nature methods

ARTICLES

https://doi.org/10.1038/s41592-020-0825-9

Putative cell type discovery from single-cell gene expression data

Zhichao Miao[1,2], Pablo Moreno[1], Ni Huang[1,2], Irene Papatheodorou[1], Alvis Brazma[1]✉ and Sarah A. Teichmann[2,3]✉

We present the Single-Cell Clustering Assessment Framework, a method for the automated identification of putative cell types from single-cell RNA sequencing (scRNA-seq) data. By iteratively applying a machine learning approach to a given set of cells, we simultaneously identify distinct cell groups and a weighted list of feature genes for each group. The differentially expressed feature genes discriminate the given cell group from other cells. Each such group of cells corresponds to a putative cell type or state, characterized by the feature genes as markers. Benchmarking using expert-annotated scRNA-seq datasets shows that our method automatically identifies the 'ground truth' cell assignments with high accuracy.

Identifying cell types in multicellular organisms and understanding the relationships between them has been a major aim of biological research since the discovery of cells by Robert Hooke almost 400 years ago[1]. Historically, cell types have been defined by their morphology as assessed by microscopy, by their locations in an organism, by their function in vivo or in vitro, or by their developmental and evolutionary history[2,3]. Data from scRNA-seq is one of the latest sources of information for the discovery of new putative cell types and refining the classification of existing ones[4]. Defining and identifying all cell types in a human body is one of the goals of the Human Cell Atlas project, which aims to apply scRNA-seq to a representative sample of all human cells[5,6].

A typical way of using scRNA-seq data for putative cell type identification relies on marker gene-based manual annotation after supervised clustering. To assist manual annotation, computational analysis tools, such as Single-Cell Analysis in Python (SCANPY)[7] and Seurat[8] cluster cells by different methods[9,10] and visualize the clustering using dimensionality reduction methods. To date, a large number of scRNA-seq datasets from mouse, human and other organisms have been annotated this way and made publicly available[11–14]. Since the number of cell types in a new dataset is unknown a priori, methods to assess the likely number of clusters in data have been developed[15] and distance-based cluster merging has been proposed to optimize clustering[16]. These methods cannot guarantee that the resulting clusters represent biologically distinct groups of cells corresponding to putative cell states or types. A careful manual inspection of each dataset is not scalable, thus, being a bottleneck in atlas projects like the Human Cell Atlas. To facilitate the annotation of new datasets, various automated methods that exploit previously identified cell types and knowledge of their markers have been proposed[17–25]. However, these reference-based methods cannot discover new, yet unannotated types of cells or new cell states.

In this study, we propose an automated method that allows for the potential discovery of novel, not-yet-annotated putative cell types. We treat the terms 'putative cell type' and 'cell state' loosely as synonyms referring to a biologically meaningful group of cells characterized by a specific pattern of gene expression. Our method, Single-Cell Clustering Assessment Framework (SCCAF), is based on the iterative application of machine learning and self-projection[14] to clusters, thus gradually merging clusters that can be defined by similar sets of feature genes.

The validity of our approach is based on three assumptions. First, as in any scRNA-seq approach, we assume that each putative cell type is defined by a specific RNA expression profile. Second, we assume that the published, human expert-annotated scRNA-seq datasets represent the ground truth. Third, we assume that if an automated method accurately reconstructs human annotation in diverse scRNA-seq datasets, it will also probably work well on new datasets. We test our approach on simulated and real scRNA-seq datasets from a range of experiments and empirically conclude that our method can reconstruct the cell types as annotated by human experts with high accuracy. Moreover, we use a 'transparent-box' classifier, building a computational model defining the respective cell group via a weighted list of feature genes. We find that our feature genes often include the known marker genes for the characterized cell types.

Results

A self-projection-based approach. The input to our method is a matrix of gene expression values as measured in an scRNA-seq experiment, with each column representing a cell and each row representing a gene (Fig. 1a). First, a clustering algorithm (for example, *k*-means, Louvain[10] or Leiden[9] clustering) is applied to the columns (that is, to the cells); the clustering can be based either on the entire set of genes, highly variable genes or done in the principal component space using a chosen number of principal components. Then, each cluster is split into training and test datasets, and a classifier is trained and then applied to the test dataset to measure how well the model discriminates between the cells in the particular cluster over all other cells. Comparing the predicted clusters to the original ones is known as self-projection[14]. Self-projection accuracy is defined as the percentage of correctly predicted cells in the entire dataset and can be used to assess the reliability of clustering. The comparison between the predicted cluster and the actual clusters in the test dataset gives us the confusion rate for each cluster and the respective confusion matrix (see Methods). If the data is 'overclustered', that

[1]European Bioinformatics Institute, European Molecular Biology Laboratory, Wellcome Genome Campus, Cambridge, UK. [2]Wellcome Sanger Institute, Wellcome Genome Campus, Cambridge, United Kingdom. [3]Department of Physics, Cavendish Laboratory, University of Cambridge, Cambridge, UK. ✉e-mail: brazma@ebi.ac.uk; st9@sanger.ac.uk

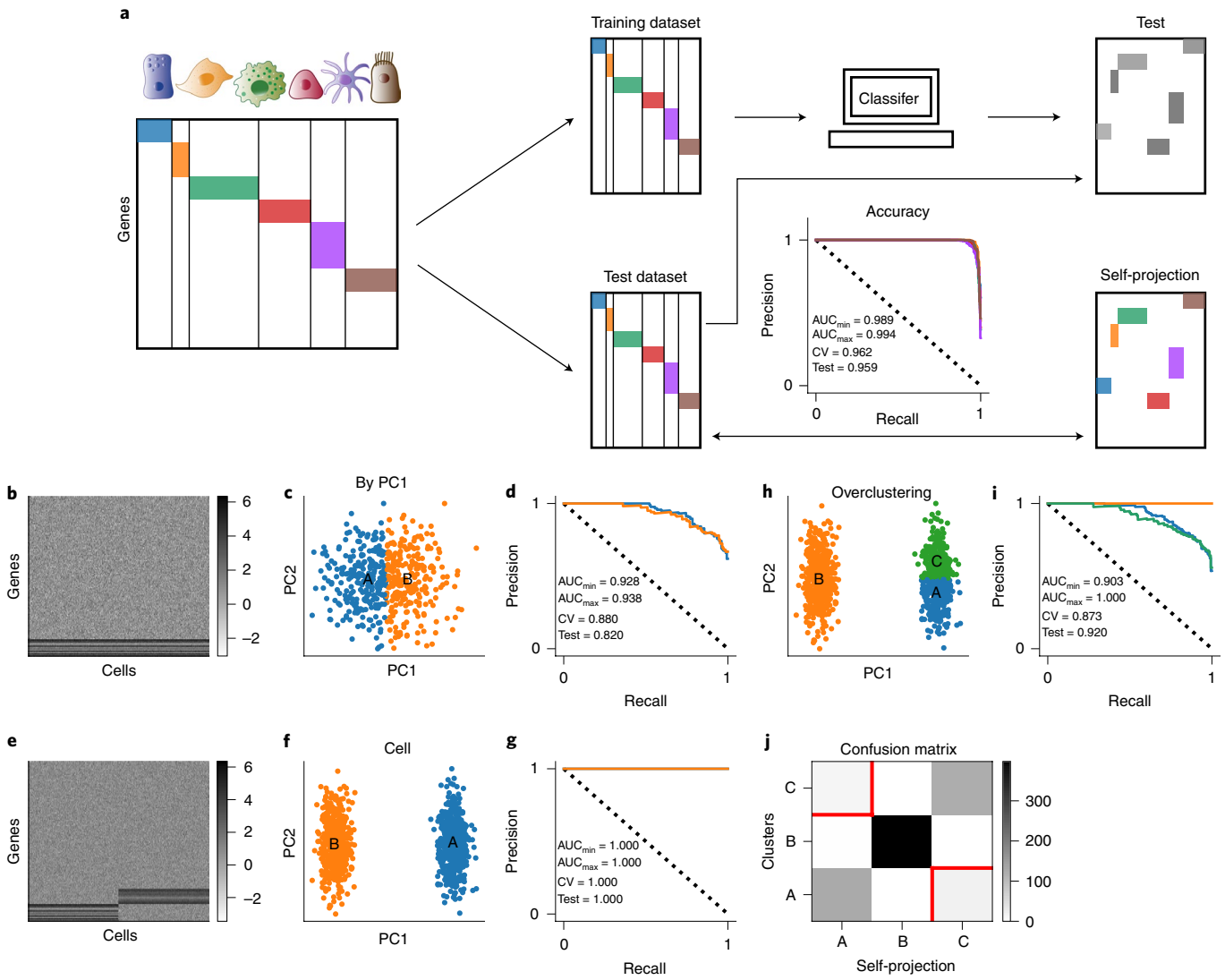


Fig. 1 | A self-projection approach. a, Scheme of machine learning-based self-projection: (1) the data are randomly split into training and test datasets; (2) the assigned clusters are used to train a machine learning classifier with cross-validation on the training set; (3) the machine learning model is applied to the test dataset, which is considered as 'self-projection'; (4) the consistency between the original cluster assignment of the test dataset and the self-projection result measures the reliability of the clustering. **b**, To validate this theory, we first simulated a single cell type of 500 cells with a number of feature genes expressed three times higher than background genes on average. **c**,**d**, When the cells are divided into two clusters based on principal component 1 (PC1) (**c**), the precision-recall curve logistic regression shows certain but not ideal predictive ability in self-projection (**d**). **e**, Next, we simulated two cell types each defined by a set of feature genes that are nonoverlapping (500 cells for each type). **f**, The PCA of these data is shown. **g**, The self-projection shows perfect predictive ability. **h**, However, if the clustering tries to find 3 clusters, the cells of one of the types are overclustered into 2 clusters (250 cells in clusters A and C, 500 cells in cluster B). **i**, The overclustered cell type (A and C) shows lower accuracy. **j**, The confusion matrix shows the size of overlaps between the original clusters and the classes predicted by the trained model. The tests were repeated 10 times independently with similar results in all the simulations. The self-projection result demonstrates that confusion always happens between the overclustered clusters, but not between clusters of different cell types.

is, if two or more clusters in fact represent the same type of cells, the classifier will not be able to discriminate between these clusters and the respective confusion rate will be high. Detecting 'underclustering' is more challenging, but as our computational experiments show, if a cluster represents a mixture of cell types, classifier performance will typically be poorer than for a cluster representing a well-defined group of cells (Extended Data Fig. 1).

Next, we normalized the confusion matrix (Fig. 2) by computing the ratio of the misclassified cells over the correctly assigned cells to account for clusters of different sizes and then discretized it by applying a maximum confusion threshold derived from the maximum confusion rate of the entire dataset (see Methods). The discretized confusion matrix represents a cluster connection graph (intuitively describing the similarity between clusters); the connected clusters were merged according to this graph. We iterated the machine learning and self-projection approach, merging the connected clusters until the overall self-projection accuracy kept growing, or reached a 98% default cutoff.

