Comparison between COTAN and state of the art libraries for the analysis of scRNA-seq data

University of Pisa
Computational Health Laboratory
Project report

Andrea Alberti
a.alberti14@studenti.unipi.it
Roll number: 588945

Simone Ianniciello
s.ianniciello@studenti.unipi.it
Roll number: 581201

Paul M. Magos
p.magos@studenti.unipi.it
Roll number: 588669

Irene Testa
i.testa@studenti.unipi.it
Roll number: 582061

Matteo Tolloso
m.tolloso@studenti.unipi.it
Roll number: 598067

May 30, 2023

1 Introduction

In recent years, *single-cell RNA sequencing* (scRNA-seq) has significantly advanced our knowledge of cellular biology, leading to a deeper understanding of a wide range of biological processes [25, 12, 4] and diseases [23, 8, 24]. Many tools have been proposed for the analysis of scRNA-seq data and there is a growing interest in exploring alternative methods. One such method is COTAN [6], a statistical and computational method, based on the analysis of the co-expression of gene pairs. In this study, our objective is to compare the performance of COTAN with four other established tools (Seurat [9], Scanpy [22], Monocle [19], and scvi-tools [7]) on 5 different scRNA-seq datasets. Our specific focus lies on two critical phases of the scRNA-seq analysis pipeline: *clustering* and *differential expression analysis*. By evaluating the results of these two steps obtained from different tools, our aim is to provide valuable insights into the strengths and limitations of each method.

The report is organized as follows. Section 2 provides an overview of the datasets, the tools and the methods employed in this study, describing the data pre-processing steps and outlining the approach used to compare the results of clustering and differential expression analysis. Section 3 presents the comparison results obtained. Lastly, Section 4 concludes the report by providing remarks on future endeavors.

2 Methods

2.1 Datasets

The comparison of the tools was conducted over five publicly available datasets, whose attributes are summarized in Table 1.

Table 1: Datasets details

Name	**# Genes**	**# Cells**	**Description**	**Species**	**Ref.**
Tabula Muris Heart	23 433	654	Heart and aorta cells	Mus Musculus	[20]
Tabula Muris Marrow[1]	23 433	2 131	Bone Marrow cells	Mus Musculus	[20]
5kPBMC	33 570	5 527	Peripheral blood mononuclear cells stained with a panel of TotalSeq-B antibodies	Homo Sapiens	[3]
Zhengmix4eq	15 568	3 994	Mixture of purified peripheral blood mononuclear cells	Homo Sapiens	[5]
Zhengmix8eq	15 717	3 994	Mixture of purified peripheral blood mononuclear cells	Homo Sapiens	[5]

[1] Mouse ID 3-F-56

All the datasets were generated with the microfluidic droplet-based 3'-end protocol. Every dataset contains a matrix of UMI counts. The *5kPBMC* dataset consists of Peripheral Blood Mononuclear Cells that where stained with TotalSeq-b antibodies [3] and, in addition to the gene count matrix per cell, includes counts for the surface proteins present on each cell.

2.2 Tools

Our paper emphasizes the comparison between COTAN and four state of the art libraries: Seurat, Scanpy, Monocle, and scvi-tools. The clustering algorithms and the differential expression analysis procedure we used with each of the tools are summarized in Table 2.

Table 2: Tools distinctive features

Tool (version)	**Clustering**	**Differential Expression Analysis**	**Ref.**
COTAN (v2)	Graph-based on PCA (Louvain) refined using Global Differentiation Index	Gene-pair analysis (ranking genes by p-value)	[6]
Seurat (v4)	Graph-based on PCA (Louvain)	Wilcoxon test (ranking genes by average $\log_2$ fold change)	[9]
Monocle (v3)	Graph-based on PCA (Leiden)	Jenson-Shannon divergence followed by fitting a regression model on top ranked genes (ranking genes by 'marker_score')	[19]
Scanpy (v1)	Graph-based on PCA (Leiden)	Wilcoxon test (ranking genes by the z-score underlying the computation of the p-value)	[22]
scvi-tools (v0.2)	Graph-based on deep latent space (Leiden)	Vanilla method [13] (ranking genes by average $\log_2$ fold change)	[7]

2.3 Pre-processing

2.3.1 Labeling

All datasets, except for the 5kPBMC dataset, contained cell type labels that we used to define the "ground true" partition of cells when evaluating the clustering methods. The 5kPBMC dataset was labeled using the surface protein markers expression levels, found by the TotalSeq-b protocol. These expression levels can be seen as a matrix of size 32×5527, where each value (i, j) represents the counts of protein i found on cell j.

Of the 32 antibodies used, 29 are specific to a protein and bind to it, while the remaining 3 are non specific IgG antibodies used as a control. The first

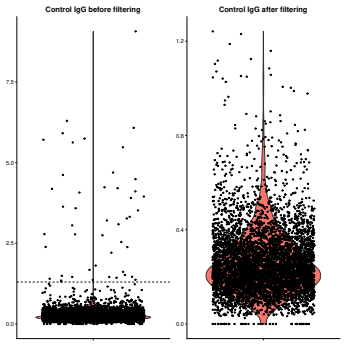

Figure 1: IgG control filtering

step was to filter out cells based on the reads of the control antibodies (see Figure 1) in order to ensure the relevance of the other reads. The following step involves clustering the log-normalized expression levels using the `scran` R library, thus creating a reference labeling for each cell not dependent on the gene counts.

Figure 9 in Supplementary materials shows, for each cluster, the deviation of the expression levels of each protein from the mean of the log-normalized counts. This data was analyzed to manually assign a cell type label to each cluster on a closest match basis, given the informations found on `R&D systems` [2] and the proteins descriptions provided by `GeneCards` [1]. For the majority of clusters we were able to find a pretty good match. The only cluster which we were not able to identify is *cluster 5*, which has only one highly expressed protein: HLA-DR.

This process generated an agnostic labeling allowing unbiased results when comparing the clustering of the tools.

2.3.2 Filtering

The initial step in the analysis pipeline involves filtering the scRNA-seq data. This step was performed on all datasets and includes the removal of doublets (multiple cells mistakenly considered as a single cell) and the exclusion of dead cells from the dataset. In our study, we did not specifically focus on comparing the filtering capabilities of individual tools. Therefore, to make results comparable, the datasets were filtered using Seurat. The filtered datasets were subsequently utilized as input for every tool.

