

scellop: a scalable redesign of cell population plots for single-cell data

Thomas C. Smits^{1, }, Nikolay Akhmetov^{1, }, Tiffany S. Liaw^{1, }, Mark S. Keller^{1, }, Eric Mörtz^{1, },
Nils Gehlenborg^{1,*, }

¹Department of Biomedical Informatics, Harvard Medical School, Boston, MA 02115, United States

*Corresponding author. Department of Biomedical Informatics, Harvard Medical School, Boston, MA 02115, United States. E-mail: nils@hms.harvard.edu

Associate Editor: Guoqiang Yu

Abstract

Summary: Cell population plots are visualizations showing cell population distributions in biological samples with single-cell data, traditionally shown with stacked bar charts. Here, we address issues with this approach, particularly its limited scalability with increasing number of cell types and samples, and present *scellop*, a novel interactive cell population viewer combining visual encodings optimized for common user tasks in studying populations of cells across samples or conditions.

Availability and implementation: *scellop* is available under the MIT licence at <https://github.com/hms-dbmi/scellop>, and is available on PyPI (<https://pypi.org/project/scellop/>) and NPM (<https://www.npmjs.com/package/scellop>). A demo is available at <https://scellop.netlify.app/>.

1 Introduction

Cell population plots are visualizations showing cell types, states, or clusters in a stratified manner, for instance among samples or conditions. These are used for comparing cell types within and between samples. They support analyzing heterogeneity across cell populations, examining present cell types, and comparing cell counts, often in different populations, such as disease states (Sultana *et al.* 2025), locations (Okendo *et al.* 2022), and subgroups of cell types (Li *et al.* 2024). In publications, these are often accompanied by dimensionality reduction plots (Yu *et al.* 2019). They can also be used for antibody isotypes in different cells (Domínguez Conde *et al.* 2022). Cell populations are traditionally shown using stacked bar charts, with samples as individual bars, cell types as colored segments of various lengths corresponding to the number or proportion of cells.

Cleveland & McGill (1984) highlight a crucial issue with comparing multiple segments in stacked bar charts in their study of human perception, noting participants are better at comparing position than length. Shifted segments are especially hard to compare. Only the bottom cell type in a stacked bar chart has the same starting point and can therefore be compared by the position of the top of its segment. Talbot *et al.* (2014) expand their study to highlight the effect of comparing lengths with separated bars, noting that comparison between separated bars is harder than adjacent bars. Nobre *et al.* (2024) show that users achieve lower accuracy and take more time for various tasks using stacked bar charts compared to other common chart types.

Now, with larger single-cell atlas studies that have the increasing ability to detect more and rarer cell types across samples and conditions, the challenge of visually comparing cell populations by the size of the segments in the stacked bar chart has become more pronounced, as bar segments are smaller and separation between bar segments in different datasets is increased. Additionally, increasing the number of cell types also requires more colors to distinguish categories. Sultana *et al.* (2025) include 30 cell types in their cell population plot. The average number of cell types in annotated RNAseq datasets from the Human BioMolecular Atlas Program (HuBMAP) (Jain *et al.* 2023) is 33 (see Supplementary Materials). However, using seven or more colors to visually encode categories impacts readability (Giovannangeli *et al.* 2021), and identification accuracy decreases with more colors (Tseng *et al.* 2023). Therefore, an alternative encoding is necessary for scalable population plots.

Here, we evaluate the user tasks and needs for cell population plots. We redesign these plots and introduce *scellop*, a flexible viewer for comparison and communication of cell populations.

2 Methods

2.1 Design considerations

To evaluate the issues with the traditional cell population view (stacked bar chart approach) and gain an understanding of desired features, we performed a user study ($N = 14$) using the cell population plot in the HuBMAP Data Portal Tissue Blocks

Received: 31 October 2025. Revised: 15 January 2026. Accepted: 20 February 2026

© The Author(s) 2026. Published by Oxford University Press.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted reuse, distribution, and reproduction in any medium, provided the original work is properly cited.

Comparison as an example (<https://hubmapconsortium.github.io/tissue-bar-graphs/>) (Fig. S1). See [Supplementary Materials](#) for details.