Our assumption is that if there is a set of feature genes expressed differentially in a cluster in comparison to other cells in the training dataset, these genes can be discovered by machine learning. Thus, applying the trained model to the test data, the clusters representing putative cell types will be restored reasonably well. If, on the contrary, a model derived from the training half of the dataset is

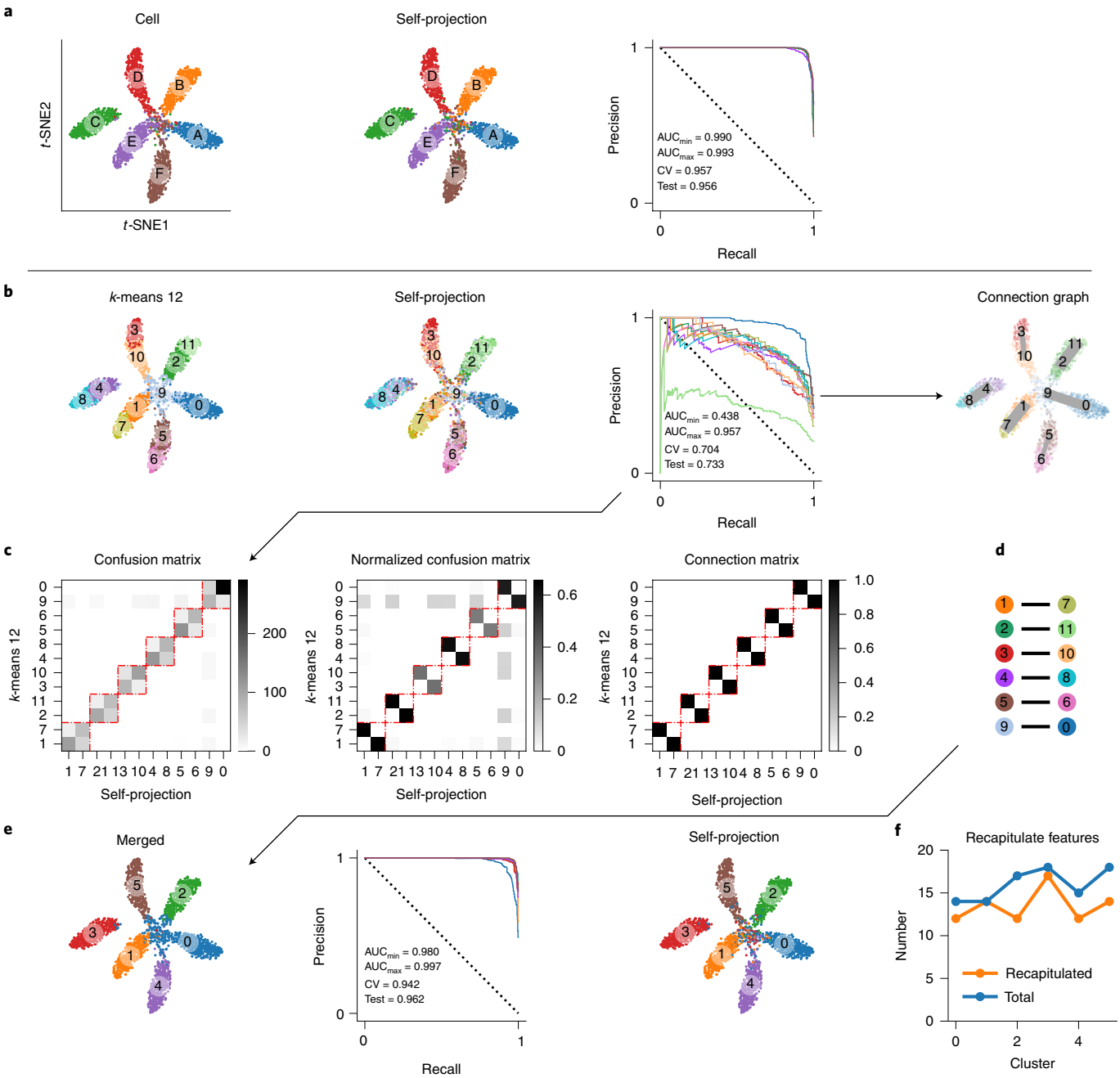


Fig. 2 | Using the connection graph to optimize clustering. a, Simulated dataset (see Methods: multivariate normal simulation) of six 'putative cell types' (500 cells for each cell type, 3,000 simulated cells in total); the *t*-SNE plot is shown. The self-projection of the ideal cluster assignment is identical to the original clustering, with a self-projection accuracy of approximately 94%. **b**, Starting from an overclustered *k*-means clustering of 12 clusters, the confusion rates point to the clusters that represent the same putative cell type. Total accuracy is low at this starting point. **c**, To optimize clustering, we first normalized the confusion matrix based on the number of correctly assigned cells in each cluster. Then, we removed the diagonal values and used a cutoff of normalized confusion rate to binarize the normalized confusion matrix into a connection matrix. **d**, A connection graph was obtained from the connection matrix and was then used to merge some clusters. (We assumed that the logistic regression model would achieve accuracy better than 60%, leaving a confusion rate threshold of 40%. If two cell clusters have a normalized confusion rate higher than 40% in the normalized confusion matrix, then the two cell clusters are connected in the matrix.) SCCAF uses a Louvain clustering[12] of fixed resolution of 1.0 to merge the identical cell clusters based on the binary connection graph. **e**, The merged clusters recovered the initial cell type assignment well, except for some noisy cells in the center of the *t*-SNE plot, which are clustered as cluster 0. Finally, the self-projection of the logistic regression model optimizes the cell cluster assignment of the noisy cells. The optimized result was assessed by the SCCAF self-projection and attained accuracies of approximately 94% on both cross-validation and test dataset, which is identical to the original simulated cell clusters in **a**. The top-weighted features (feature genes) captured by the logistic regression model are then compared with the feature genes that were used to simulate the cell clusters. **f**, The majority of feature genes are recapitulated by the logistic regression model. The total number of feature genes and the number of recapitulated are plotted as blue and orange, respectively.

'confused' (unable to find 'good' discriminative genes), this indicates that there is no good set of feature genes whose expression defines this cluster unambiguously. In the context of expression data, a cluster without discriminative genes cannot be considered as a biologically coherent and distinct group of cells. Specifically, in our approach, we used logistic regression and fivefold cross-validation (hence, by default we required a minimum of ten cells in a cluster), which was shown by Ntranos et al.[26] as a fast and effective method for capturing differentially expressed genes. Four other machine learning models (support vector machine, decision tree, random forest and Gaussian naive Bayes) are also implemented in the SCCAF framework and have been tested by us (Extended Data Fig. 2). In the tests on both simulated and real data (Fig. 1 and Extended Data Fig. 3), the self-projection results indicate that confusion of cell cluster assignments happens almost exclusively between clusters of the same cell type.

Evaluating SCCAF on simulated data. To test our method, we used both simulated and expert-annotated datasets. We used two simulation approaches: first, based on the multivariable normal distribution (see Methods) and second, based on the scRNA-seq data simulator Splatter[27]. We first simulated 3,000 cells of 6 cell types using a multivariable normal distribution. Each cell type included 500 cells of 10–20 feature genes (Fig. 2a). A logistic regression model was trained on half of the dataset and applied to the whole dataset. The original cell clusters and the projection results are plotted in Fig. 2a as *t*-distributed stochastic neighbor embedding (*t*-SNE)[28]. The projection result was identical to the original assignment, since the cell type-related features are captured in the logistic regression model, and the self-projection assessment accuracy was above 94%.

Since logistic regression weighs each gene in a linear model, we extracted the top-weighted genes and compared them with the feature genes used in the simulation (Fig. 2f). The top-weighted genes recovered >75% of the simulated feature genes and both lists overlapped well. While this result is not surprising, since our simulation method and a logistic regression model are well-matched, the multivariate simulation approach has been shown previously[26] to capture important features of scRNA-seq data. Further tests on simulated and real datasets can be found in the Supplementary Information.

Testing the method on expert-annotated scRNA-seq datasets. To assess to what extent SCCAF recovers expert-annotated groups of cells, we tested the method on eight published and annotated scRNA-seq datasets. First, we used the mouse retina dataset from Shekhar et al.[11]. For this dataset, the outputs from Louvain clustering in the principal component space at a resolution of 1.0 (Fig. 3a) show overclustering of the rod bipolar cells, while the resolution 0.3 clustering shows both underclustering and overclustering in comparison to the expert annotation. SCCAF starts from Louvain clustering resolution 1.0 and goes through four rounds of optimization (Extended Data Fig. 4). The cluster connection graph in the first round of the SCCAF optimization (Fig. 3b) shows that the clusters containing cells of the same cell type (as described in the text) are highly confused. By merging the clusters, SCCAF gradually reduces the normalized confusion rate until the total self-projection accuracy exceeds the chosen cutoff of 95%. The accuracy of the cross-validation and of the test dataset continuously increase during the four rounds of optimization from 61 to 98% (Fig. 3c). As the *t*-SNE plots in Fig. 3d show, the SCCAF confusion matrix-based cluster optimization achieves a similar result to the 'ground truth' published cell annotation. The river plot comparing SCCAF-optimized clustering and the expert-annotated dataset in Fig. 3e shows that they are almost perfectly matched; the adjusted Rand index[29] is over 0.99. Thus, in this dataset, the 'ground truth' can be recovered almost automatically with high accuracy. Furthermore, the feature genes that discriminate the cell clusters may facilitate any subsequent manual annotation of the clusters.

To test our method more broadly, we used seven additional published datasets annotated by experts, assuming that the annotated cell types represented the 'ground truth'. Specifically, we used human and mouse pancreas datasets from Baron et al.[12], a mouse cortex dataset from Zeisel et al.[30], a retinal bipolar neuron dataset from Shekhar et al.[11], a Smart-Seq2 human pancreatic islets dataset from Segerstolpe et al.[13], an inDrop mouse visual cortex dataset from Hrvatin et al.[31], an mCEL-seq2 human liver dataset from Aizarani et al.[32] and a Smart-Seq mouse cortex dataset from Tasic et al.[33]. Cell type annotations were obtained from the original publications. In almost all cases, SCCAF achieved clusterings close to the manual annotation of published clusters (Extended Data Figs. 5 and 6), with an average adjusted Rand index >0.94.

A key challenge in scRNA-seq data analysis is finding the correct number of cell clusters in a dataset. For instance, SC3 (ref. [15]) clustering attempts to estimate the optimal number of cell clusters using Tracy–Widom distribution[34] on random matrices. In Fig. 4, the self-projection clustering optimization process starts from a Louvain clustering of resolution 3.5, where many cells are overclustered. The self-projection clustering optimization stops after four rounds when self-projection accuracy is >96%. We further lowered the confusion rate threshold until the results were underclustered (Fig. 4b). For all the clustering 'snapshots' throughout 9 rounds of optimization, self-projection clustering assessment was repeated 100 times by random sampling the training and test datasets. According to the distribution of the self-projection accuracy, optimal clustering shows better accuracy than overclustering or underclustering. Additionally, the standard deviation of the self-projection accuracy on random sampling is smaller for optimal clustering than other clustering solutions (that is, overclustering or underclustering, or intermediate optimization rounds). We also tested our method on a Smart-seq2 dataset, specifically the pancreas islet cell dataset[13], which revealed similar results (Fig. 4c,d). Our computational experiments (Extended Data Fig. 1) show that in real-world datasets, self-projection accuracy is the highest for the cell type assignments corresponding to the ground truth in almost all cases and that in most cases, the iterations can run until the self-projection accuracy stops increasing; only in some rare cases, user intervention is needed.

SCCAF defines cell states in erythroid maturation. A recent mouse hematopoietic stem cell differentiation dataset from Tusi et al.[35] characterized cell states in a continuous differentiation process. Applying SCCAF to a dataset describing a continuous process leads to clusters that correspond to the most populated regions in the differentiation trajectory. We sought to define these major cell states during differentiation by starting from a Louvain clustering at a resolution of 1.5 (Fig. 5a). This results in an optimized clustering corresponding to 12 putative cell states (Fig. 5b), with a self-projection assessment accuracy of 92% on the test data. Several cell clusters are closely related to their annotation reported in the literature (Extended Data Fig. 7). The logistic regression model derived from SCCAF shows good discrimination of all cell clusters (Fig. 5c). The top-weighted feature genes in the logistic regression model were retrieved and 35 of them were reported as marker genes in the publication[35] and used as probes in the reverse transcription–PCR validation (Fig. 5d).