Table 3: Datasets overview after pre-processing

Dataset	**#Genes**	**#Cells**	**#Clusters**
Tabula Muris Heart	13 646	617	5
Tabula Muris Marrow	13 387	1 870	14
5kPBMC	18 306	3 658	11
Zhengmix4eq	12 871	3 959	4
Zhengmix8eq	12 990	3 934	8

Considering the theoretical ability of COTAN to detect correlations of genes even when expression levels are low, we adopted loose filtering thresholds on genes. This decision aimed to retain a broader range of gene expression levels.

The filtering thresholds were chosen based on violin plots:

- on the amount of uniquely expressed genes per cell: outliers that exhibit unusually high gene counts are considered to be doublets.
- on the fraction of mitochondrial counts per cell: outliers with high proportion of mitochondrial gene expression are considered dead.

As an example, Figure 10 in Supplementary materials shows the violin plots of the amount of uniquely expressed genes per cell (named *nFeature_RNA*) and the fraction of

mitochondrial counts per cell (named *percent.mt*) for *5kPBMC* dataset. Figure 11 in Supplementary materials presents the distribution of cells after the filtering phase, where a threshold of 5000 was chosen to filter out doublets and a threshold of 10% was used to exclude dead cells.

Table 3 provides the number of cells and genes retained after the filtering phase on each dataset.

Further pre-processing was performed depending on the tool. For example, normalization was done with every tool but COTAN since it isn't suggested to be used in its pipeline.

2.4 Clustering

For each dataset, we performed a clustering of the cells with each tool. This approach allows a comparison of the tools while also being the starting step of the differential expression analysis, that will be detailed in Section 2.5. In each dataset, the cells were either already labeled or an agnostic labeling was performed as described in Section 2.1, therefore we know how many cell types are present in each dataset. This information was used to make results more comparable by tuning the parameters of the tools in order to get the same amount of clusters. For COTAN, the tuning regards both the first phase (in which uniform clusters are computed) and the merge phase.

2.4.1 Alignment

If a binary classifier that has to predict whether a person is healthy or not answers always with the opposite class, that classification is considered the worst possible. However this is not true for clustering: if two clustering just have swapped labels, they are considered the same.

In order to make results of different tools comparable, as well as to understand how clusters obtained through each tool should be linked with the available ground truth, we 'aligned' the labels of each tool with respect to the labels of the ground truth. This works under the assumption that there is a meaningful overlap between the computed clusters and the known labels: if many cells with a certain ground truth label are present in a computed cluster, then we assume that the label of that cluster should match that ground truth label. This is obviously a best effort approach: if, for example, a true label is related to an equal portion of each computed cluster, it is not trivial to understand how the match should be performed.

The alignment is computed by considering the confusion matrix of true labels and computed cluster labels. We used *complete pivoting* strategies [17] to maximize the sum of the elements in the diagonal of the confusion matrix. From the permutations of rows and columns needed in the pivoting step, we compute the permutation to apply to the computed labels.

2.5 Differential Expression Analysis

The following subsections describe the criteria we used to evaluate the results of the differential expression analysis obtained from different tools.

2.5.1 Validation with a classifier

To evaluate the quality of the markers, we assessed their effectiveness in separating cells into classes looking at the classification performance of a classifier trained on the markers to predict the true classes cells belong to. Specifically, given a tool, for each cell class, we trained a binary Random Forest classifier to distinguish between cells in that class and cells in other classes (one vs rest approach) using as features only the markers of the cluster aligned to the class. We then considered the average classification score attained by the binary classifiers as a comprehensive measure to evaluate the quality of the cluster markers generated by the tool. An important aspect to note is that this measure depends on the quality of the clustering and on the procedure we used to align the cluster to the truth labels. This is because the classifiers are trained and tested to predict the ground truth class of a cell, but the features used to train the classifier are the markers of the cluster identified by the tool and subsequently aligned to the cell classes.

For this analysis, we used the implementation of the Random Forest classifier provided in the Python package scikit-learn [18], setting its parameters to their default values. To validate the classification results, a 5-fold cross-validation approach was employed. The classifier's performance was assessed by computing the *f1 score* on the predictions made by the classifier across every fold. Since there is a meaningful class imbalance on some datasets (see Supplementary materials, Table 5), to obtain a single performance measure we computed a weighted averaged the f1 score achieved by the binary classifiers using as weights the number of cells in each class.

2.5.2 Enrichment analysis

For each cluster, we conducted an enrichment analysis on the top 50 markers from each tool and examined whether the cell classes associated with these genes reflected the true cell labels. This task was accomplished utilizing *Enrichr* [11] with the Tabula Muris [21] and Tabula Sapiens [10] databases (depending on the dataset). By providing a list of genes and selecting a specific database for analysis, Enrichr generates a list of enriched terms of the database along with their respective p-values. For each dataset, we queried Enrichr with the list of markers found by each given tool and also with the list of markers in the intersection of every tool.

3 Results

3.1 Clustering

The clustering obtained with each tool was compared both in qualitative and quantitative terms. The qualitative analysis was done using *scatter* plots and *Sankey* plots. The latter shows how cluster labels were related to the ground truth, thus implicitly testing the validity of the alignment of the labels. From Sankey plots one can evince that the alignment worked well where the overlaps of true and computed labels is significant, which is the case of Tabula Muris Heart (as shown in Figure 12) and Zhengmix4eq datasets. For the rest of the datasets results are more noisy: for example, from Figure 13 it is possible to notice that for some tools in 5kPBMC dataset there isn't a clear relation between the true and computed clusters. In fact they show that the alignment likely failed for some clusters like *Macrophage M2b Activation*: since no computed cluster has a significant fraction linked to it, the alignment is likely to fail for such cluster. One thing to notice is that the alignment was more successful where clusters have uniform size. In the datasets with highly unbalanced clusters, some of the small ones were included in the same computed cluster. This makes any alignment not meaningful since a computed cluster should be aligned with more than one true cluster (due to it containing a big fraction of the cells of more than one true cluster).