The main desired interactions of a visualization of cell populations were normalization, grouping by cell type hierarchy, overview-to-detail navigation, the ability to filter and group by metadata, and showing additional context for cell types and samples. Additional interactions raised were related to cell type sorting and exporting the visualization. The main issues raised were related to the color scheme. Users could not understand what the different colors represented with many samples, and were unable to change the color scheme to better represent the sample distribution. Other raised issues were regarding the amount and granularity of cell types, making it challenging to get an overview of the distributions. With larger numbers of cell types and samples, identifying absent and universally present cell types proves to be challenging. Samples with different cell type granularities, e.g. with cell type annotation performed by different algorithms, are hard to compare. Other potential issues arise from small fractions, which can be hard to identify and compare. Additionally, when the order of cell types is determined by their overall abundance, this order may not reflect the relative cell type proportions in each sample, confusing the viewer when examining a single sample.

We distinguish different groups of user tasks: (i) viewing the structure of a single sample (e.g. what is the most common cell type, what is the proportion of a given cell type, how do the proportions of multiple cell types in the same sample compare), (ii) comparing the structures of multiple samples (e.g. how do proportions of a given cell type compare in different samples, in how many samples is a particular cell type identified, what percentage of total cells of all samples does a given cell type contribute), and (iii) comparing the structures of multiple samples in relation to their metadata (e.g. what is the most common cell type for a given organ, is there a correlation between proportion of a cell type and sample metadata). In all of these tasks, the ability to show cell type hierarchy and group and filter on these is imperative.

To support these user tasks, we developed an interactive tool called *scellop*. We redesign cell population plots to better support cell type comparisons. We use a central heatmap for general trends, encoding samples and cell types as rows and columns. Several tools exist for heatmap-like views with flexible encodings [Bertier (Perin *et al.* 2014), Clustergrammer (Fernandez *et al.* 2017), Funkyheatmap (Cannoodt* *et al.* 2025)], building on Bertin's matrix principles (Bertin 1977). Although these can show overall patterns and help users to compare trends in multiple samples, they do not support inspection of individual sample structures, various normalization and transformation operations (except Clustergrammer), and operations on cell type hierarchies. Furthermore, they require significant pre-processing to show cell populations of single-cell data. In *scellop*, to allow for within-sample and between-sample comparisons, each heatmap row can be expanded into a bar chart. Bar and violin plot panels aligned to the heatmap display cell counts and distributions. *scellop* supports all desired interactions identified from our design study, including normalization, grouping and filtering. Together, these features comprise the full set of tasks from Schneiderman's task taxonomy for information

visualization: overview—zoom—filter—details-on-demand—relate—history—extract (Shneiderman 1996).

2.2 Implementation

scellop is available as a Python package on PyPI (<https://pypi.org/project/scellop/>) and a JavaScript package on npm (<https://www.npmjs.com/package/scellop>). The Python package provides a Jupyter widget implemented with anywidget (Manz *et al.* 2024). *scellop* is implemented in React, using visx (<https://airbnb.io/visx/>) to incorporate D3-based visualizations (Bostock *et al.* 2011) for various scales and axis rendering. Undo and redo are supported through the Zustand state manager (<https://zustand.docs.pmnd.rs/>) with zundo middleware (<https://github.com/charkour/zundo>). All visualization panels can be resized, and the heatmap allows for zooming in on rows and columns. A configuration panel allows users to select theme and color schemes, set normalizations and transformations, determine side panels (bars, stacked bars, or violin plots), transpose the heatmap, and set different zoom levels. Users can select rows to display as bar charts embedded into the heatmap. Users can sort by counts, alphabetical, or sample and cell type metadata, such as donor age or cell ontology hierarchy. Data can be filtered based on related metadata values. Colors can also be configured individually for samples and cell types for bar charts and side panels. The resulting visualization can be exported as a high-resolution PNG or SVG file. All interactions are also available from the context menu. Information on the performance of *scellop*'s interactions is included in the [Supplementary Materials](#). By transposing the view, removing the heatmap, and using stacked bars in the side panel, the traditional stacked bar charts cell population plot can be created. This configuration can be accessed instantly as a preset from the settings. Data loading from (zarr-indexed) AnnData (Virshup *et al.* 2024) are supported, making this an scverse-compatible visualization tool (Virshup *et al.* 2023). The Python package includes additional data loading functionality, supporting Pandas DataFrames (McKinney 2010) and various ways of supplying metadata.

A demo with kidney RNAseq datasets is available at <https://scellop.netlify.app/>. *scellop* is integrated within the HuBMAP Data Portal (Jain *et al.* 2023) as a default visualization to show organ-level overviews of cell type populations (e.g. <https://portal.hubmapconsortium.org/organs/kidney#cell-population-plot>) and as a Python analysis template in the integrated JupyterLab analysis environment for HuBMAP members (Workspaces).