The erythrocytes further cluster into three cell states: committed erythroid progenitors (CEP), also described in the literature, and early and late erythroid terminal differentiated cells (early and late erythroid terminal differentiation), which we propose as new distinct states of erythroid maturation. These three cell states (1, 7 and 6) show clear separation in the principal component space (Fig. 5e) by the first two components, indicating clear mathematical

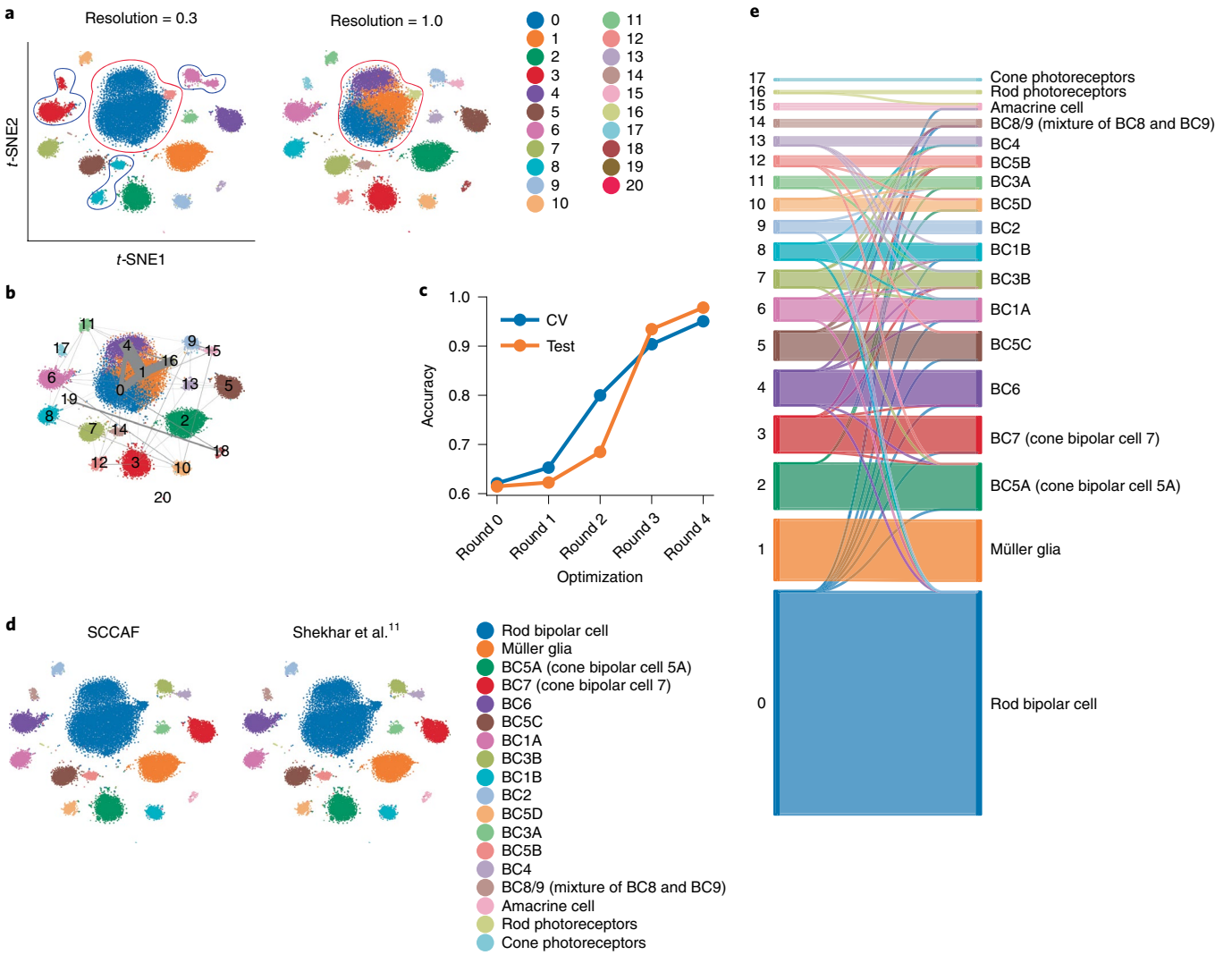


Fig. 3 | Self-projection-based clustering optimization compared with ground truth. a, In a mouse retina dataset[11] of 27,499 cells, Louvain clustering with a resolution of 0.3 and 1.0 results in overclustering of the rod bipolar cells (red circle) and underclustering (blue circle). The SCCAF clustering optimization starts from a Louvain clustering of resolution 1.0. SCCAF merges the cell clusters based on the confusion matrix derived from the machine learning model. **b**, The confusion matrix-derived connection graph of the first round optimization indicates that clusters 0, 1, 4 and 16 are connected. The optimization process increases the consistency between the clustering assignment and the model prediction. **c**, Starting from a Louvain clustering resolution of 3.5, the cross-validation accuracies and the accuracies on the test datasets increase over the four rounds of clustering optimization. **d**,**e**, The *t*-SNE plots show that the optimized result is identical to an ideal annotation (**d**). A river plot shows the match between SCCAF result and the published cell type annotation (**e**).

discrimination on their transcriptomes. Upregulated and downregulated genes between these states (Extended Data Fig. 8) provide further confidence in their biological relevance.

Finally, we projected the logistic regression models trained on the data from Tusi et al.[35] to an independent mouse hematopoiesis dataset from Giladi et al.[36]. The majority of the cell populations were recapitulated and a uniform manifold approximation and projection (UMAP)[37,38] plot shows an identical distribution (Fig. 5f). The erythroid and granulocytic neutrophil branches show the same order of cell clusters (2 to 4 to 1 to 7 to 6 and 3 to 0 to 5). Focusing on the three erythroid states in the Giladi et al.[36] dataset (Fig. 5g), the self-projection accuracy was 90%, indicating little confusion between the states. During the maturation process (Fig. 5h), the cells in the CEP and early erythroid terminal differentiation stages are cycling and include more genes (larger cells), whereas the late erythroid terminal differentiation stage stops proliferating and the hemoglobins are expressed. Details about marker gene analysis are included in the Supplementary Information.

Discussion

We demonstrated that SCCAF restores the expert-annotated cell type assignments with high accuracy. Since SCCAF associates each of the discovered cell types with a ranked list of feature genes defining this cell type, it effectively provides an initial cell type annotation. Moreover, the associated gene lists can then be used to derive a biologically meaningful cell type annotation, either by a human expert or by applying one of the published automated reference-based methods[21,22]. We also demonstrated that in datasets providing temporal information about cell type/state progression, for instance, during cell differentiation, SCCAF could restore biologically meaningful cell states (Fig. 5).

We have compared the results from SCCAF with cell type annotations obtained from reference-based methods on previously published expert-annotated datasets[31–33]. SCCAF finds the correct cell groups, often outperforming the tested reference-based methods (Supplementary Notes). We have also shown that a model (that is, the weighted gene lists) trained by SCCAF on one dataset, can then

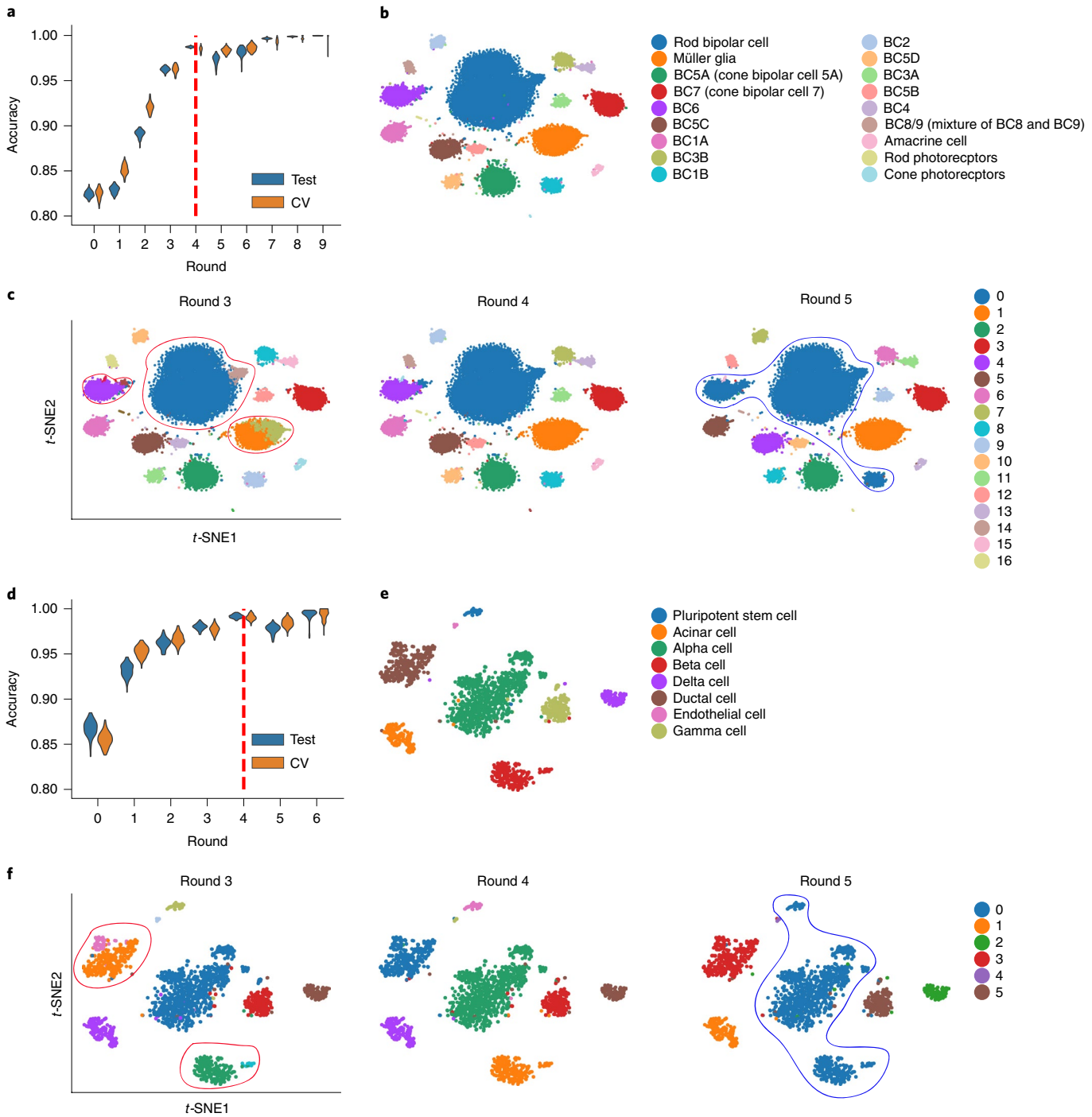


Fig. 4 | Self-projection accuracy indicates optimal clustering during clustering optimization. a–f, In the Shekhar et al.[11] (26,830 cells; **a**–**c**) and Segerstolpe et al.[13] (2,108 cells; **d**–**f**) datasets, we tested the self-projection accuracies in each round of optimization by repeating the random sampling 100 times (**a**,**d**). The violin plots show the distributions of the self-projection accuracies in cross-validation (orange) and on the test dataset (blue) in each round of the SCCAF optimization. The self-projection optimization result after four rounds is most similar to the gold standard annotation from the publication (**b**,**e**). The clusterings in rounds 3, 4 and 5 in the Shekhar et al.[11] (26,830 cells) and Segerstolpe et al.[13] (2,108 cells) datasets, respectively, are shown in **c** and **f**. Overclustering exists in the results of the third round (red circles) and underclustering can be found in the fifth round (blue circles) results. The self-projection-based clustering optimization stops after four rounds, while further merging of clustering can happen when we lower the confusion rate cutoff. Optimal clustering demonstrates better self-projection accuracy than cases with overclustering or underclustering.

be successfully applied to an independent dataset of the same tissue to classify the cells automatically (Fig. 5). Moreover, when a reference is only available for a different organism (for example, comparing human and mouse brain), the model that SCCAF builds can be used for cross-species comparisons (Extended Data Fig. 9).