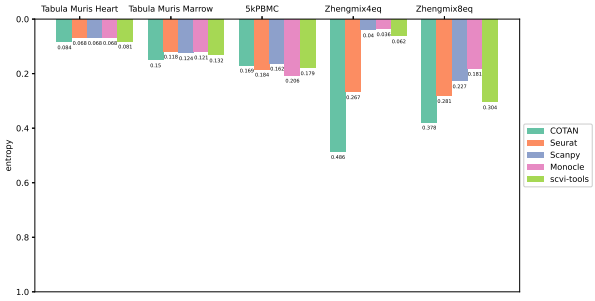

Figure 2: Clustering entropy

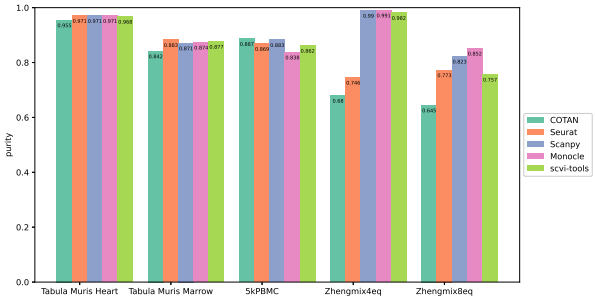

Figure 3: Clustering purity

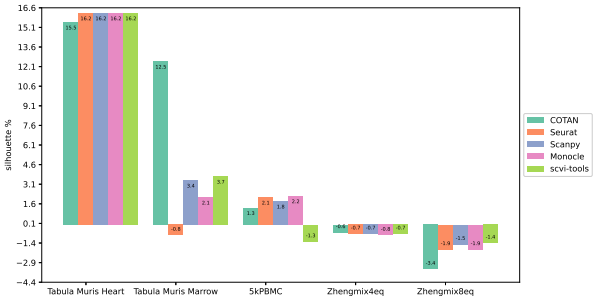

Figure 4: Clustering silhouette

For the quantitative part of clustering analysis, we computed some scores like *Purity*, *Entropy* and *Silhouette*. Both these scores are well known to the literature and they also are independent from the label alignment. From Figure 4, one can notice how, in absolute terms, the silhouette score is quite low. We interpreted this results not as a

problem of the quality of the clustering computed by the tools, but as a side effect of high dimensionality while computing the distance matrix of the cells (further analysis should consider to select less features for the computation of the distance matrix). Regarding purity (Figure 3) and entropy (Figure 2), it is possible to evince that every tool had an high score (meaning high purity value and low entropy value) on absolute terms. The tools are very close to each other in Tabula Muris Heart, Tabula Muris Marrow and 5kPBMC datasets, while in the Zhengmix4eq dataset, COTAN and Seurat had worse results. In Zhengmix8eq dataset, COTAN also had worse scores, with Monocle being the best.

3.2 Differential Expression Analysis

3.2.1 Tools markers overlap

To gain insights into the similarities of the final outputs of the tools' pipeline we examined the overlap among the cluster markers identified by the tools generating *Venn diagrams* (see, as an example, Supplementary materials, Figure 14). We observed that, on average, 20% of the markers (10 markers out of 50) belong to the intersection among all the tools. Examining the intersection sizes between COTAN markers and markers from other tools (Supplementary materials, Table 6), we noticed that the percentages of overlap vary depending on the dataset. Specifically, on the Tabula Muris dataset, the highest percentage of overlap is between Monocle and scivi-tools markers. On the Zheng datasets the highest overlap is observed with Seurat markers, while on the 5kPBMC dataset there isn't a trivial overlapping pattern regarding the tools. Scanpy markers showed the smallest overlap with COTAN markers on all the datasets.

3.2.2 Markers quality assessment

Effectiveness of markers in cell classification Figure 5 shows the weighted f1 score of the Random Forest classifiers trained on the top 50 markers from each tool and on all the genes in the dataset (RF base). On the Tabula Muris Heart and Zhengmix4eq datasets the performances achieved by the classifiers when trained on all the genes are higher then when trained on the markers only, indicating that probably 50 features are not enough to accurately separate cells into classes. The performances of the classifiers trained on the tools markers exhibit remarkable similarity, especially on Tabula Muris Heart, Tabula Muris Marrow and Zhengmix8eq datasets. On the 5kPBMC dataset COTAN markers achieve the highest score. However, on the Zhengmix4eq dataset they achieve the worst score.

To explore the potential impact of the number of top markers used to train the classifiers on classification performance, we repeated the same process varying the number of markers from 1 to 50. The results of this analysis on the 5kPBMC and Zhengmix8eq datasets are shown in Figure 7, the complete results can be found in Supplementary

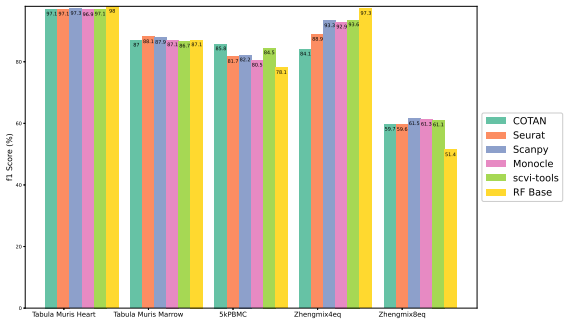

Figure 5: Weighted f1 score of the Random Forest classifiers

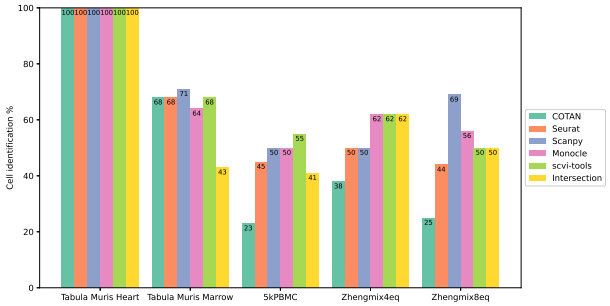

Figure 6: Cell identification scores achieved by the tools

materials. On the 5kPBMC dataset COTAN markers are more effective than the other tool markers, even when training the classifier with a number of markers smaller than 50. On the Zhengmix8eq dataset, instead, the classifier trained on COTAN markers achieves the worst classification results regardless of the number of features. We observed that, for all the tools, on the Tabula Muris Heart, Tabula Muris Marrow and on the 5kPBMC dataset, the classification performance reaches a plateau at approximately 20 features, beyond which there is no substantial improvement. On the Zhengmix4eq and Zhengmix8eq datasets, instead, the performance of the classifier trained on COTAN markers improves even beyond 20 features. Finally, we noticed that scivi-tools and COTAN top 5 markers gave significantly worse classifications across all the datasets, with the exception of the Tabula Muris Marrow dataset. This finding suggests that the top 5 markers obtained from scvi-tools and COTAN may be less effective compared to those computed by the other tools.