3 Results

Figure 1 shows the *scellop* viewer with data from the Human Lung Cell Atlas (Sikkema *et al.* 2023) (available from <https://cellxgene.cziscience.com/collections/6f6d381a-7701-4781-935c-db10d30de293>), which constitutes of 484 datasets with over 2.3 million cells of 51 cell types (including “unknown”). *scellop*'s default view shows a heatmap of cell counts with bar charts of total cell and sample counts on the side (Fig. 1A). Sorting and filtering allow for grouping of cell types and datasets. Additional settings allow for log normalization and setting colors of bars, aiding communication of interesting results. It shows that donors with

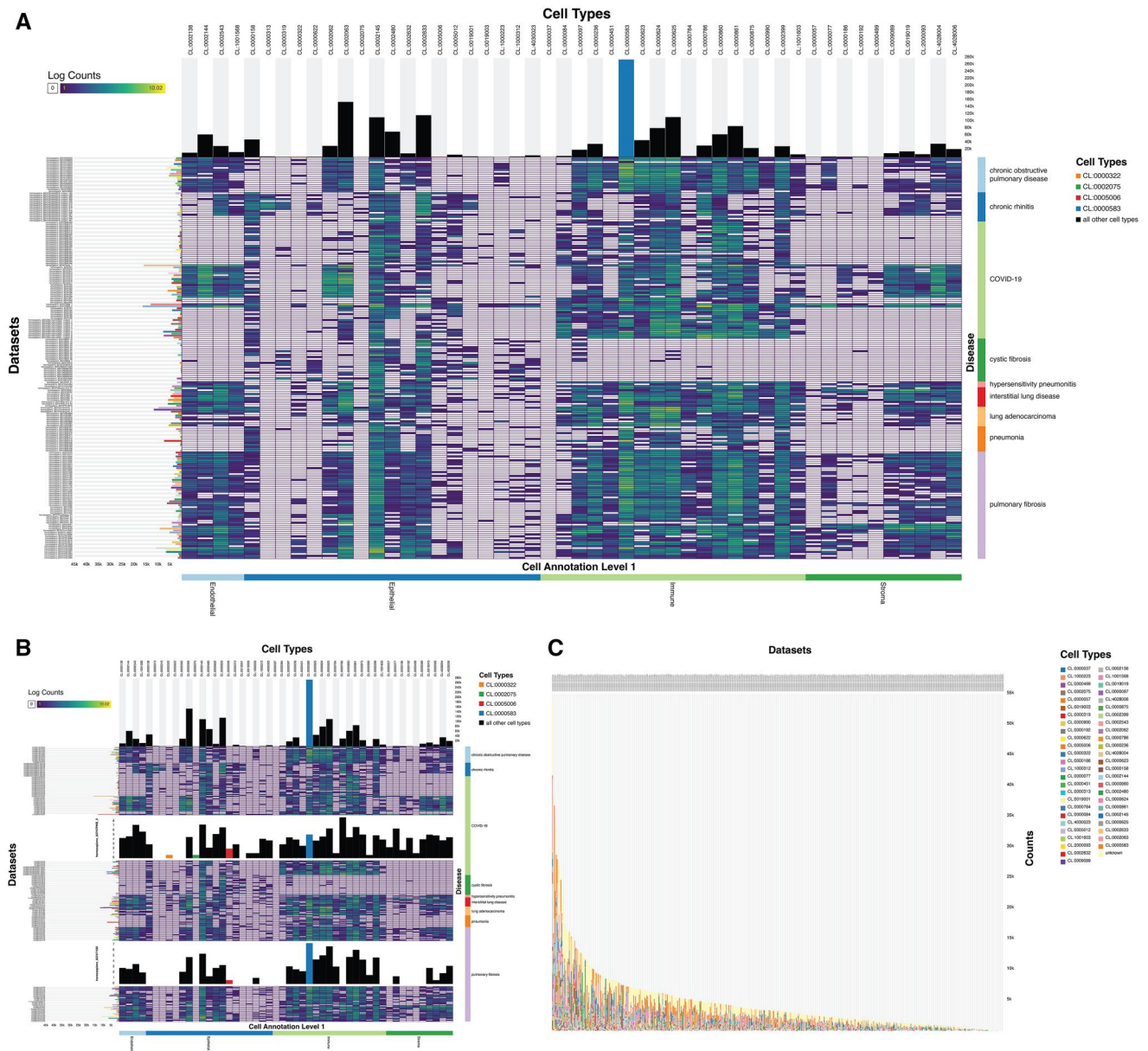


Figure 1 *scellop* showing populations in Human Lung Cell Atlas datasets. (A) *scellop* view sorted by diseases and cell type annotation. (B) Two datasets are converted to bar charts to compare. (C) Stacked bar chart view.

