L., Rossetti G., Ranzani V., Bason R., De Simone M., Panariello F., Ferrari I., Fabbris T., Zanconato F., Forcato M,. Romano O., Caroli J., Gruarin P., Sarnicola M. L., Cordenonsi M., Bardelli A., Zucchini N., Ceretti A. P., Mariani N. M., Cassingena A., Sartore-Bianchi A., Testa G., Gianotti L., Opocher E., Pisati F., Tripodo C., Macino G., Siena S., Bicciato S., Piccolo S., Pagani M. (2021). Epigenomic landscape of human colorectal cancer unveils an aberrant core of pan-cancer enhancers orchestrated by YAP/TAZ. Nature Communications. DOI:10.1038/s41467-021-22544-y.

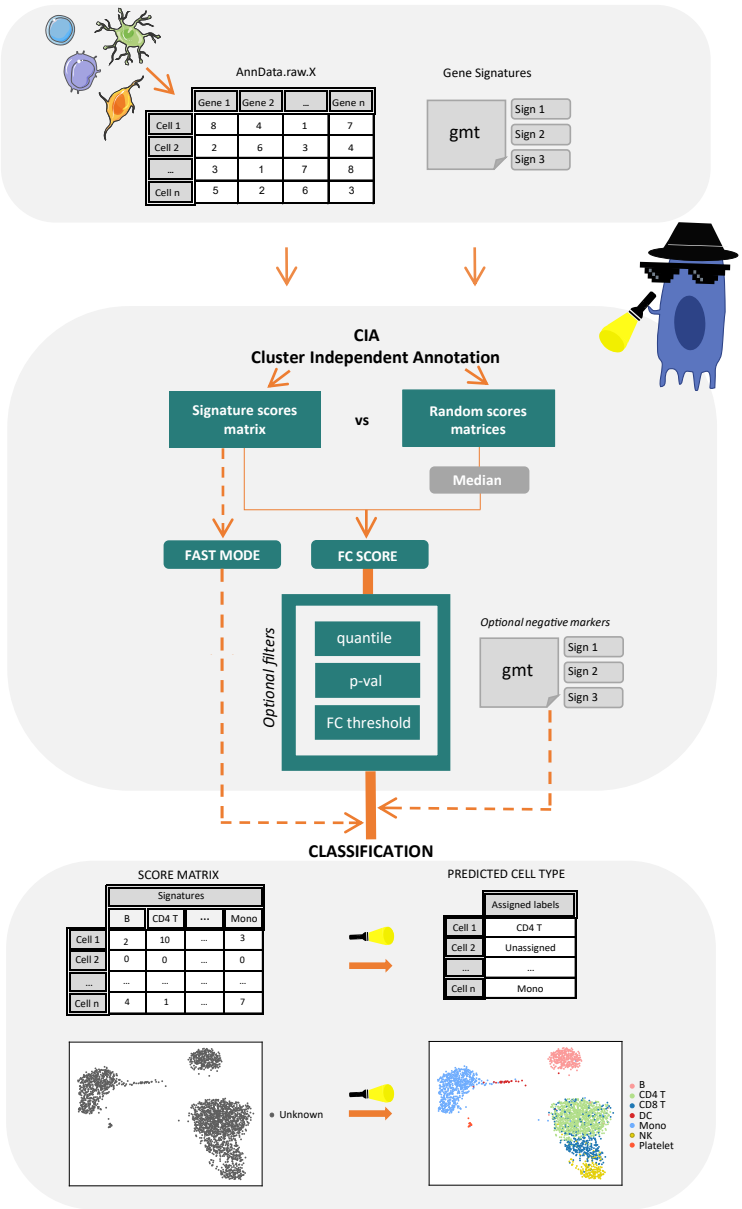

Fig1 Schematic representation of CIA workflow. CIA required inputs are a list of named gene signatures and an Anndata object containing the raw gene expression matrix. The signature_score function calculates for each cell and for each signature a gene signature score. The signature_based_classification function can be called in two different ways. The “fast mode" simply scales raw scores from the signature score matrix in the interval [0, 1]. Alternatively with the FC-based mode, CIA applies a bootstrap approach to obtain gene signature scores of random gene lists. Subsequently, a fold change is obtained by dividing the gene signature scores by the median of the random signature scores. An optional filtering step can be applied based on the distribution of FC scores, p-value or a user-defined threshold. Scores can be further refined providing a list of negative markers. CIA finally assigns cell labels based on the top scored signature name.