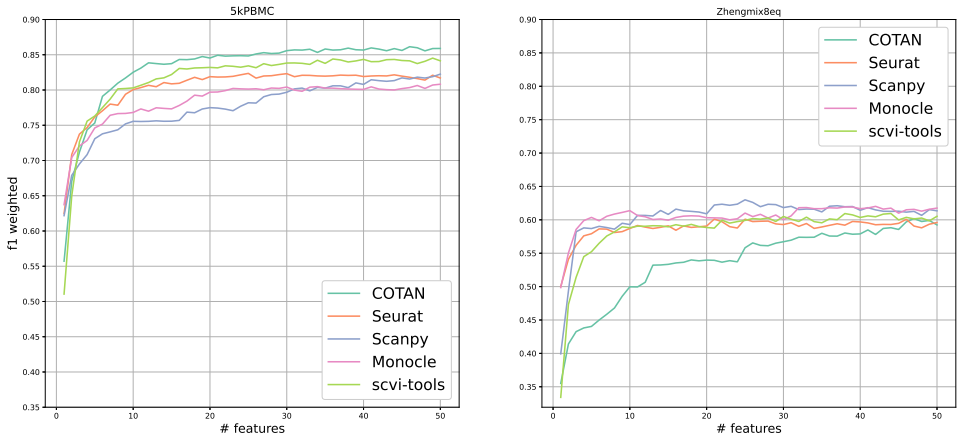

Figure 7: F1 score using top-k markers as features

Cell identification scores We utilized the list of cell types found by Enrichr (ordered by p-value) to manually assign a "cell identification score" to the tools, using the criteria described in the following. For each cluster, we assigned a score of 1 if the true cell label aligned to that cluster was the most probable cell class. If it did not rank as the most probable class but appeared among the top 20 most probable cell classes, we assigned a score of 0.5, while if it did not appear among the top 20 classes, we assigned a score of 0. Finally, we averaged the scores across the clusters. The choice of considering this score was taken because most of the times the top results had a much smaller p-value with respect to the other. Figure 6 shows the cell identification scores achieved by the tools on all the datasets. Interestingly, the scores assigned to the markers in the intersection are relatively lower compared to the scores assigned to the markers identified by individual tools. This could be attributed to the limited number of genes in those lists (10 genes on average), which might not be sufficient to obtain statistically significant results: the resulting ranking is likely not significant. COTAN cell identification scores are slightly worse than the ones of the other tools, even on 5kPBMC dataset, where, conversely, COTAN's markers proved to be the most effective in distinguishing cells into distinct classes (see Figure 7). This contrasting result suggests that the computed cell identification scores may not accurately reflect the quality of the tools markers. Given that the score likely relies on the specific database employed for the enrichment analysis, future studies may explore the utilization of multiple databases to enhance the robustness of the enrichment analysis, allowing for more reliable comparisons between tools.

Table 4: Known markers for Tabula Muris Heart cell classes

Cell class	**Markers**
Cardiac muscle cell	$Nppa^+$, $Myl7^+$, Sln^+
Endocardial cell	$Npr3^+$, $Pecam1^+$
Endothelial cell	$Fabp4^+$, $Cdh5^+$, $Cav1^+$
Fibroblast	$Ddr2^+$, $Tcf21^+$, $Col3a1^+$, $Col1a2^+$, $Col1a1^+$, $Myh11^+$, $Tcf21^+$

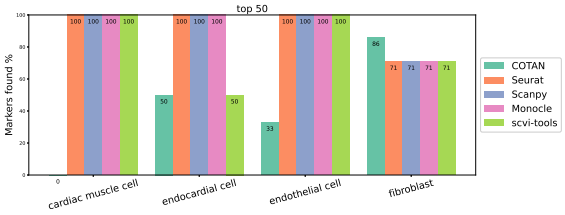

Figure 8: Percentage of known markers for the Tabula Muris Heart cells classes in the top 50 markers identified by the tools

Comparison with known markers For the Tabula Muris Heart dataset, we also assessed whether the markers listed in Table 4, which were known from literature and used by the authors of the study to label cell clusters, appeared among the top 50 markers identified by the tools. The results of this analysis are presented in Figure 8. COTAN exhibits a higher percentage of known markers identified for the *fibroblast* cluster compared to the other tools (it is the only tool retrieving the marker $Ddr2^+$).

However, the percentage of known markers retrieved by COTAN is relatively lower for the cardiac muscle cells, endocardial cells and endothelial cells. Despite the fact that

for these clusters COTAN does not retrieve certain known markers, it is worth noting that the markers it identifies still hold biological significance, as evidenced in Figure 6.

4 Conclusion

This article presents a comprehensive comparison between COTAN and state-of-the-art libraries for the analysis of scRNA-seq data. The study specifically focuses on evaluating the performance of these tools in clustering and differential expression analysis using five distinct datasets: Tabula Muris Heart, Tabula Muris Marrow, Zhengmix4eq, Zhengmix8eq, and 5kPBMC.

Upon careful examination of the results, it was observed that the performance of the analyzed tools exhibited remarkable similarity. However, one notable exception was COTAN, which displayed significantly slower processing times, particularly in scenarios characterized by high gene expression levels, such as the Zhengmix8eq dataset. Despite this drawback, the results obtained from all the tools were comparable, with no particular tool demonstrating superior accuracy over the others.

COTAN showcased its capability in identifying valuable markers for cell classification in specific situations. Notably, it proved effective in accurately identifying markers for classifying cells in the 5kPBMC dataset (Figure 7), as well as for specific tissue types, such as fibroblast cells in Tabula Muris Heart (Figure 8).

In light of these findings, we propose several future research directions. Firstly, we propose evaluating the biological effectiveness of the most significant markers found by the tested tools by comparing these with the top features of a Random Forest classifier. Secondly, it would be interesting to validate the ranking of the markers computed by each tool. Additionally, it would be beneficial to assess the markers identified by the tools using not only Enrichr but also other enrichment analysis tools or databases: this expanded analysis would provide a more comprehensive understanding of the functional relevance of the identified markers.

Code availability

All the scripts used to perform the data analysis presented in this report are available on GitHub at https://github.com/arcotan/single-cell-data-analysis. At the same repository location it is also possible to download the comprehensive analysis results.

References

[1] Genecards - human genes — gene database — gene search. https://www.genecards.org/.

[2] Immune cells: R&d systems. https://www.rndsystems.com/resources/cell-markers/immune-cells.

[3] 10x Genomics. 5k Peripheral Blood Mononuclear Cells (PBMCs) from a healthy donor with a panel of TotalSeq-B Antibodies (Next GEM). https://www.10xgenomics.com/resources/datasets/5-k-peripheral-blood-mononuclear-cells-pbm-cs-from-a-healthy-donor-with-cell-surface-proteins-next-gem-3-1-standard-3-1-0, 2019.