Cell type taxonomy is often presented as a hierarchy, but such representations are only approximations of biological reality, since they overlook the temporal aspect of cell development or progressive transitions, as well as cell state convergence[39,40]. Most scRNA-seq experiments typically study cells at a particular level

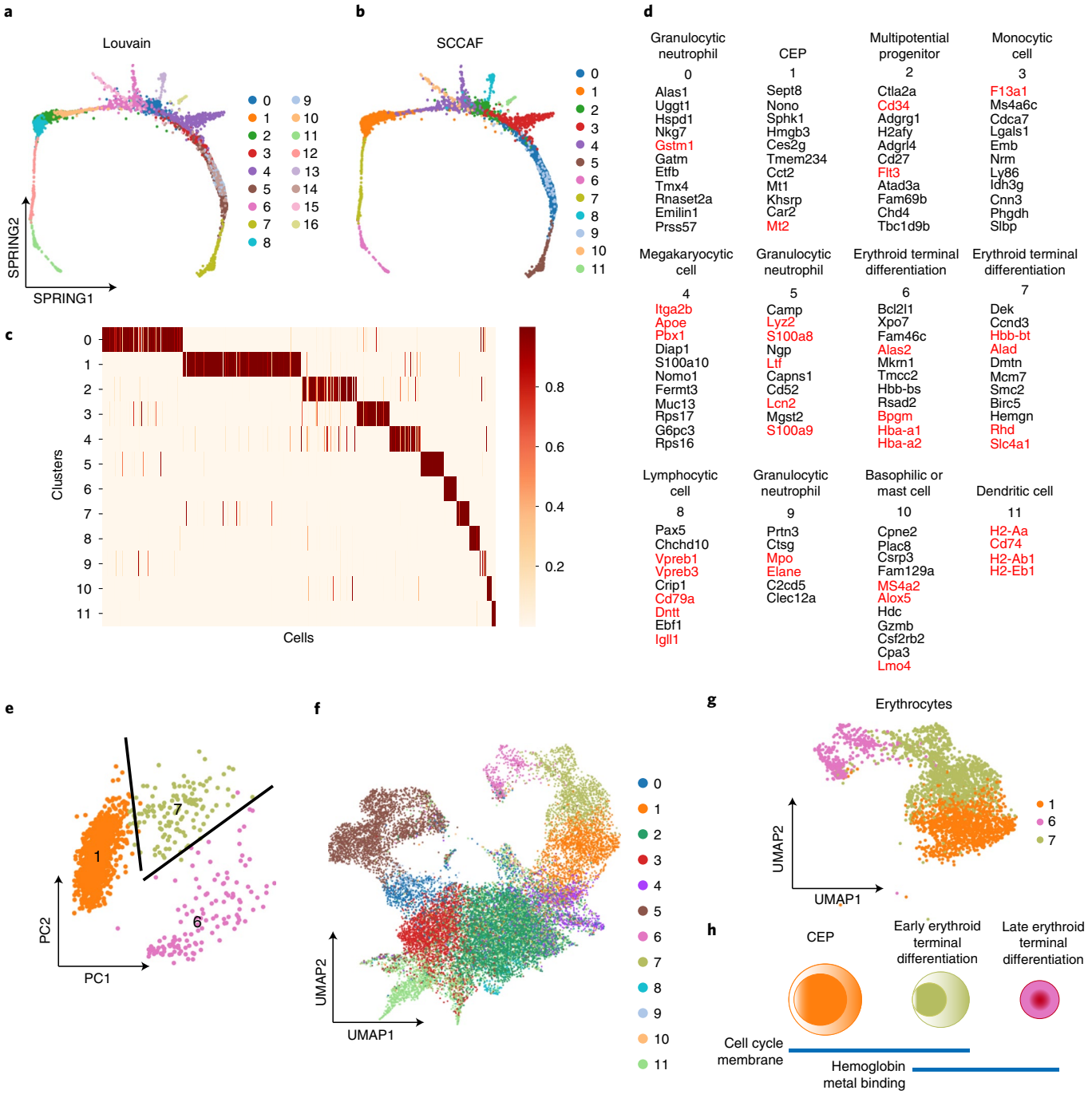


Fig. 5 | SCCAF captures the key stages in mouse hematopoiesis. a,**b**, Mouse hematopoiesis data (4,016 cells) from Tusi et al.[35] is clustered with SCCAF. The SCCAF clustering optimization starts from Louvain clustering (**a**) and is merged into 12 cell clusters (**b**). **c**, The resulting cell clusters are highly discriminative in the logistic regression model. **d**, The features encoded in the logistic regression model captured many of the known marker genes reported in previous publications. The top-ranked features are listed and known marker genes are highlighted in red. The cell clusters correspond to different cell types. **e**, Furthermore, the erythrocytes (1,469 cells) are further clustered into 3 subpopulations, which can be clearly separated in the PCA spaces. These three subpopulations correspond to CEPs, and early and late erythroids. **f**, Using the logistic regression model trained on the Tusi et al.[35] dataset and applied to another mouse hematopoiesis dataset (20,202 cells) from Giladi et al.[36], most of the cell groups are recapitulated. **g**,**h**, The separation of the 3 erythroid stages (2,953 cells) of different biological functions (**h**) are well captured, while the self-projection accuracy is 90% on the Giladi et al.[36] dataset (**g**).

of resolution; SCCAF is designed to uncover the most appropriate 'flat' classifications for the particular level of resolution. Nevertheless, we showed that if a dataset provides sufficient variances in gene expression, then by applying SCCAF iteratively, we can sometimes reconstruct parts of the underlying cell type hierarchy (Extended Data Fig. 10).

Importantly, the performance of our method is not notably affected by the size of the clusters; thus, it can be used to discover rare cell types, as long as they are detected by the initial clustering. In the real-world datasets we tested, the smallest cluster was 7 cells, but more typically the smallest cell clusters are around 30 cells, a regime where SCCAF works well (Supplementary Notes). There is

always a compromise to be struck between the minimal number of cells in a group that is used to define a cell type versus the potential introduction of noise.

The limitation of our method may reside in the underclustering cases. Since a machine learning classifier performs better when discriminating fewer clusters, the discrimination of underclustering is not always clear (Extended Data Fig. 1). In some cases, manual inspection may be required to stop cluster merging. When 'noisy cells' exist in the initial clustering, SCCAF merges all the clusters similar to these 'noisy cells' and may cause underclustering in the results. Further discussions about noisy cells and underclustering have been included in the Supplementary Notes.

We have experimented with different types of classifiers in the machine learning step of SCCAF and found that a regularized linear regression outperformed more sophisticated nonlinear classifiers on both simulated and real-world datasets. Although it has been demonstrated previously that the linear regression classifiers can discriminate between cell types with good accuracy[26], this observation may appear surprising. One can argue that this is a consequence of how cell types are currently defined in the context of scRNA-seq data (that is, via highly expressed marker genes), which is also consistent with how scRNA-seq data is usually clustered (either based on highly variable genes or principal components in expression space). It is an open question whether there is an alternative biologically meaningful cell type definition (for instance, based on decision trees), for which nonlinear classifiers would outperform linear approaches.

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41592-020-0825-9.

Received: 15 July 2019; Accepted: 2 April 2020;
Published online: 18 May 2020

References

1. Hooke, R. *Micrographia: or Some Physiological Descriptions of Minute Bodies Made by Magnifying Glasses. With Observations and Inquiries Thereupon* (J. Martyn and J. Allestry, 1665).
2. Arendt, D. et al. The origin and evolution of cell types. *Nat. Rev. Genet.* **17**, 744–757 (2016).
3. Nagasawa, T. Microenvironmental niches in the bone marrow required for B-cell development. *Nat. Rev. Immunol.* **6**, 107–116 (2006).
4. Trapnell, C. Defining cell types and states with single-cell genomics. *Genome Res.* **25**, 1491–1498 (2015).
5. Rozenblatt-Rosen, O., Stubbington, M. J. T., Regev, A. & Teichmann, S. A. The Human Cell Atlas: from vision to reality. *Nature* **550**, 451–453 (2017).
6. Regev, A. et al. The Human Cell Atlas. *eLife* **6**, e27041 (2017).
7. Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol.* **19**, 15 (2018).
8. Butler, A., Hoffman, P., Smibert, P., Papalexi, E. & Satija, R. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat. Biotechnol.* **36**, 411–420 (2018).
9. Traag, V. A., Waltman, L. & van Eck, N. J. From Louvain to Leiden: guaranteeing well-connected communities. *Sci. Rep.* **9**, 5233 (2019).
10. Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. *J. Stat. Mech: Theory Exp.* **2008**, P10008 (2008).
11. Shekhar, K. et al. Comprehensive classification of retinal bipolar neurons by single-cell transcriptomics. *Cell* **166**, 1308–1323.e30 (2016).
12. Baron, M. et al. A single-cell transcriptomic map of the human and mouse pancreas reveals inter- and intra-cell population structure. *Cell Syst.* **3**, 346–360.e4 (2016).
13. Segerstolpe, Å. et al. Single-cell transcriptome profiling of human pancreatic islets in health and type 2 diabetes. *Cell Metab.* **24**, 593–607 (2016).
14. Han, X. et al. Mapping the Mouse Cell Atlas by Microwell-Seq. *Cell* **173**, 1307 (2018).
15. Kiselev, V. Y. et al. SC3: consensus clustering of single-cell RNA-seq data. *Nat. Methods* **14**, 483–486 (2017).
16. Zhang, J. M., Fan, J., Fan, H. C., Rosenfeld, D. & Tse, D. N. An interpretable framework for clustering single-cell RNA-Seq datasets. *BMC Bioinformatics* **19**, 93 (2018).
17. de Kanter, J. K., Lijnzaad, P., Candelli T., Margaritis, T. & Holstege, F. C. P. CHETAH: a selective, hierarchical cell type identification method for single-cell RNA sequencing. *Nucleic Acids Res.* **47**, e95 (2019).
18. Xie, P. et al. SuperCT: a supervised-learning framework for enhanced characterization of single-cell transcriptomic profiles. *Nucleic Acids Res.* **47**, e48 (2019).
19. Pliner, H. A., Shendure, J. & Trapnell, C. Supervised classification enables rapid annotation of cell atlases. *Nat. Methods* **16**, 983–986 (2019).
20. Zhang, A. W. et al. Probabilistic cell-type assignment of single-cell RNA-seq for tumor microenvironment profiling. *Nat. Methods* **16**, 1007–1015 (2019).
21. Aran, D. et al. Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nat. Immunol.* **20**, 163–172 (2019).
22. Tan, Y. & Cahan, P. SingleCellNet: a computational tool to classify single cell RNA-seq data across platforms and across species. *Cell Syst.* **9**, 207–213.e2 (2019).
23. Wagner, F. & Yanai, I. Moana: a robust and scalable cell type classification framework for single-cell RNA-Seq data. Preprint at *bioRxiv* https://doi.org/10.1101/456129 (2018).
24. Ma, F. & Pellegrini, M. ACTINN: automated identification of cell types in single cell RNA sequencing. *Bioinformatics* **36**, 533–538 (2020).
25. Lin, Y. et al. scClassify: hierarchical classification of cells. Preprint at *bioRxiv* https://doi.org/10.1101/776948 (2019).
26. Ntranos, V., Yi, L., Melsted, P. & Pachter, L. A discriminative learning approach to differential expression analysis for single-cell RNA-seq. *Nat. Methods* **16**, 163–166 (2019).
27. Zappia, L., Phipson, B. & Oshlack, A. Splatter: simulation of single-cell RNA sequencing data. *Genome Biol.* **18**, 174 (2017).
28. Dimitriadis, G., Neto, J. P. & Kampff, A. R. *t*-SNE visualization of large-scale neural recordings. *Neural Comput.* **30**, 1750–1774 (2018).
29. Hubert, L. & Arabie, P. Comparing partitions. *J. Classif.* **2**, 193–218 (1985).
30. Zeisel, A. et al. Brain structure. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347**, 1138–1142 (2015).
31. Hrvatin, S. et al. Single-cell analysis of experience-dependent transcriptomic states in the mouse visual cortex. *Nat. Neurosci.* **21**, 120–129 (2018).
32. Aizarani, N. et al. A human liver cell atlas reveals heterogeneity and epithelial progenitors. *Nature* **572**, 199–204 (2019).
33. Tasic, B. et al. Shared and distinct transcriptomic cell types across neocortical areas. *Nature* **563**, 72–78 (2018).
34. Tracy, C. A. & Widom, H. Level-spacing distributions and the Airy kernel. *Comm. Math. Phys.* **159**, 151–174 (1994).
35. Tusi, B. K. et al. Population snapshots predict early haematopoietic and erythroid hierarchies. *Nature* **555**, 54–60 (2018).
36. Giladi, A. et al. Single-cell characterization of haematopoietic progenitors and their trajectories in homeostasis and perturbed haematopoiesis. *Nat. Cell Biol.* **20**, 836–846 (2018).
37. McInnes, L., Healy, J., Saul, N. & Großberger, L. UMAP: Uniform Manifold Approximation and Projection. *J. Open Source Softw.* **3**, 861 (2018).
38. Becht, E. et al. Dimensionality reduction for visualizing single-cell data using UMAP. *Nat. Biotechnol.* **37**, 38–44 (2019).
39. Konstantinides, N. et al. Phenotypic convergence: distinct transcription factors regulate common terminal features. *Cell* **174**, 622–635.e13 (2018).
40. Gerber, T. et al. Single-cell analysis uncovers convergence of cell identities during axolotl limb regeneration. *Science* **362**, eaaq0681 (2018).