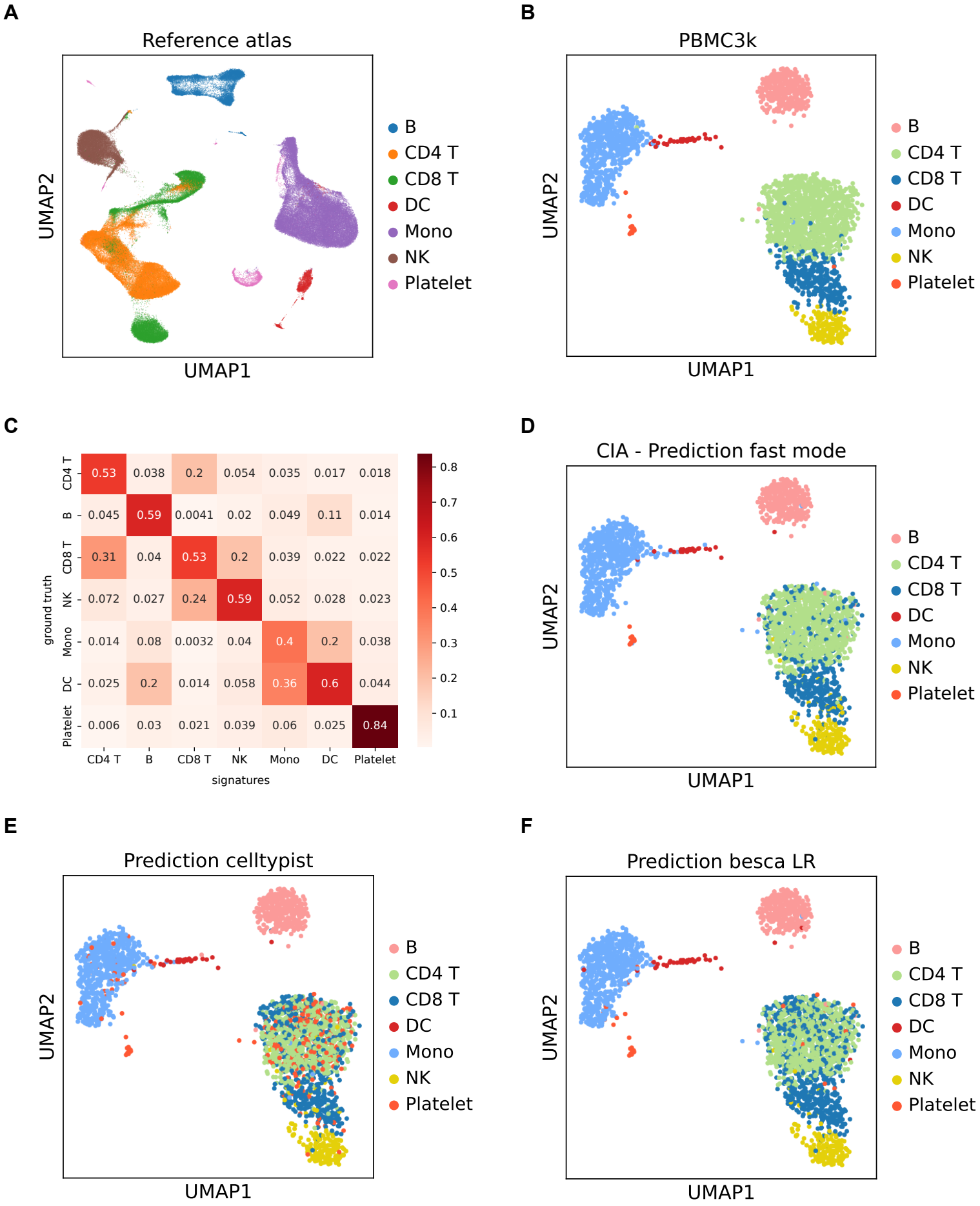

Fig2 A) UMAP plot showing the PBMC atlas dataset with the coarser level of annotation. B) UMAP plot showing the PBMC3k dataset with ground truth annotation. C) Heatmap showing the median scaled score of each signature (x axis) in each PBMC3k cluster (y axis). D) UMAP plot of PBMC3k dataset showing cell labels predicted by CIA “fast mode”. E) UMAP plot showing the classifications of PBMC3k performed by Celltypist. F) UMAP plot showing the classifications of PBMC3k performed by Besca with linear regression.