[4] James A Briggs, Caleb Weinreb, Daniel E Wagner, Sean Megason, Leonid Peshkin, Marc W Kirschner, and Allon M Klein. The dynamics of gene expression in vertebrate embryogenesis at single-cell resolution. *Science*, 360(6392):eaar5780, 2018.

[5] Angelo Duò, Mark D Robinson, and Charlotte Soneson. A systematic performance evaluation of clustering methods for single-cell rna-seq data. *F1000Research*, 7:1141, 2020.

[6] Silvia Giulia Galfre, Francesco Morandin, Marco Pietrosanto, Federico Cremisi, and Manuela Helmer-Citterich. Cotan: scrna-seq data analysis based on gene co-expression. *NAR Genomics and Bioinformatics*, 3(3):lqab072, 2021.

[7] Adam Gayoso, Romain Lopez, Galen Xing, Pierre Boyeau, Valeh Valiollah Pour Amiri, Justin Hong, Katherine Wu, Michael Jayasuriya, Edouard Mehlman, Maxime Langevin, Yining Liu, Jules Samaran, Gabriel Misrachi, Achille Nazaret, Oscar Clivio, Chenling Xu, Tal Ashuach, Mariano Gabitto, Mohammad Lotfollahi, Valentine Svensson, Eduardo da Veiga Beltrame, Vitalii Kleshchevnikov, Carlos Talavera-López, Lior Pachter, Fabian J. Theis, Aaron Streets, Michael I. Jordan, Jeffrey Regier, and Nir Yosef. A python library for probabilistic analysis of single-cell omics data. *Nature Biotechnology*, Feb 2022.

[8] Jeffrey M Granja, Sandy Klemm, Lisa M McGinnis, Arwa S Kathiria, Anja Mezger, M Ryan Corces, Benjamin Parks, Eric Gars, Michaela Liedtke, Grace XY Zheng, et al. Single-cell multiomic analysis identifies regulatory programs in mixed-phenotype acute leukemia. *Nature biotechnology*, 37(12):1458–1465, 2019.

[9] Yuhan Hao, Stephanie Hao, Erica Andersen-Nissen, William M Mauck, Shiwei Zheng, Andrew Butler, Maddie J Lee, Aaron J Wilk, Charlotte Darby, Michael Zager, et al. Integrated analysis of multimodal single-cell data. *Cell*, 184(13):3573–3587, 2021.

[10] Robert C Jones, Jim Karkanias, Mark A Krasnow, Angela Oliveira Pisco, Stephen R Quake, Julia Salzman, Nir Yosef, Bryan Bulthaup, Phillip Brown, et al. Tabula sapiens. `https://tabula-sapiens-portal.ds.czbiohub.org`, 2022.

[11] Maxim V Kuleshov, Matthew R Jones, Andrew D Rouillard, Nicolas F Fernandez, Qiaonan Duan, Zichen Wang, Simon Koplev, Sherry L Jenkins, Kathleen M Jagodnik, Alexander Lachmann, et al. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic acids research*, 44(W1):W90–W97, 2016.

[12] Lipin Loo, Jeremy M Simon, Lei Xing, Eric S McCoy, Jesse K Niehaus, Jiami Guo, ES Anton, and Mark J Zylka. Single-cell transcriptomic analysis of mouse neocortical development. *Nature communications*, 10(1):134, 2019.

[13] Romain Lopez, Jeffrey Regier, Michael B Cole, Michael I Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature methods*, 15(12):1053–1058, 2018.

[14] Malte D Luecken and Fabian J Theis. Current best practices in single-cell rna-seq analysis: a tutorial. *Molecular systems biology*, 15(6):e8746, 2019.

[15] Aaron Lun, Karsten Bach, Jong Kyoung Kim, and Antonio Scialdone. scran. `https://doi.org/doi:10.18129/B9.bioc.scran`, 2016.

[16] Davis J McCarthy, Kieran R Campbell, Aaron TL Lun, and Quin F Wills. Scater: pre-processing, quality control, normalization and visualization of single-cell rna-seq data in r. *Bioinformatics*, 33(8):1179–1186, 2017.

[17] Christopher Melgaard and Ming Gu. Gaussian elimination with randomized complete pivoting. *arXiv preprint arXiv:1511.08528*, 2015.

[18] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[19] Xiaojie Qiu, Qi Mao, Ying Tang, Li Wang, Raghav Chawla, Hannah A Pliner, and Cole Trapnell. Reversed graph embedding resolves complex single-cell trajectories. *Nature methods*, 14(10):979–982, 2017.

[20] Nicholas Schaum, Jim Karkanias, Norma F Neff, Andrew P May, Stephen R Quake, Tony Wyss-Coray, Spyros Darmanis, Joshua Batson, Olga Botvinnik, Michelle B Chen, et al. Single-cell transcriptomics of 20 mouse organs creates a tabula muris: The tabula muris consortium. *Nature*, 562(7727):367, 2018.

[21] Nicholas Schaum, Jim Karkanias, Norma F Neff, Andrew P May, Stephen R Quake, Tony Wyss-Coray, Spyros Darmanis, Joshua Batson, Olga Botvinnik, Michelle B Chen, et al. Tabula muris. `https://tabula-muris.ds.czbiohub.org`, 2018.

[22] F Alexander Wolf, Philipp Angerer, and Fabian J Theis. Scanpy: large-scale single-cell gene expression data analysis. *Genome biology*, 19:1–5, 2018.

[23] Sunny Z Wu, Ghamdan Al-Eryani, Daniel Lee Roden, Simon Junankar, Kate Harvey, Alma Andersson, Aatish Thennavan, Chenfei Wang, James R Torpy, Nenad Bartonicek, et al. A single-cell and spatially resolved atlas of human breast cancers. *Nature genetics*, 53(9):1334–1347, 2021.

[24] Chen Yao, Hong-Wei Sun, Neal E Lacey, Yun Ji, E Ashley Moseman, Han-Yu Shih, Elisabeth F Heuston, Martha Kirby, Stacie Anderson, Jun Cheng, et al. Single-cell rna-seq reveals tox as a key regulator of cd8+ t cell persistence in chronic infection. *Nature immunology*, 20(7):890–901, 2019.

[25] Scott A Yuzwa, Michael J Borrett, Brendan T Innes, Anastassia Voronova, Troy Ketela, David R Kaplan, Gary D Bader, and Freda D Miller. Developmental emergence of adult neural stem cells as revealed by single-cell transcriptional profiling. *Cell reports*, 21(13):3970–3986, 2017.