cystic fibrosis have a different cell type population, especially for immune cells (highlighted in Fig. S2). Certain datasets of donors with covid have different populations, so we can expand these datasets to compare them to different diseases (Fig. 1B). The traditional stacked bar chart view can also be shown (Fig. 1C). The amount of datasets complicates comparisons, especially those separated by distance (Talbot *et al.* 2014). Absent cell types and small fractions are hard to directly see, compared to *scellop*'s main view. Individual datasets cannot be examined in a detailed view. Although datasets can be grouped in the stacked bar chart view, cell types hierarchies cannot be indicated. Because the *scellop* viewer can sort by subsequent hierarchy orders, we can show cell type relations. Thus, *scellop* is better suited for the identified user tasks than the commonly used stacked bar charts. Overall, the redesigned visualization

approach proposed here allows for more granular exploration of cell populations. To facilitate inclusion in presentations and manuscripts, *scellop* views can also be exported as high-resolution images.

4 Discussion

scellop allows users to better explore cell populations with its interactive viewer, with easier pattern detection with its heatmap overview, and increased accuracy in comparing populations with its expandable bar charts. It better supports all of the user tasks identified via the design study, and can be integrated in Python and web environments for easy usage. *scellop* can also be used in other domains where stacked bar chart visualizations are prevalent, such as in metagenomics (Xiang *et al.* 2025,

Zhang *et al.* 2025). A potential extension of *scellop* can include a network graph for hierarchical cell types, such as Collapsible Tree (Gao *et al.* 2024). Sorting and filtering by hierarchies is supported by *scellop*, showing their relation can aid the exploration process. Hierarchical features would also support datasets with different cell type granularities better. Additionally, while *scellop* supports the widely used AnnData data format, data loading options can be expanded to alternative file formats.

Acknowledgements

The authors wish to thank Trevor Manz for their support with using anywidget, and Yan Ma for supporting the progress of this project.

Supplementary material

Supplementary material is available at *Bioinformatics Advances* online.

Conflicts of interest

N.G. is a co-founder and equity owner of Datavisyn. The remaining authors declare no competing interests.

Funding

This work was supported by the National Institutes of Health [OT2 OD033758 to N.G.].

Data availability

The code underlying this article are available at <https://github.com/hms-dbmi/scellop>. Data for Fig. 1 is available at <https://cellxgene.cziscience.com/collections/6f6d381a-7701-4781-935c-db10d30de293>.