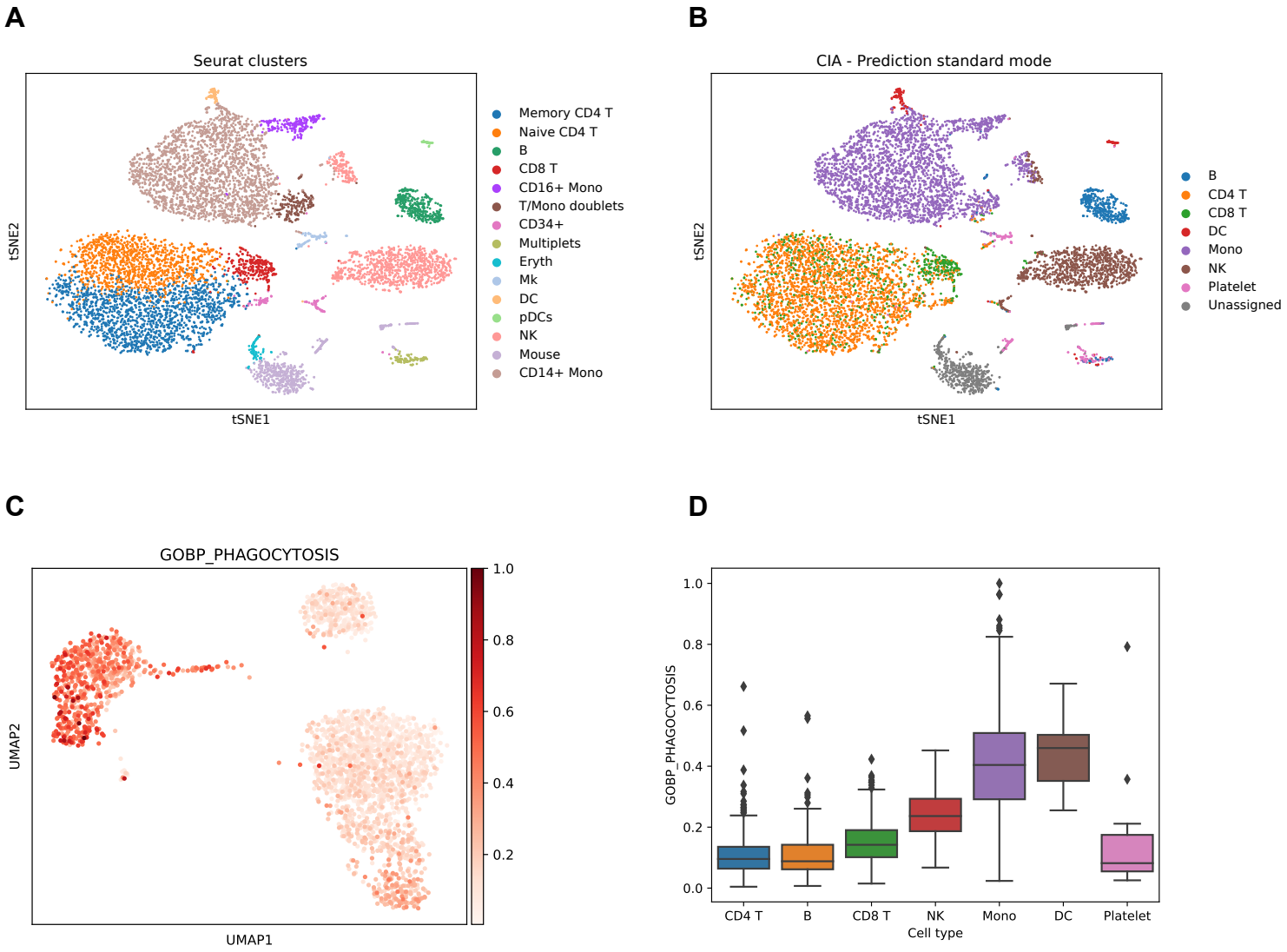

Fig3 A) TSNE plot showing the CBMC dataset with the ground truth annotation. B) TSNE plot of CBMC dataset showing cell labels predicted by CIA "standard mode". C) UMAP plot of PBMC3k showing the scaled signature scores of "GOBP_PHAGOCYTOSIS". D) Boxplot of the scaled signature score distributions of "GOBP_PHAGOCYTOSIS" grouped by ground truth labels.