[26] Grace X. Zheng, Jessica M. Terry, Phillip Belgrader, Paul Ryvkin, Zachary W. Bent, Ryan Wilson, Solongo B. Ziraldo, Tobias D. Wheeler, Geoff P. McDermott, Junjie Zhu, and et al. Massively parallel digital transcriptional profiling of single cells. *Nature Communications*, 8(1), 2017.

Supplementary materials

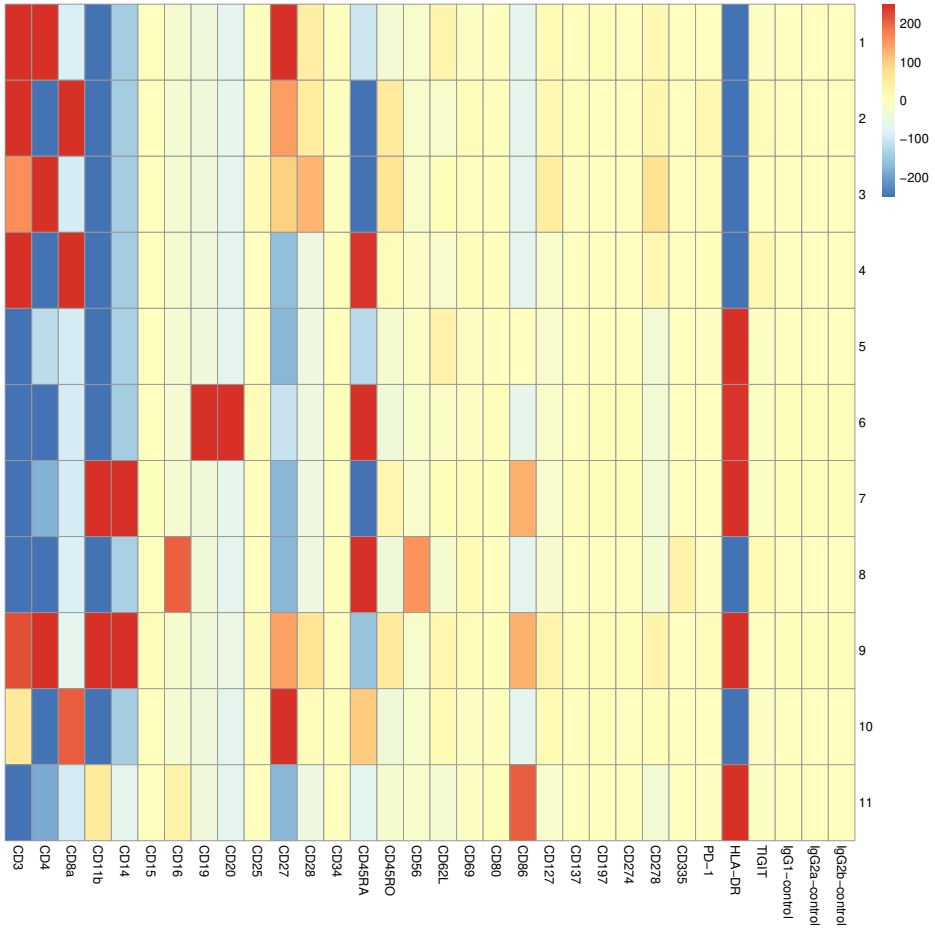

Figure 9: Cell Surface Markers clustering Heatmap

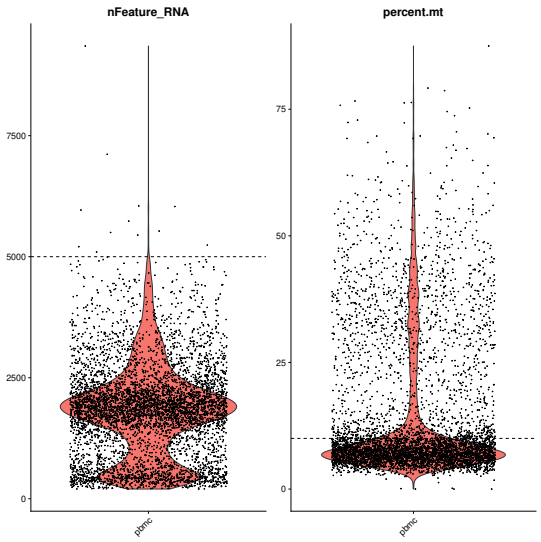

Figure 10: 5kPBMC cells before filtering

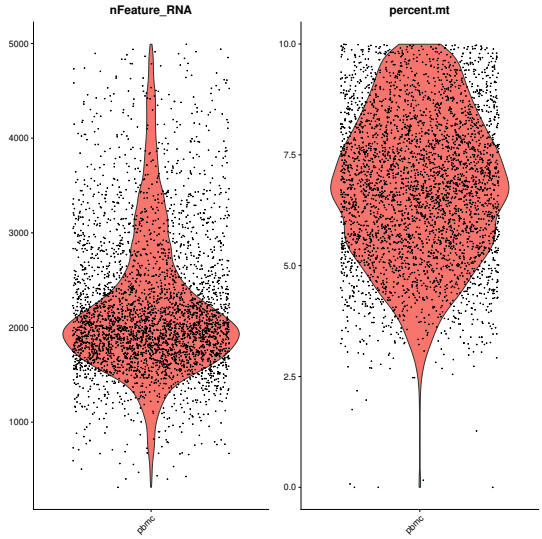

Figure 11: 5kPBMC cells after filtering

Table 5: Cell classes in the datasets along with their corresponding sizes

Dataset	**Cell class**	**Class size (%)**
Tabula Muris Heart	Fibroblast	36
	Endothelial cell	28
	Unkown cluster	16
	Endocardial cell	10
	Cardiac muscle cell	10
Tabula Muris Marrow	Granulocyte	19
	Monocyte	14
	Granulocytopoietic cell	11
	Hematopoietic precursor cell	9
	Proerythroblast	9
	Promonocyte	7
	Macrophage	7
	Late pro-B cell	6
	Erythroblast	5
	T cell	4
	Immature B cell	4
	Basophil	2
	Fraction A pre-pro B cell	2
	Early pro-B cell	1
5kPBMC	Monocyte	31
	$CD28^-$ Helper T Cell	22
	$CD28^+$ Helper T Cell	17
	Follicular B Cell	8
	Natural Killer Cell	7
	Memory Cytotoxic T cell	5
	$HLA\text{-}DR^+$	3
	Naive Cytotoxic T Cell	3
	Effector Cytotoxic T cell	2
	$CD4^+$ Monocyte	1
	Macrophage M2b Activation	1
Zhengmix4eq	CD14 Monocytes	25
	Naive cytotoxic	25
	B cells	25
	Regulatory T cells	25
Zhengmix8eq	CD14 Monocytes	15
	CD56 nk	15
	Memory T cells	13
	Naive T cells	13
	Regulatory T cells	13
	B cells	12
	CD4 T helper	10
	Naive cytotoxic	10

Table 6: Percentage of overlap between the 50 markers found by COTAN and other tools. The highest overleap is highligthed in bold, the lowest in italic.