Methods

Machine learning-based self-projection. As shown in Fig. 1a, the expression profile and cluster assignment are first split into training and test datasets. If a cluster of cells includes more than 200 cells, 100 cells are randomly selected from a cluster and used as a training set, while the rest of the cells are used as the test dataset. When a cluster contains between 10 and 200 cells, half of the cells are randomly selected as the training dataset and the other half is used as the test dataset. Given that the algorithm uses machine learning and fivefold cross-validation to build a classifier, if the cluster contains fewer than ten cells, but at least six cells, our algorithm splits the cluster asymmetrically, so that five cells are included in the training dataset, while the remaining ones are included in the test dataset. Two parameters, the maximum number of cells used for training and the fraction of cells used for training, have been implemented to adjust the training/testing ratio. We trained a multi-class machine learning classifier based on the training dataset. Taking the advantage of sklearn (version 0.22.1)[41], we implement five different machine learning models: logistic regression, random forest, Gaussian process classification, support vector machine and decision tree. In the case of logistic regression, we used 'L1' regularization to avoid overfitting. Fivefold cross-validation, which is implemented in sklearn.model_selection, was applied to the training dataset. The average value of the cross-validation accuracies was used as the accuracy of the cross-validation. The model trained on the training dataset is then applied to the test dataset and the predicted results of the test dataset are compared with its original clustering. Self-projection accuracy is defined as the percentage of correctly predicted cells in the test dataset. It can be considered as a metric to assess clustering reliability.

The precision-recall curves were calculated by the sklearn.metrics.precision_recall_curve function. The receiver operating characteristic curves were calculated on the relationship between the false positive and true positive rates using the sklearn.metrics.roc_curve function. The area under the curve (AUC), calculated using the sklearn.metrics.auc function, was used as a metric to benchmark the accuracy of a prediction model.

Confusion matrix-directed cluster optimization. Supplementary Fig. 1 describes the whole workflow of SCCAF optimization. SCCAF uses the confusion matrix[42] obtained from the test dataset to identify cell clusters that probably include cells of the same type. The confusion matrix C is an $n \times n$ matrix, where n is the number of clusters, and the elements of the matrix $c_{i,j}$ represent the number of cells in cluster j but predicted to belong to cluster i. Thus, $c_{i,i}$ is the number of cells in cluster i also predicted as belonging to i. We refer to $c_{i,j}$ as the confusion rate. The confusion matrix is calculated by the sklearn.metrics.confusion_matrix function based on the clustering assignment of the test dataset and the predicted clustering from the machine learning model.

This confusion matrix is then normalized. We defined the normalized confusion rate $r(i,j)$ between clusters i and j as the maximum of ratios of misclassified and correctly classified cells as follows:

$$r(i,j) = \max\left\{\frac{C_{i,j}}{C_{i,i}}, \frac{C_{j,i}}{C_{j,j}}\right\}$$

Intuitively, $r(i,j)$ accounts for the confusion rate relative to the correctly assigned cell numbers in a cluster. For example, $r = 0.3$ means that 30% of the cells are confused between clusters. A high normalized confusion rate indicates that clusters i and j probably represent cells of the same type.

The normalized confusion matrix is then binarized into a connection matrix by a threshold of normalized confusion rate. The threshold is defined as $r_{\text{threshold}} = \max\{r(i,j)\} - 0.01$, which is 1% lower than the maximum normalized confusion rate of the current clustering. An example of the normalization that corresponds to the simulated data in Fig. 2 is shown in Supplementary Fig. 2. The connection matrix is then converted into a connection graph using the igraph Python library implemented in SCANPY (version 1.4.4). Merged groups are obtained by applying the Louvain clustering algorithm on the generated connection graph. The whole cluster merging optimization process is performed iteratively until the preset self-projection accuracy is achieved.

During clustering optimization, graphical plots can be output to show the clustering assignments, the self-projection result, the precision-recall curves, the confusion matrix and the normalized confusion matrices. All graphical plots were generated by SCANPY and matplotlib (version 3.1.3)[43].

Data simulation. In the theory test, two types of data simulations were used: a multivariate normal simulation and the Splatter[27] simulation.

Multivariate normal distribution simulation. We first simulated data with a multivariate normal distribution using the 'multivariate_normal' function in Scipy (version 1.4.1)[44] using the same approach as outlined by Svensson et al.[45]. Gene expression profiles $x = (x_1, \ldots, x_N)$ for each gene followed a normal distribution:

$$X \approx \mathcal{N}(\mu, \Sigma)$$

The background genes had an average expression value of 1 and marker genes had a mean expression value of 3. Each cell state included a random number of marker genes between 10 and 20, while 100 background genes were added besides all marker genes.

Splatter simulation. We use the splatter[27] program to simulate data in a more realistic way. The default parameters for simulation were estimated using the splatEstimate function. The differential expression parameters were set to group.prob = 0.5, de.prob = 1, de.facLoc = 0.1, de.facScale = 0.5 and nGenes = 200. The function splatSimulateGroups was used to simulate the data.

Extracting marker genes based on the logistic regression model. We obtained the weight for each gene in each of the cell clusters from the '.coef_' parameter of the logistic regression model. Only positive weights were extracted. The weights were then sorted in decreasing order, while the top 20 ranked genes were extracted as potential feature genes.

Datasets and data processing. *Mouse retina dataset.* We downloaded the digital gene expression data from Gene Expression Omnibus (GEO) accession code GSE65785, which is referenced in Shekhar et al.[11]. Cells with more than 10% mitochondrial content were excluded. Cells from Bipolar5 and Bipolar6 were assigned as batch 2, while all other cells were assigned as batch 1. The COMBAT function from svaseq (version 3.34.0)[46] was used to correct the batch effect. Cells annotated as doublets/contaminants were excluded from the analysis. One hundred principal components were used to analyze the cell clusters.

Mouse cortex dataset (Zeisel et al.). We downloaded the count matrix together with its annotation of mouse cortex data from Zeisel et al.[30] study from the Hemberg Group scRNA-seq datasets website (https://hemberg-lab.github.io/scRNA.seq.datasets/). We filtered cells expressing fewer than 200 genes and genes expressed in fewer than 3 cells. We used the top 2,000 variable genes to represent the variance of the dataset based on the standard deviation of the genes, similar to the approach outlined by Azizi et al.[47].

Pancreatic islets dataset. We downloaded the processed expression matrix and cell type annotation from ArrayExpress[48] (accession code E-MTAB-5061), which corresponds to the data in Segerstolpe et al.[13]. We removed the uncertain cells annotated as 'not applicable', 'unclassified endocrine cell', 'unclassified cell', 'co-expression cell', 'MHC class II cell' as well as the cell types of fewer than ten cells (mast and epsilon cells). Highly variable genes were selected based on mean expression and dispersions.

Pancreas dataset. The count matrices together with their annotations for mouse and human were downloaded from GEO accession no. GSE84133 as Baron et al.[12]. Cells with fewer than 200 genes or more than 12,000 cells were removed from consideration, while genes expressed in fewer than 3 cells were removed. The linear regression function from NaiveDE (version 1.1.1, https://github.com/Teichlab/NaiveDE) was used to regress out the donor effect as well as the technical variances from the number of genes and counts.

Visual cortex dataset. We downloaded the raw count matrix and the cell type annotation from GEO accession no. GSE102827 as Hrvatin et al.[31]. We filtered out cells of fewer than 200 genes and genes expressed in fewer than 3 cells. Cells annotated as 'nan' were removed from the analysis. Highly variable genes were selected based on mean expression and dispersion.

Hematopoiesis dataset (Tusi et al.). The count matrix for Tusi et al.[35] was downloaded from GEO accession no. GSE89754. We removed the cells from 'basal_bm1' to avoid dealing with batch effects.

Hematopoiesis dataset (Giladi et al.). The count matrix for Giladi et al.[36] was downloaded from GEO accession no. GSE92575. Only cells without any treatment were used in this analysis. ERCC spikes-ins were removed from the analysis, while the batch effect related to the 'Seq_batch_ID' was regressed out by COMBAT.

Human brain dataset. Human brain single nuclei-seq datasets for the middle temporal gyrus (15,928 nuclei), primary visual cortex (8,998 nuclei), anterior cingulate cortex (7,283 nuclei) and lateral geniculate (1,576 nuclei) were downloaded from the Allen Brain Atlas Data Portal[49]. We filtered out cells of fewer than 200 genes and genes expressed in fewer than 3 cells.

Human liver dataset (MacParland et al.). Human liver data from MacParland et al.[50] was used as a reference dataset for the reference-based cell type annotation methods. Data was downloaded from GEO accession no. GSE115469. The expression matrix was log-transformed without preprocessing.

Human liver dataset (Aizarani et al.). Human liver data from Aizarani et al.[32], including the count matrix and cluster assignment, was downloaded from the GEO accession no. GSE124395 and cell types were annotated according to the clusters in Fig. 1 of the reference paper[32]. Cells expressing fewer than 200 genes and genes expressed in fewer than 20 cells were excluded. Batch information was inferred

from the cell names and the batch effect was regressed out using the 'regress_out' function in SCANPY.

Mouse cortex datasets (Tasic et al.). The mouse cortex datasets from Tasic et al.[33,51] were used to benchmark the reference-based cell type annotation methods. The Tasic et al. 2016 data[51] were used as a reference, while the Tasic et al. 2018 data[33] were used as a benchmark. The count matrix and cell type annotation of the Tasic et al. 2016 data were downloaded from GEO accession no. GSE71585. We excluded cells expressing fewer than 200 genes and genes expressed in fewer than 20 cells. The exon counts matrix and metadata table of the Tasic et al. 2018 data were downloaded from GEO accession no. GSE115746. Cells expressing fewer than 200 genes and genes expressed in fewer than 3 cells were excluded.

All expression data and metadata were imported into the SCANPY[7] Python class and saved as HDF5 files. All data preprocessing steps were saved in Jupiter notebooks and are available at GitHub (https://github.com/SCCAF/sccaf_example).

The SCANPY analysis workflow. All datasets were visualized using a common analysis process based on a standard SCANPY workflow, which includes 7 steps of processing: (1) normalize the data to 10,000 counts per cell; (2) identify highly variable genes based on the mean expression and normalized dispersion of genes; (3) log-transform the data and scale it to unit variance and zero mean; (4) undertake dimension reduction by principal component analysis (PCA); (5) measure the nearest neighbor graph based on the top 15 principal components; (6) undertake dimension reduction by *t*-SNE and UMAP; and (7) Louvain clustering based on the nearest neighbor graph.

Reference-based cell type annotation methods. To benchmark SCCAF, we compared it with reference-based cell type annotation methods, including logistic regression, SingleR (version 1.0.4)[21], SingleCellNet (version 0.1.0)[22], Moana (version 0.1.1)[23], Automated Cell Type Identification using Neural Networks (ACTINN)[24], scClassify (version 0.2.0)[25] and CHaracterization of cEll Types Aided by Hierarchical classification (CHETAH, version 1.2.0)[17]. All methods read the data from the SCANPY HDF5 files generated using the preprocessed notebooks. All scripts used to run these programs are available at GitHub (https://github.com/SCCAF/sccaf_example). SCCAF is available as an open source software package at GitHub (https://github.com/SCCAF/sccaf) and as a Python package index; it has also been implemented as a Galaxy tool.