References

- Bertin J. *La Graphique et le traitement graphique de l'information*. Nouvelle bibliothèque scientifique. Paris: Flammarion, 1977.
- Bostock M, Ogievetsky V, Heer J. D3 data-driven documents. *IEEE Trans Vis Comput Graph* 2011;**17**:2301–9. <https://doi.org/10.1109/TVCG.2011.185>.
- Cannoodt* R, Deconinck* L, Couckuyt* A *et al.* Funkyheatmap: visualising data frames with mixed data types. *J Open Source Softw* 2025;**10**:7698. <https://doi.org/10.21105/joss.07698>.
- Cleveland WS, McGill R. Graphical perception: theory, experimentation, and application to the development of graphical methods. *J Am Stat Assoc* 1984;**79**:531–54. <https://doi.org/10.1080/01621459.1984.10478080>.
- Domínguez Conde C, Xu C, Jarvis LB *et al.* Cross-tissue immune cell analysis reveals tissue-specific features in humans. *Science* 2022;**376**:eabl5197. <https://doi.org/10.1126/science.abl5197>.
- Fernandez NF, Gundersen GW, Rahman A *et al.* Clustergrammer, a web-based heatmap visualization and analysis tool for high-dimensional biological data. *Sci Data* 2017;**4**:170151. <https://doi.org/10.1038/sdata.2017.151>.
- Gao Y, Patro R, Jiang P. Collapsible tree: interactive web app to present collapsible hierarchies. *Bioinformatics* 2024;**40**:btac645. <https://doi.org/10.1093/bioinformatics/btac645>.
- Giovannangeli L, Giot R, Auber D *et al.* Impacts of the numbers of colors and shapes on outlier detection: from automated to user evaluation. arXiv: 2103.06084, 10 March 2021, preprint: not peer reviewed.
- Jain S, Pei L, Spraggins JM, *et al.* Advances and prospects for the human BioMolecular atlas program (HuBMAP). *Nat Cell Biol* 2023;**25**:1089–100. <https://doi.org/10.1038/s41556-023-01194-w>.
- Li Y, Du Y, Wang M *et al.* CSER: a gene regulatory network construction method based on causal strength and ensemble regression. *Front Genet* 2024;**15**:1481787. <https://doi.org/10.3389/fgene.2024.1481787>.
- Manz T, Abdennur N, Gehlenborg N. Anywidget: reusable widgets for interactive analysis and visualization in computational notebooks. *Joss* 2024;**9**:6939. <https://doi.org/10.21105/joss.06939>.
- McKinney W. Data structures for statistical computing in python. *SciPy* 2010:56–61.
- Nobre C, Zhu K, Mörtz E *et al.* Reading between the pixels: investigating the barriers to visualization literacy. *Proc CHI Conf Hum Factors Comput Syst*, 11 May 2024, 1–17. <https://doi.org/10.1145/3613904.3642760>.
- Okendo J, Okanda D, Mwangi P *et al.* Proteomic deconvolution reveals distinct immune cell fractions in different body sites in SARS-Cov-2 positive individuals. Preprint, *Health Informatics* **23** Jan 2022. <https://doi.org/10.1101/2022.01.21.22269631>.
- Perin C, Dragicevic P, Fekete JD. Revisiting bertin matrices: new interactions for crafting tabular visualizations. *IEEE Trans Vis Comput Graph* 2014;**20**:2082–91. <https://doi.org/10.1109/TVCG.2014.2346279>.
- Shneiderman B. The eyes have it: a task by data type taxonomy for information visualizations. *Proc 1996 IEEE Symp Vis Lang* 1996, 336–43. <https://doi.org/10.1109/VL.1996.545307>.
- Sikkema L, Ramírez-Suástegui C, Strobl DC, *et al.* An integrated cell atlas of the lung in health and disease. *Nat Med* 2023;**29**:1563–77. <https://doi.org/10.1038/s41591-023-02327-2>.
- Sultana Z, Khatri R, Yousefi B *et al.* Spatiotemporal interaction of immune and renal cells controls glomerular crescent formation in autoimmune kidney disease. *Nat Immunol* 2025;**26**:1977–88. <https://doi.org/10.1038/s41590-025-02291-8>.
- Talbot J, Setlur V, Anand A. Four experiments on the perception of bar charts. *IEEE Trans Vis Comput Graph* 2014;**20**:2152–60. <https://doi.org/10.1109/TVCG.2014.2346320>.
- Tseng C, Quadri GJ, Wang Z *et al.* Measuring categorical perception in color-coded scatterplots. *Proc 2023 CHI Conf Hum Factors Comput Syst*, 19 Apr 2023, 1–14. <https://doi.org/10.1145/3544548.3581416>.

- Virshup I, Bredikhin D, Heumos L, *et al.* The scverse project provides a computational ecosystem for single-cell omics data analysis. *Nat Biotechnol* 2023;**41**:604–6. <https://doi.org/10.1038/s41587-023-01733-8>.
- Virshup I, Rybakov S, Theis FJ *et al.* Anndata: access and store annotated datamatrices. *Joss* 2024;**9**:4371. <https://doi.org/10.21105/joss.04371>.
- Xiang G, Wang Y, Ni K *et al.* Nasal *Staphylococcus aureus* carriage promotes depressive behaviour in mice via sex hormone degradation. *Nat Microbiol* 2025;**10**:2425–40. <https://doi.org/10.1038/s41564-025-02120-6>.
- Yu X, Chen YA, Conejo-Garcia JR *et al.* Estimation of immune cell content in tumor using single-cell RNA-seq reference data. *BMC Cancer* 2019;**19**:715. <https://doi.org/10.1186/s12885-019-5927-3>.
- Zhang X, Li Y, Qin Y *et al.* Vientovirus capsid protein mimics auto-antigens and contributes to autoimmunity in sjögren's disease. *Nat Microbiol* 2025;**10**:2591–602. <https://doi.org/10.1038/s41564-025-02115-3>.