Dataset	**Class**	**Seurat**	**Scanpy**	**Monocle**	**scivi-tools**
Tabula Muris Heart	Fibroblast	62	62	**78**	*50*
	Endothelial cell	66	*52*	66	**68**
	Unkown cluster	*30*	46	**62**	36
	Endocardial cell	52	*46*	54	**70**
	Cardiac muscle cell	38	*6*	**60**	38
	Avg	49.6	*42.4*	**64**	52.4
Tabula Muris Marrow	Granulocyte	42	*30*	34	**46**
	Monocyte	50	*38*	**72**	60
	Granulocytopoietic cell	*24*	40	42	**48**
	Hematopoietic precursor cell	4	*2*	4	**36**
	Proerythroblast	*28*	*28*	**60**	58
	Promonocyte	6	*4*	8	**26**
	Macrophage	32	*22*	60	**62**
	Late pro-B cell	*22*	24	56	**70**
	Erythroblast	*38*	*38*	44	**64**
	T cell	46	*16*	**74**	68
	Immature B cell	44	*16*	40	**60**
	Basophil	34	*32*	**62**	**62**
	Fraction A pre-pro B cell	***0***	***0***	***0***	***0***
	Early pro-B cell	***0***	***0***	***0***	***0***
	Avg	26.4	*20.7*	39.7	**47.1**
5kPBMC	Monocyte	*48*	52	**64**	54
	$CD28^-$ Helper T Cell	**34**	*6*	40	**34**
	$CD28^+$ Helper T Cell	**44**	*28*	*28*	34
	Follicular B Cell	64	*56*	62	**66**
	Natural Killer Cell	68	*52*	**76**	74
	Memory Cytotoxic T cell	**68**	56	*36*	66
	HLA-DR^+	*14*	20	**44**	38
	Naive Cytotoxic T Cell	*4*	14	16	**42**
	Effector Cytotoxic T cell	4	**50**	4	*0*
	$CD4^+$ Monocyte	***0***	***0***	***0***	***0***
	Macrophage M2b Activation	*38*	40	**52**	42
	Avg	35.1	*34*	38.4	**40.9**
Zhengmix4eq	CD14 Monocytes	**84**	68	72	*48*
	Naive cytotoxic	**34**	*12*	*12*	28
	B cells	**68**	50	*32*	52
	Regulatory T cells	**38**	*2*	*2*	*2*
	Avg	**56**	33	*29.5*	32.5
Zhengmix8eq	CD14 Monocytes	**82**	60	66	*56*
	CD56 nk	**84**	*56*	72	74
	Memory T cells	**44**	24	*10*	34
	Naive T cells	**26**	20	*14*	16
	Regulatory T cells	*4*	*4*	*4*	**6**
	B cells	**82**	48	*44*	68
	CD4 T helper	16	18	**48**	*2*
	Naive cytotoxic	*0*	*0*	*0*	**2**
	Avg	**42.3**	*28.8*	32.3	32.3

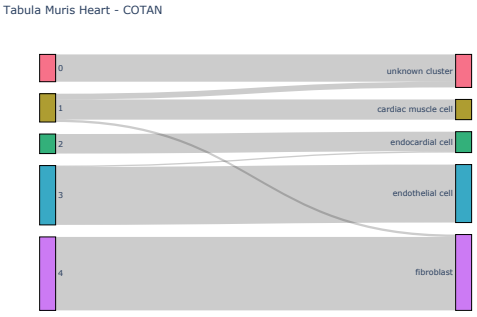

Figure 12: COTAN clusters in Tabula Muris Heart dataset

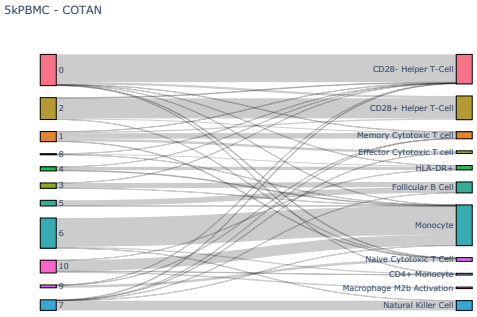

Figure 13: COTAN clusters in 5kPBMC dataset

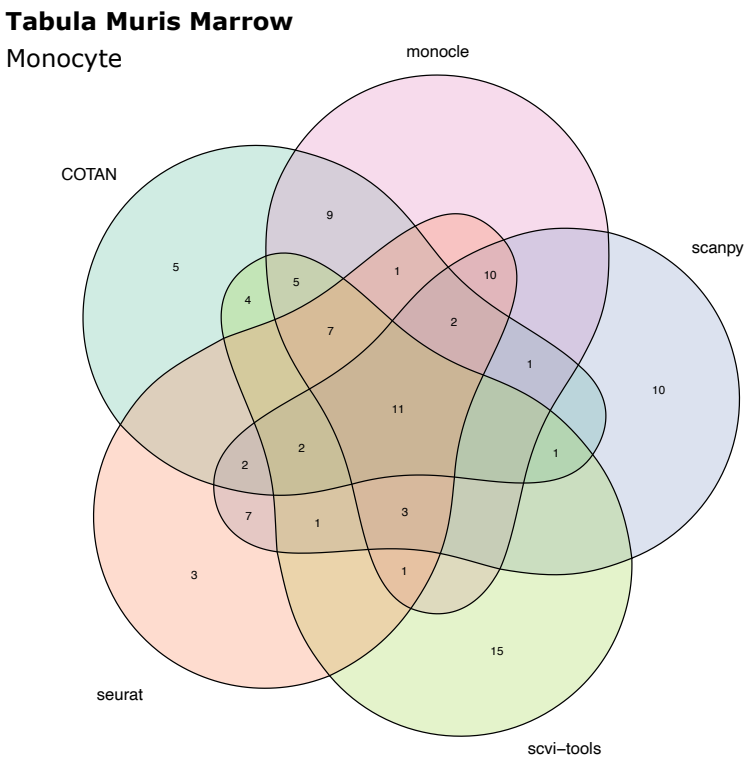

Figure 14: Venn diagram showing the intersections between the top 50 markers identified by different tools for the Monocyte cell class of the Tabula Muris Marrow dataset