Reporting Summary. Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Data availability

The datasets together with the accession codes are as follows: pancreas[12], accession no. GSE84133; cortex[30], accession no. GSE60361; retinal bipolar neurons[11], accession no. GSE81904; pancreatic islets[13], accession no. E-MTAB-5061; visual cortex[31], accession no. GSE102827; hematopoiesis[35], GSE89754; hematopoiesis[36], accession no. GSE92575; cortex[51], accession no. GSE71585; cortex[33], accession no. GSE115746; liver[32], accession no. GSE124395; liver[50], accession no. GSE115469. Source data for Figs. 1–5 are included with this paper.

Code availability

An open source implementation of SCCAF is available at GitHub (https://github.com/SCCAF/sccaf) and (https://doi.org/10.5281/zenodo.3695975) under the MIT license. The release includes tutorials and example vignettes for reproducing the analyses presented in this article, as well as all preprocessed datasets considered in this study. The software version used to generate the results presented in this article is also available as Supplementary Software. SCCAF is also accessible from the Python package index (https://pypi.org/project/SCCAF/) and it is implemented as a Galaxy tool in the Human Cell Atlas (https://humancellatlas.usegalaxy.eu/). The SCCAF Galaxy modules are available to install with a few clicks on any Galaxy instance through the main Galaxy Tool Shed at https://toolshed.g2.bx.psu.edu/view/ebi-gxa/suite_sccaf/.

References

41. Pedregosa, F. et al. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
42. Stehman, S. V. Selecting and interpreting measures of thematic classification accuracy. *Remote Sens. Environ.* **62**, 77–89 (1997).
43. Hunter, J. D. Matplotlib: a 2D graphics environment. *Comput. Sci. Eng.* **9**, 90–95 (2007).
44. Hill, C. *Learning Scientific Programming with Python* 333–401 (Cambridge Univ. Press, 2016).
45. Svensson, V., Teichmann, S. A. & Stegle, O. SpatialDE: identification of spatially variable genes. *Nat. Methods* **15**, 343–346 (2018).
46. Leek, J. T. svaseq: removing batch effects and other unwanted noise from sequencing data. *Nucleic Acids Res.* **42**, e161 (2014).
47. Azizi, E., Prabhakaran, S., Carr, A. & Pe'er, D. Bayesian inference for single-cell clustering and imputing. *Genom. Comput. Biol.* **3**, e46 (2017).
48. Athar, A. et al. ArrayExpress update—from bulk to single-cell expression data. *Nucleic Acids Res.* **47**, D711–D715 (2019).
49. Allen Brain Atlas Data Portal. Cell types: overview of the data (Allen Institute, 2015); http://celltypes.brain-map.org
50. MacParland, S. A. et al. Single cell RNA sequencing of human liver reveals distinct intrahepatic macrophage populations. *Nat. Commun.* **9**, 4383 (2018).
51. Tasic, B. et al. Adult mouse cortical cell taxonomy revealed by single cell transcriptomics. *Nat. Neurosci.* **19**, 335–346 (2016).

Acknowledgements

We thank all members of the Teichmann and Brazma labs for helpful discussions. We thank S. Aldridge for proofreading the text. Z.M. is supported by the Single Cell Gene Expression Atlas grant from the Wellcome Trust (no. 108437/Z/15/Z).

Author contributions

Z.M. conceived the method, implemented the algorithm and website, conducted the analyses, created the figures and contributed to the manuscript. P.M., N.H. and I.P. packed the algorithm and implemented it as a Galaxy tool. A.B. and S.A.T. supervised the work and contributed to the manuscript.

Competing interests

In the last three years S.A.T. has consulted for Biogen, Genentech and Roche, and is a member of the Scientific Advisory Board of Foresite Labs and of the Functional Genomics & AI Scientific Advisory Board of GlaxoSmithKline.

Additional information

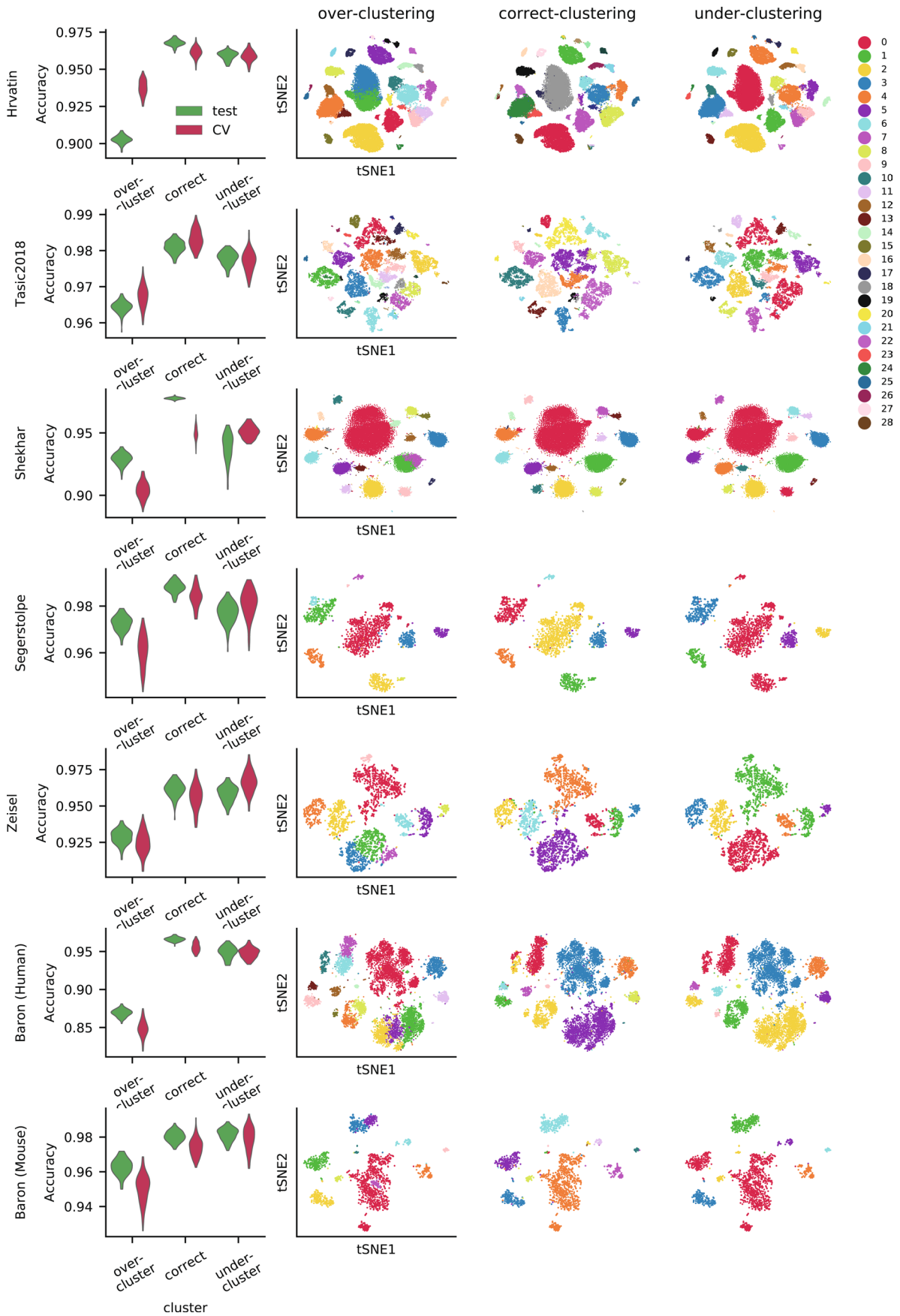
Extended data is available for this paper at https://doi.org/10.1038/s41592-020-0825-9.

Supplementary information is available for this paper at https://doi.org/10.1038/s41592-020-0825-9.

Correspondence and requests for materials should be addressed to A.B. or S.A.T.

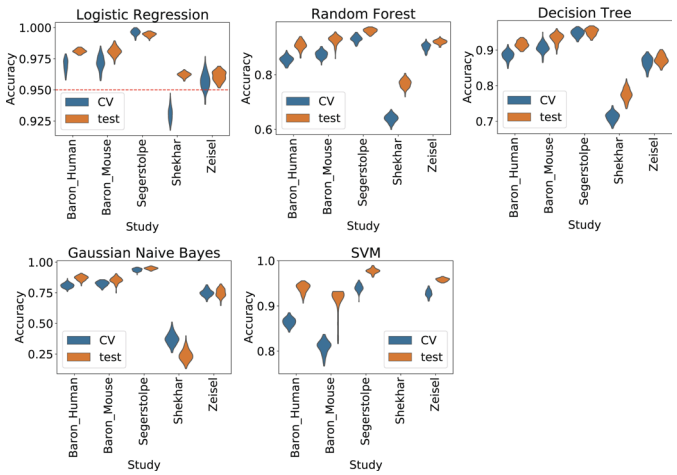
Peer review information Nicole Rusk and Lin Tang were the primary editors on this article and managed its editorial process and peer review in collaboration with the rest of the editorial team.

Reprints and permissions information is available at www.nature.com/reprints.

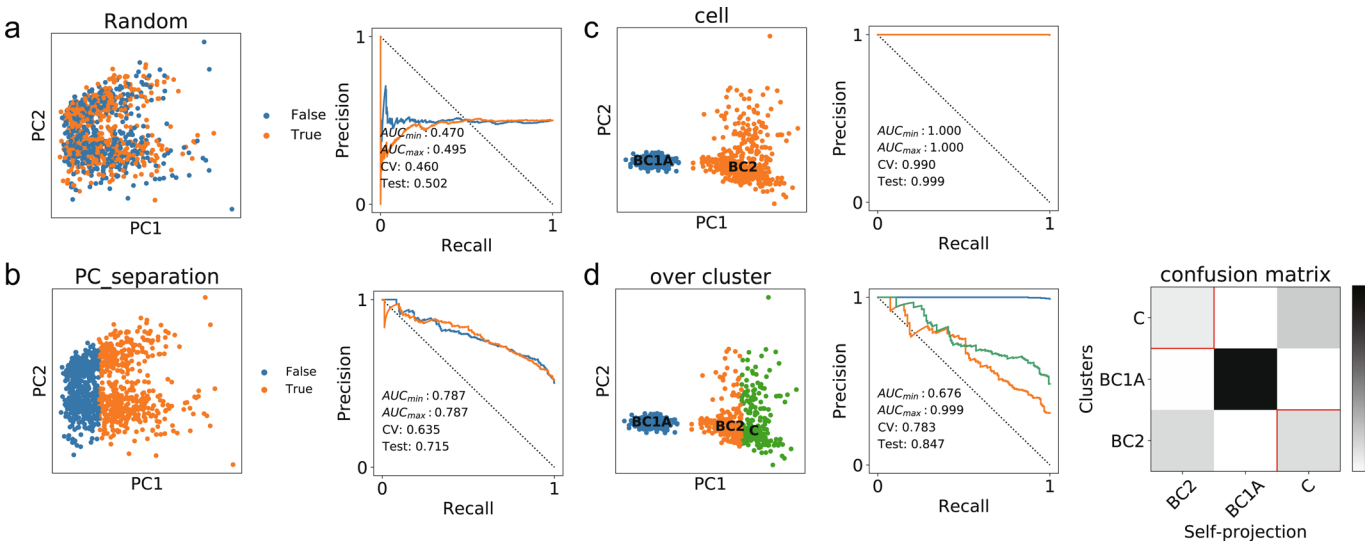


Extended Data Fig. 1 | See next page for caption.