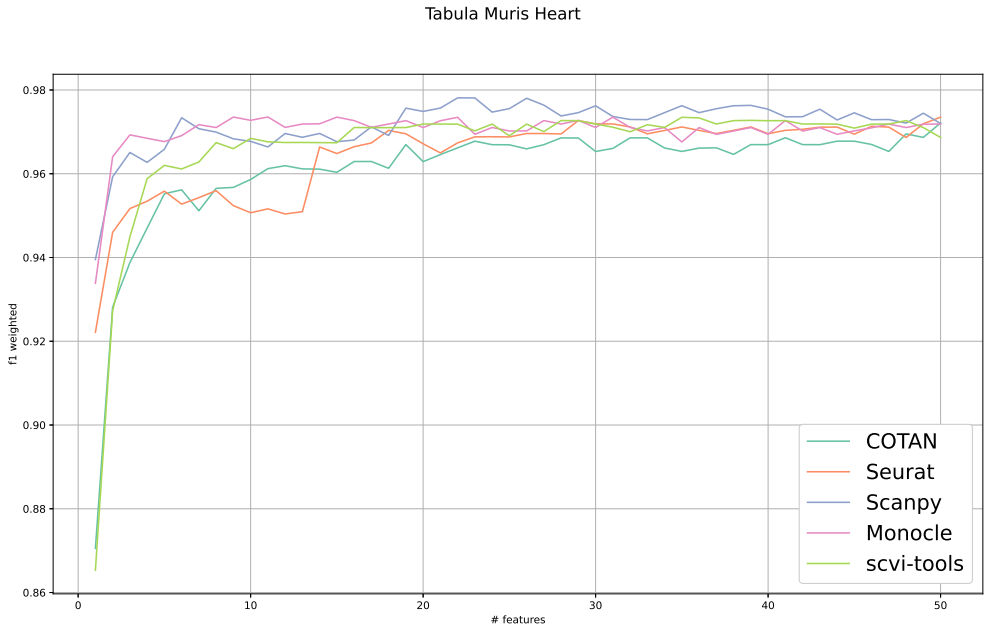

Figure 15: F1 score using top-k markers as features on dataset Tabula Muris Heart

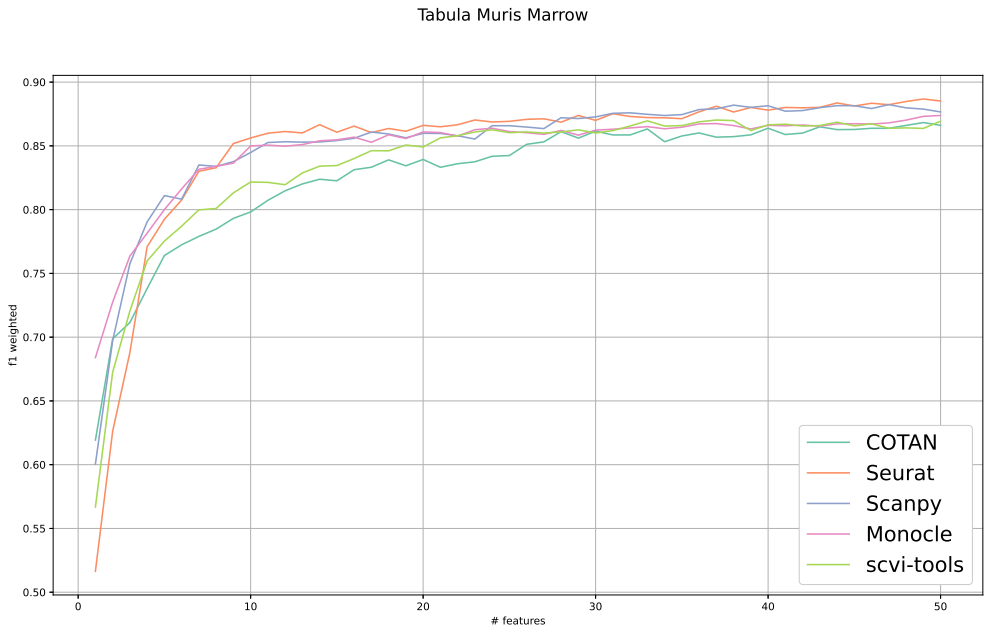

Figure 16: F1 score using top-k markers as features on dataset Tabula Muris Marrow

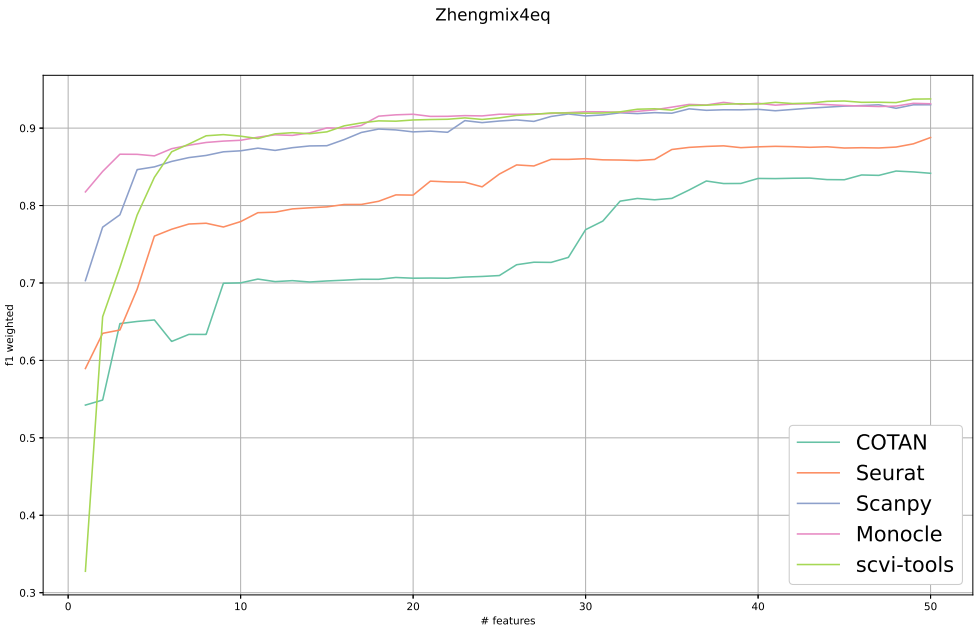

Figure 17: F1 score using top-k markers as features on dataset Zhengmix4eq