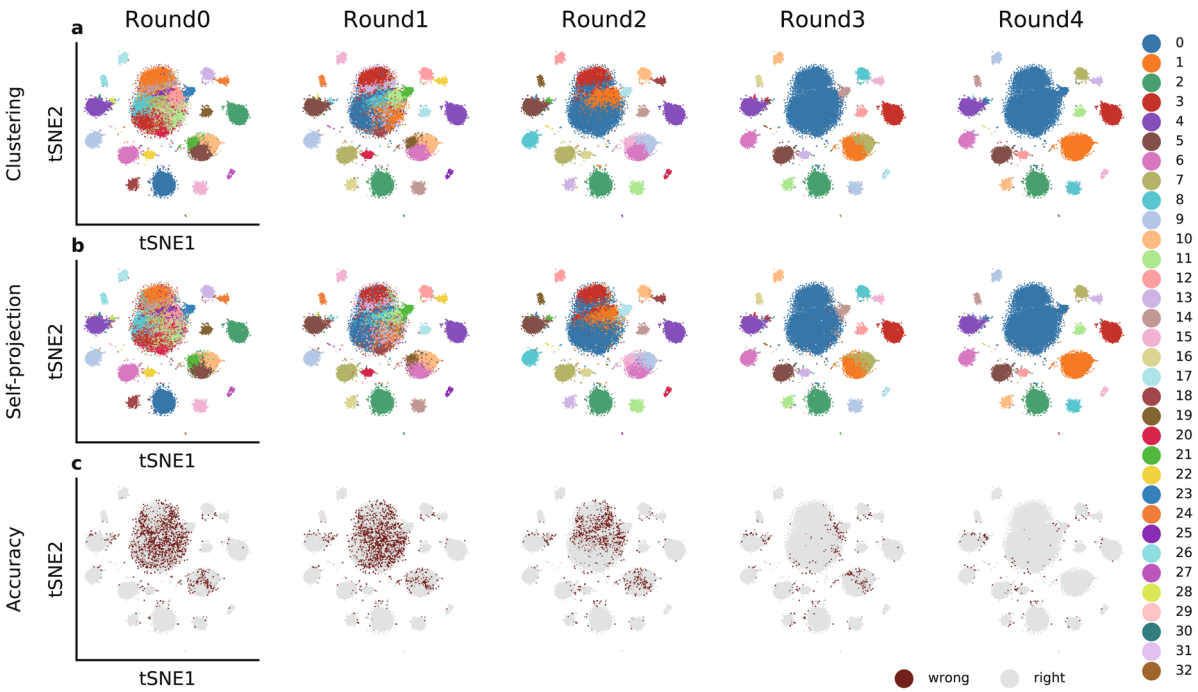
Extended Data Fig. 1 | Self-projection accuracy comparison between the ground truth annotation and the clustering with under-clustering or over-clustering. This test measures the self-projection accuracy on three conditions: 1) the "ground truth" clustering as annotated by human experts (marked as 'correct-clustering'); 2) over-clustering and 3) under-clustering. The violin plots on the left column show the self-projection accuracy distributions (of both cross-validation as red and on the test set as green) for these three conditions in all the datasets by repeating the random sampling 100 times. These plots demonstrate that the "ground truth" clustering corresponds to the highest self-projection accuracy in almost all cases. According to the test results on the datasets: Hrvatin(48,266 cells), Tasic2018 (21,874 cells), Shekhar (26,830 cells), Segerstolpe (2,108 cells), Zeisel (3,005 cells), Baron (Mouse, 1,886 cells) and Baron (Human, 8,199 cells), it is possible to identify the best clustering using self-projection as the clustering consistency test. As for any classifier, it is always easier to perform well on fewer clusters. Thus if two clusterings show a similar level of self-projection accuracy, *for example*, Baron (Mouse), the clustering with more clusters should be chosen for consideration.



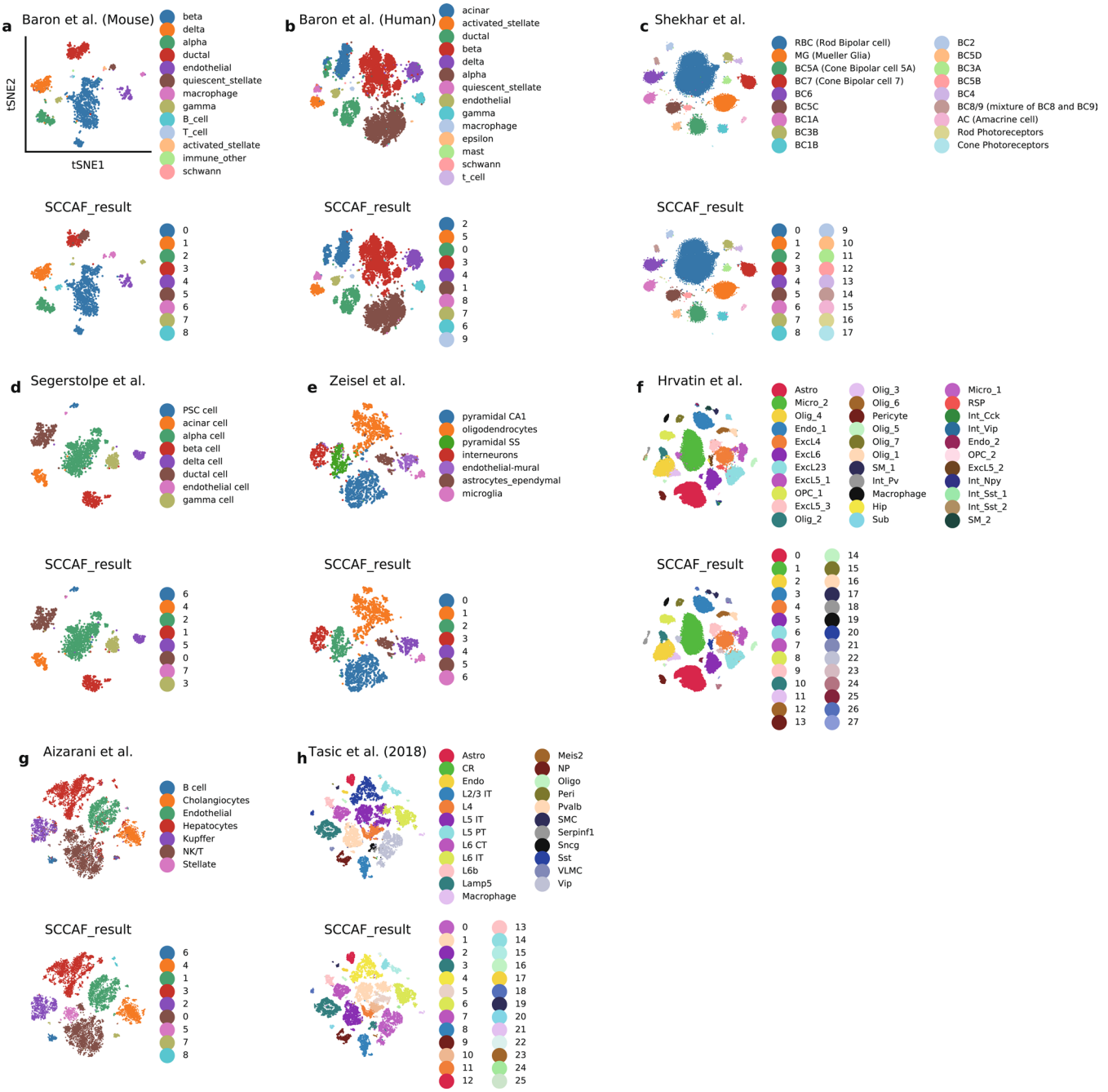
Extended Data Fig. 2 | Testing machine learning methods on ground truth datasets. Five machine learning models were tested on the five ground truth datasets (Baron mouse cells (1,886 cells), Baron human cells (8,199 cells), Shekhar (26,830 cells), Segerstolpe (2,108 cells), Zeisel (3,005 cells)). The data were randomly split into a training set and a test set for self-projection, and this process was repeated 100 times. The distributions of the self-projection accuracies and the mean accuracy of cross-validation in the training were plotted as violin plots.



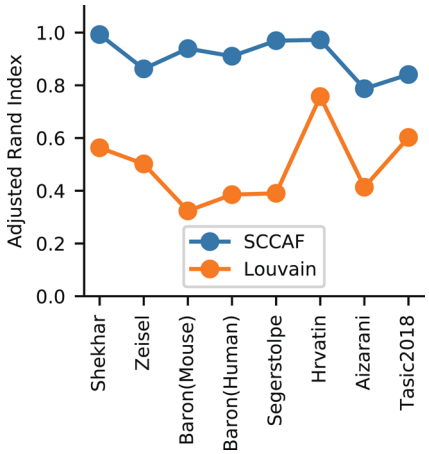
Extended Data Fig. 3 | Self-projection can assess over-clustering on real data. The performance on 1000 BC1A cells from the mouse retina dataset (Shekhar et al.). When the data randomly assigned as two clusters, logistic regression cannot demonstrate any predictive ability in self-projection. When splitting the 1000 BC1A cells into two clusters based on the first principal component (PC), logistic regression shows certain but not ideal predictive ability in self-projection. The performance on 500 BC2 cells and 500 BC1A cells. Self-projection shows high predictive ability. When the 500 BC2 cells are over-clustered into two clusters based on PC1, the confusion always happens between the over-clustered clusters but hardly between BC1A cells and BC2 cells.



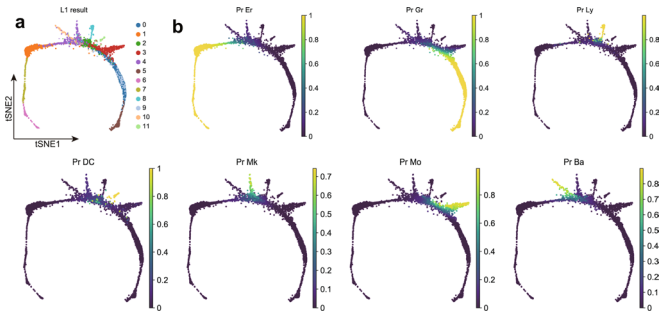
Extended Data Fig. 4 | SCCAF clustering optimization on mouse retina data. The mouse retina data of Shekhar includes 26,830 cells. **a**, show the t-SNE plot of the initial clustering (Round0) and the clustering during the four Rounds (Round1 to Round4) of SCCAF optimization, (**b**) shows the self-projection results, while (**c**) shows the consistency (self-projection accuracy) between the clustering assignment and the self-projection results.



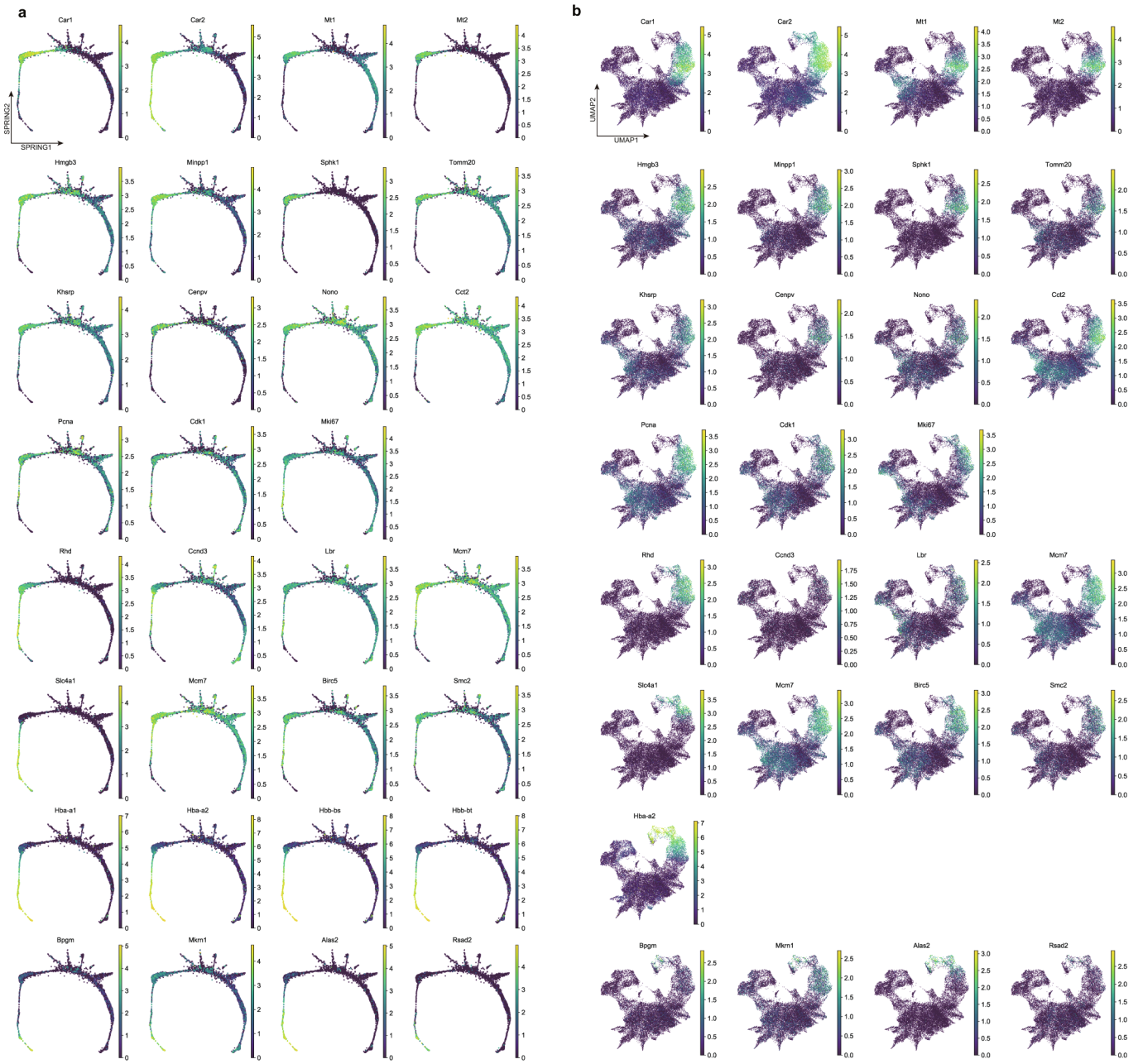
Extended Data Fig. 5 | The self-projection-based clustering optimization achieves clustering identical to human expert annotation. The six expert-annotated datasets **a** Baron mouse cells (1,886 cells), **b**: Baron human cells (8,199 cells), **c**: Shekhar (26,830 cells), **d**, Segerstolpe (2,108 cells), **e**: Zeisel (3,005 cells), **f**: Hrvatin (48,266 cells), **g**: Aizarani (10,305 cells), **h**: Tasic2018 (21,874 cells) are used to compare the SCCAF clustering result and the human expert annotation.



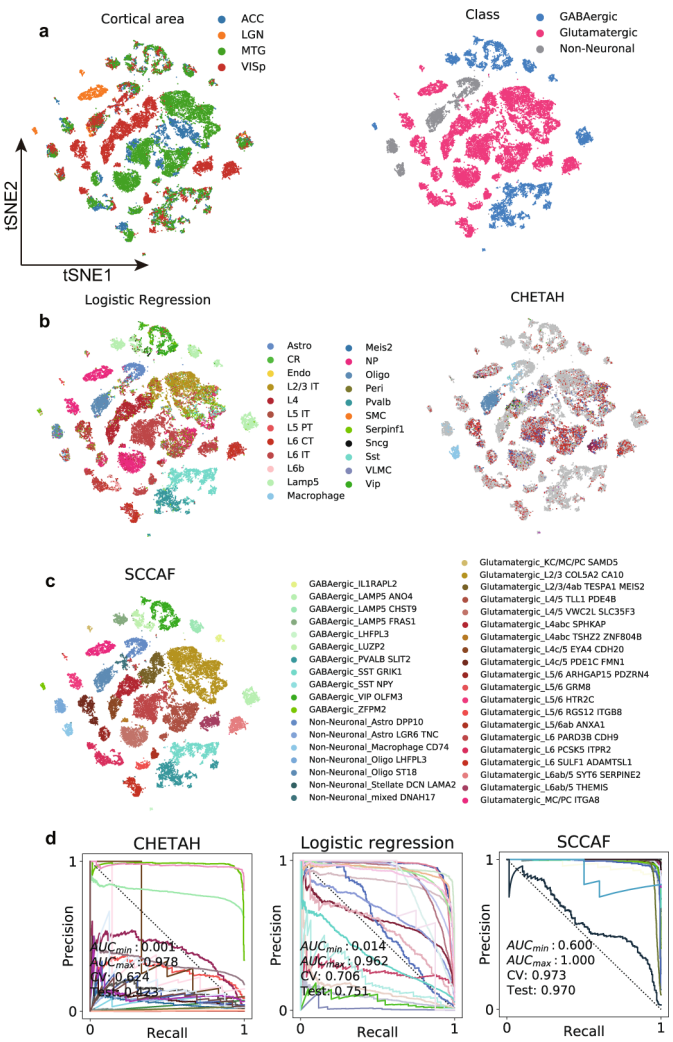
Extended Data Fig. 6 | Adjusted Rand Index evaluation of the SCCAF results compared with Louvain clustering. The Adjusted Rand Index (ARI) is calculated between the clustering results and the human expert annotation. The blue dots show the ARI of SCCAF, while the orange dots show the ARI of the initial Louvain clustering before SCCAF optimization (the initial clustering).



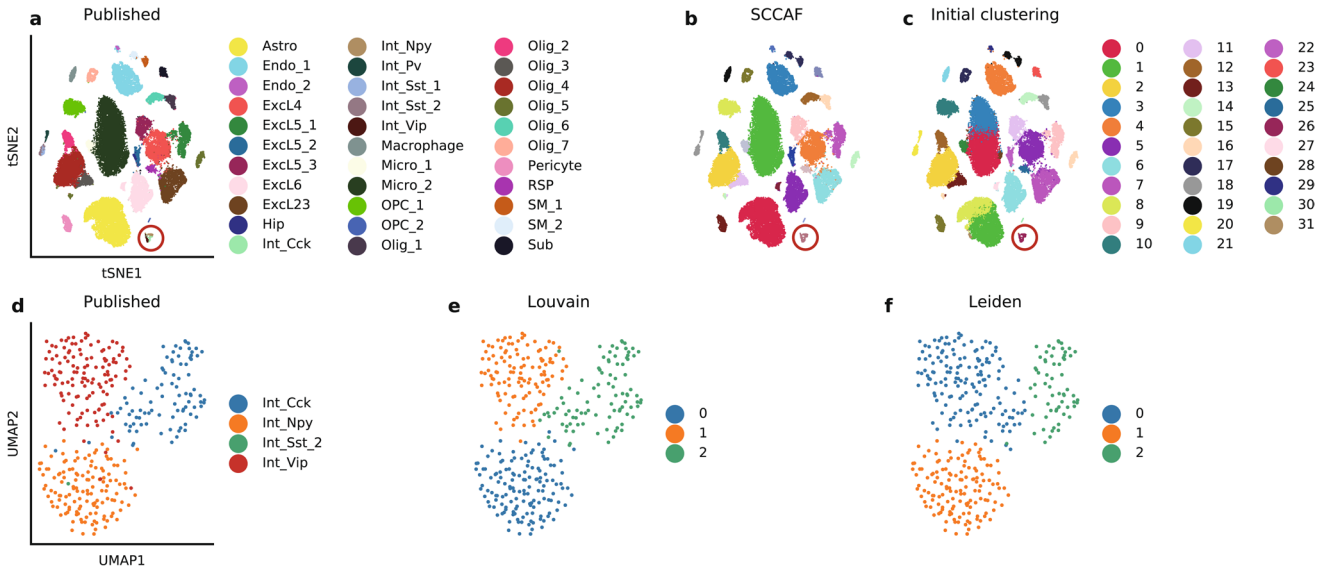
Extended Data Fig. 7 | SCCAF clustering compared with published annotation. a, The clustering shows the result from the SCCAF clustering of the mouse hematopoiesis data (4,016 cells) from Tusi *et al*. **b**, The cell potential (of the 4,016 cells) to develop into different cell lineages (Er: Erythrocytes, Gr: Granulocytes, Ly: Lymphocytes, Mk: Megakaryocytes, Mo: Monocytes, Ba: Basophilic or mast cell) are colored as Viridis.



Extended Data Fig. 8 | Upregulated and downregulated genes in erythrocytes development. The upregulated and downregulated genes in the erythrocytes' development are colored on the SPRING plot and the UMAP plot of the Tusi dataset (4,016 cells) (**a**) and the Giladi dataset (20,202 cells) (**b**).



Extended Data Fig. 9 | SCCAF helps in annotating a new unannotated dataset of the human brain. Human brain single nuclei-Seq data (http://celltypes.brain-map.org/rnaseq) from Middle Temporal Gyrus, Primary Visual Cortex, Anterior Cingulate Cortex and Lateral Geniculate (33,782 cells in total) were analyzed together. In the t-SNE plots, cells are colored according to the **a**) cortical area and the cell classes. Projection-based annotation approaches (logistic regression and CHETAH) were used to annotate the dataset using the mouse brain data from Tasic *et al.* were applied considering the ortholog genes between human and mouse. And the results are colored in the t-SNE plots in **b**. SCCAF was also used to identify the discriminative cell clusters and resulted in 38 clusters. **c**, Each cell cluster is annotated according to the top-ranked feature genes extracted from the SCCAF model. **d**, Self-projection accuracy and ROC curves are compared for these three annotation approaches.



Extended Data Fig. 10 | A hierarchical approach to cluster mouse visual cortex data. a, The t-SNE plot shows the cell types in the Hrvatin dataset (48,266 cells). Three cell types, under the main cell type "Interneurons" (350 cells), cluster together in the red circle. The variances of these three cell types are not dominant when considering the whole dataset. **b**, The first round SCCAF clustering of the Hrvatin dataset (48,266 cells) cannot find such subpopulations, because (**c**) they are clustered together from the initial state. When looking at these three clusters (350 cells) (**d**), Louvain clustering (**e**) may achieve a similar clustering result as the manual annotation. Leiden clustering (**f**) can also identify the three cell types but shows a difference in the center cells.

nature research

Corresponding author(s): Alvis Brazma, Sarah Teichmann

Last updated by author(s): Mar 7, 2020

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see Authors & Referees and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed
☐	☒ The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒ A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☒	☐ The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐ A description of all covariates tested
☒	☐ A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☒	☐ A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☒	☐ For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐ For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐ For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐ Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	All the codes for data preprocessing are available as open source at GitHub (https://github.com/SCCAF/sccaf).
Data analysis	All the codes for data analysis are available as open source at GitHub (https://github.com/SCCAF/sccaf_example).

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors/reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

The single cell data that support the findings of this study are available in GEO with the identifier(s) GSE84133,GSE60361,GSE81904,GSE102827,GSE89754,GSE92575,GSE71585,GSE115746,GSE124395,GSE115469 and in ArrayExpress with the identifier E-MTAB-5061.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	The Baron datasets include 8077 human cells and 1886 mouse cells. The Giladi dataset includes 20202 cells. The Hrvatin dataset includes 48266 cells. The Segerstolpe dataset includes 2108 cells. The Tsui dataset includes 4016 cells. The Shekhar dataset includes 26830 cells. The Zeisel dataset includes 3005 cells. The immune cell atlas dataset includes 762000 cells. The Allen Brain dataset includes 33782 nuclei. The Aizarani dataset includes 10305 cells. The Tasic2018 dataset includes 21874 cells.
Data exclusions	cells annotated as 'doublets' or 'unknown' are excluded from consideration. The exclusion criteria was pre-established according to the publication of the datasets.
Replication	It is the same as the original publication of the datasets.
Randomization	It is the same as the original publication of the datasets.
Blinding	It is the same as the original publication of the datasets.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology
☒	☐ Animals and other organisms
☒	☐ Human research participants
☒	☐ Clinical data

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging