Comp683: Computational Biology

Lecture 14

February 28, 2024

Today

- Finish cPCA
- Combining disjoint single-cell datasets
- Sketching, or principled downsampling for single-cell data.

Sign up for project proposal presentations:
`https://docs.google.com/spreadsheets/d/1T3bDGpmppFo6VTtrlhapmIW4mR8JXduNQPGm4HbaPzc/edit?usp=sharing`

Recap of the Optimization Problem of cPCA

Given a contrast parameter $\alpha \geq 0$ that quantifies the trade-off between having high target variance and low background variance, cPCA computes the contrastive direction $\mathbf{v}$ by optimizing

$$\mathbf{v}^* = \text{argmax}_{\mathbf{v} \in \mathbb{R}^d_{\text{unit}}} \; \lambda_X(\mathbf{v}) - \alpha \lambda_Y(\mathbf{v})$$

This problem can be rewritten as

$$\mathbf{v}^* = \text{argmax}_{\mathbf{v} \in \mathbb{R}^d_{unit}} \; \mathbf{v}^T \left(C_X - \alpha C_Y \right) \mathbf{v}$$

What is happening here and what does this remind you of?

Given a contrast parameter $\alpha \geq 0$ that quantifies the trade-off between having high target variance and low background variance, cPCA computes the contrastive direction $\mathbf{v}$ by optimizing

$$\mathbf{v}^* = \text{argmax}_{\mathbf{v} \in \mathbb{R}^d_{\text{unit}}} \; \lambda_X(\mathbf{v}) - \alpha \lambda_Y(\mathbf{v})$$

This problem can be rewritten as

$$\mathbf{v}^* = \text{argmax}_{\mathbf{v} \in \mathbb{R}^d_{unit}} \; \mathbf{v}^T \left(C_X - \alpha C_Y \right) \mathbf{v}$$

cPCA is Quite Simple!

Algorithm 1 cPCA for a Given α

Inputs: target data $\{\mathbf{x}_i\}_{i=1}^n$; background data $\{\mathbf{y}_i\}_{i=1}^m$; contrast parameter α; the number of components k.

Centering the data $\{\mathbf{x}_i\}_{i=1}^n$, $\{\mathbf{y}_i\}_{i=1}^m$.
Calculate the empirical covariance matrices:

$$C_X = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i^T, C_Y = \frac{1}{m}\sum_{i=1}^{m} \mathbf{y}_i\mathbf{y}_i^T.$$

Perform eigenvalue decomposition on

$$C = (C_X - \alpha C_Y).$$

Compute the the subspace $V \in \mathbb{R}^k$ spanned by the top k eigenvectors of C.
Return: the subspace V.

Figure: Just do eigendecomposition on **C** and consider the eigenvectors corresponding to the top k eigenvalues of C.

Effect of Varying α

For $\alpha = 0$, cPCA will create directions that maximize the target variance. For higher α, directions with smaller background variance become more important.

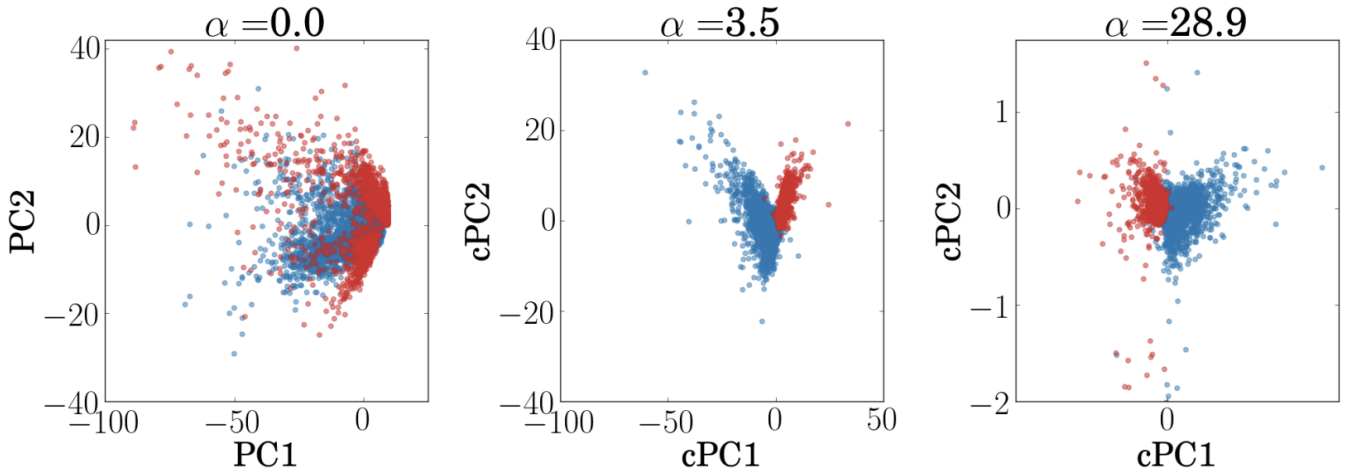


Figure: from Abid *et al.* Nature Communications. 2018. This dataset is visualizing cells from two different samples.

Recap

We have now seen Meld vs Cydar vs Milo vs cPCA (if we have a background). Thoughts? What would you do to compare your samples from disparate clinical or experimental groups?

Combining Multiple Single-Cell Datasets

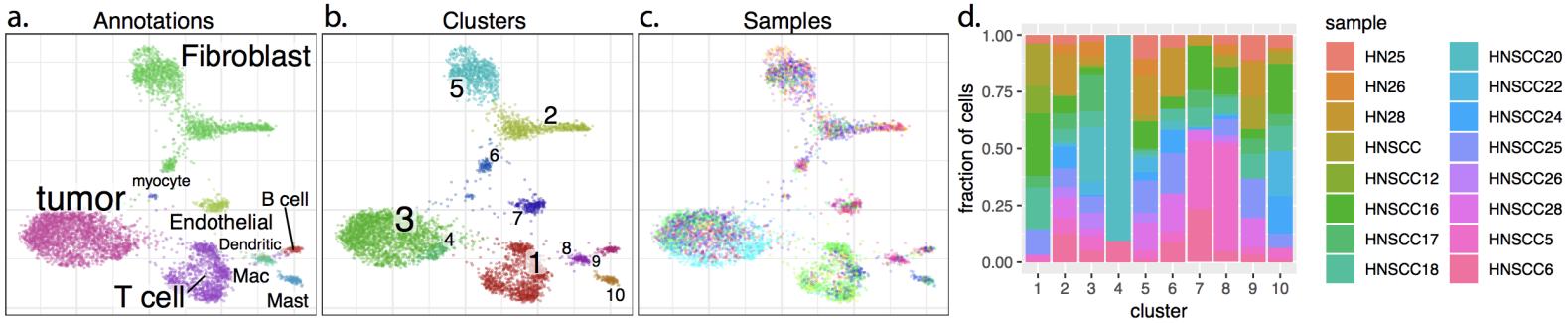


Figure: from Barkas *et al.* Nature Methods. 2019. Conos looks at how to integrate cells from multiple datasets (patients, tissues, etc.)

- The problem is a bit different from batch effect effect correction where you can identify technical artifacts and get rid of them. Cell-populations might be completely missing from particular datasets.

Conos Overview: Construct a Joint Between-Cell Graph

The goal is to establish a unified graph representation of the multiple single-cell datasets. Specifically, to infer cell-populations across all datasets, Conos seeks to infer inter-cell edges between datasets.

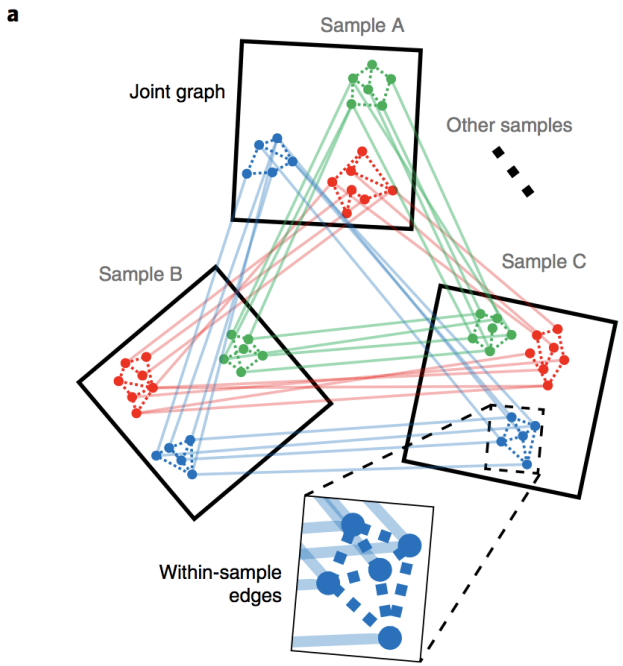


Figure: from Barkas *et al.* Nature Methods. 2019.

Pairwise Dataset Alignment

- As a pre-processing step, choose a set of high-variance genes. (The authors use 2,000).
- For a pair of datasets, i and j, let G_i and G_j denote their corresponding set of features measured per cell. Then consider only features that are measured in both datasets (so $G_i \cap G_j$)
- The similarity between cells K and l in datasets i and j is $w_{kl} = \exp(-\frac{||M_k^i - M_l^j||}{\sigma})$

Creating the Joint Graph

- Use w_{kl} for k-NN gaphs
- For **inter-sample edges**, connect each cell to its 15 nearest neighbors by default
- For **intra-sample edges**, connect each cell to its 5 nearest neighbors.
- Create joint clusters by clustering the graph with a graph-partitioning method.

Controlling Mixing Between Datasets

- Add a k_1 parameter or mixing parameter that allows for an increase of the nearest neighbor search radii, k. Control k_1 with an alignment strength parameter, $k_1 = \alpha^2 K_{\text{max}}$ (K_{max} is the maximum number of total cells across samples.

α ranges between 0 and 1 and 0 corresponds to alignment with no addition edges, and 1 corresponds to a full alignment.

This is followed by a pruning step...

They have a little strategy to reduce maximal degree closer to k and to make the graph less dense.

- Order nodes from highest to lowest degree
- For each node, order edges by the degree of target vertices (high to low)
- Algorithm goes through nodes and corresponding edges and removes an edge if the degrees of both incident nodes are larger than a specific cutoff, k_0

Rebalance Edge Weights

- Since samples are often collected across conditions, the authors wanted to provide flexibility to control how likely pairs of cell populations are to be mapped to each other, between conditions. Specifically, balance edge weights between cells connected between the same or different values of a factor (e.g. experimental vs control or experiment type.).

The solution is to minimize the following,

$$\sum_{l=1}^{N_{\text{factors}}} \sum_{s=1}^{N_{\text{cells}}} \left| \frac{\sum_{t \in \text{adj}_l(s)} w_{st}}{\sum_{t \in \text{adj}(s)} w_{st}} - \frac{1}{N^s_{\text{factors}}} \right|$$

Unpacking...

$$\sum_{l=1}^{N_{\text{factors}}} \sum_{s=1}^{N_{\text{cells}}} \left| \frac{\sum_{t \in \text{adj}_l(s)} w_{st}}{\sum_{t \in \text{adj}(s)} w_{st}} - \frac{1}{N^s_{\text{factors}}} \right|$$

- N_{factors} is the total number of factor levels
- N_{cells} is the total number of cells.
- $adj(s)$ is the set of cells adjacent to cell s.
- $adj_l(s)$ is the set of cells adjacent to s and belong to factor level l.
- w_{st} is the weight of the edge between cells s and t
- N^s_{factors} is the number of different factors of cells connected to s.

Imbalance Between Factor l and cell s

For their minimization they first estimate the imbalance ratio for a cell s and a factor level, l as,

$$u_{sl} = N^s_{\text{factors}} \frac{\sum_{t \in \text{adj}_l(s)} w_{st}}{\sum_{t \in \text{adj}(s)} w_{st}}$$

Using Imbalance to Update Edge Weights

Edge weights are updated using the imbalance computed in the previous slide as,

$$w_{st} = \frac{w_{st}}{\sqrt{u_{sl}\, u_{tl_s}}}.$$

- Here l_c denotes the factor level of cell c.
- This process is repeated 50 times.

Effect of Alignment Strength

Here is an example varying alignment strength on a dataset containing cells from multiple technologies.

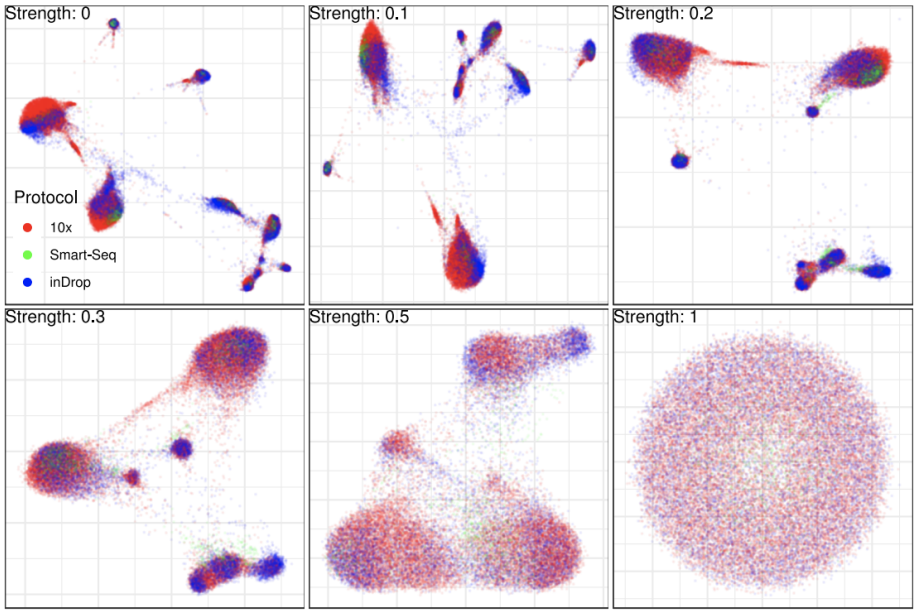


Figure: from Barkas *et al.* Nature Methods. 2019.

Example 1: Bone Marrow and Chord Blood

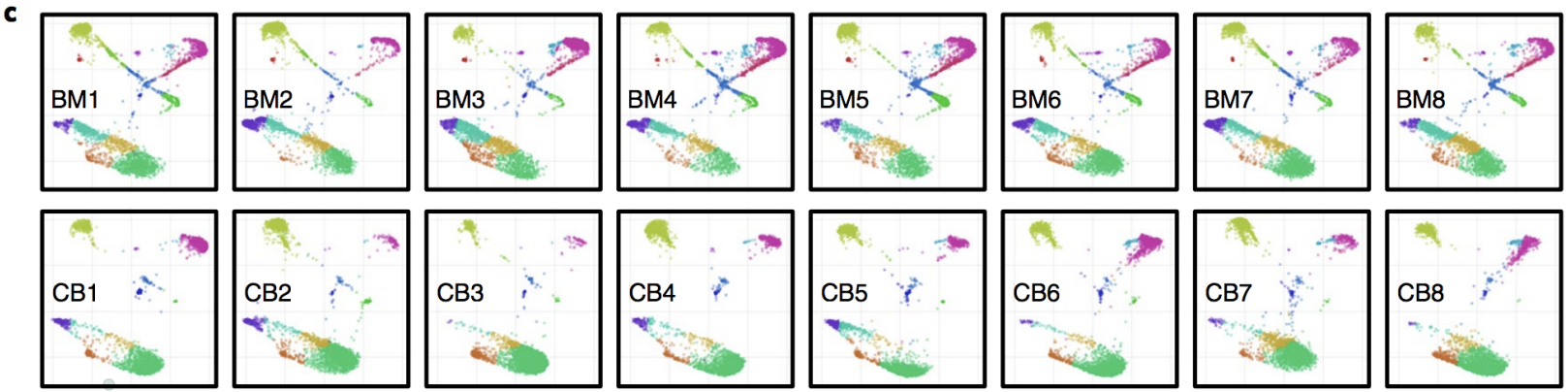


Figure: from Barkas *et al.* Nature Methods. 2019. You can see similarities and differences between cell-populations in each dataset.

Visualizing the Joint Graph

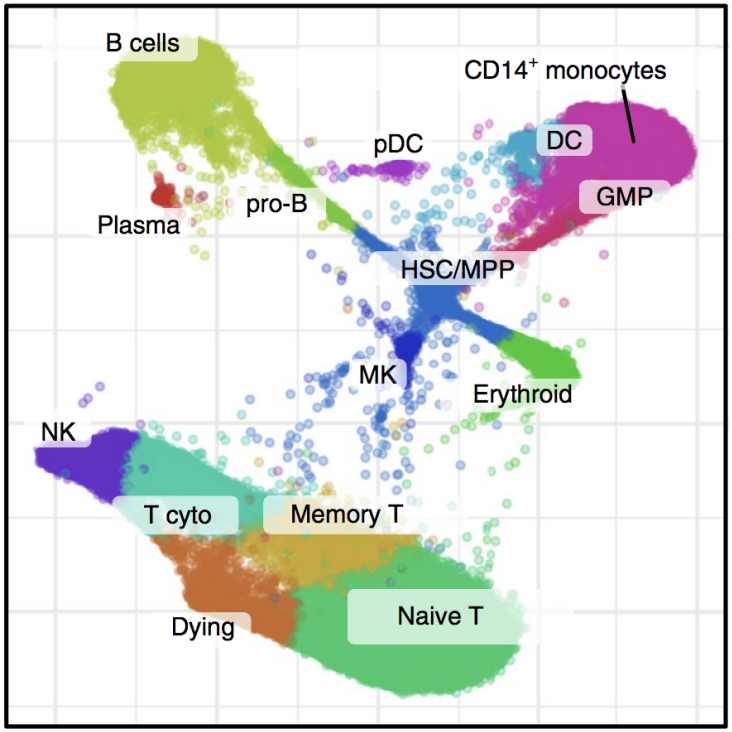


Figure: from Barkas *et al.* Nature Methods. 2019. The layout of the joint graph is determined by LargeVis.

Predicting a Cell's Label from the Graph Structure

Label propagation can be used to predict a cell's label based on the labels of neighboring cells.

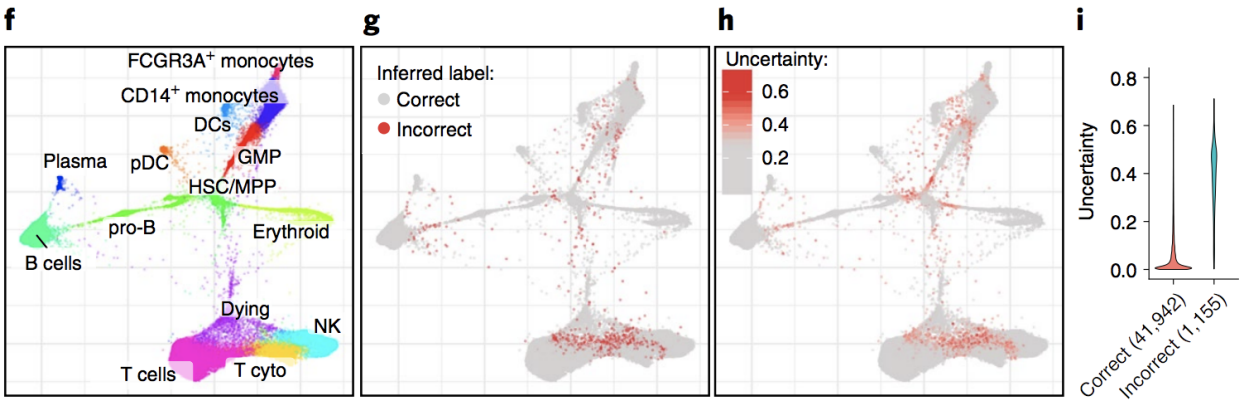


Figure: from Barkas *et al.* Nature Methods. 2019. Cells colored red represent those with incorrect predictions.

Run-time Based on Cells and Datasets

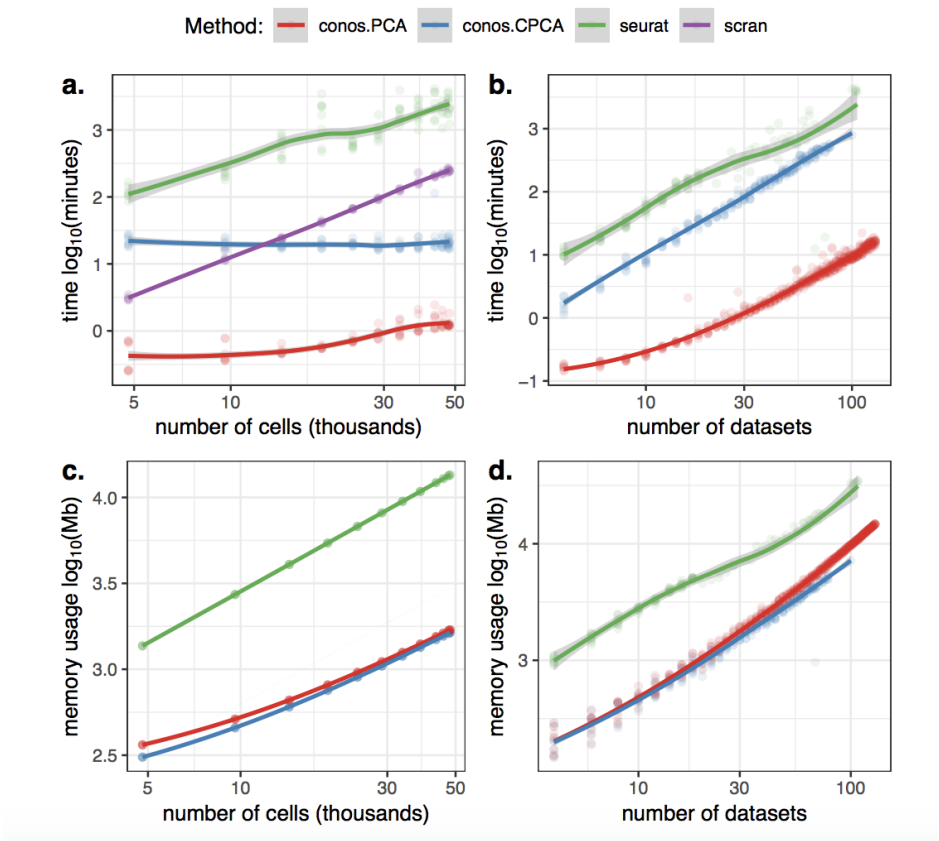


Figure: from Barkas *et al.* Nature Methods. 2019.

Switching Gears $\rightarrow$ Issues We Have Seen

Consider a mass cytometry or scRNA seq (:D) dataset from a huge clinical cohort (hundreds of samples, each with 100s of thousands of cells)

- We can't deal with all of these cells at once (clustering, visualization, etc)
- Downsampling is biased towards the higher frequency cell-types
- **Compression:** What if we want to take a representative subsample of the entire dataset?

An Important Task from a Practical Viewpoint: Patient Outcome Prediction

If you want to build a regression model, get a prediction for each cell and pool predictions for cells across patients, you are limited by the number of cells

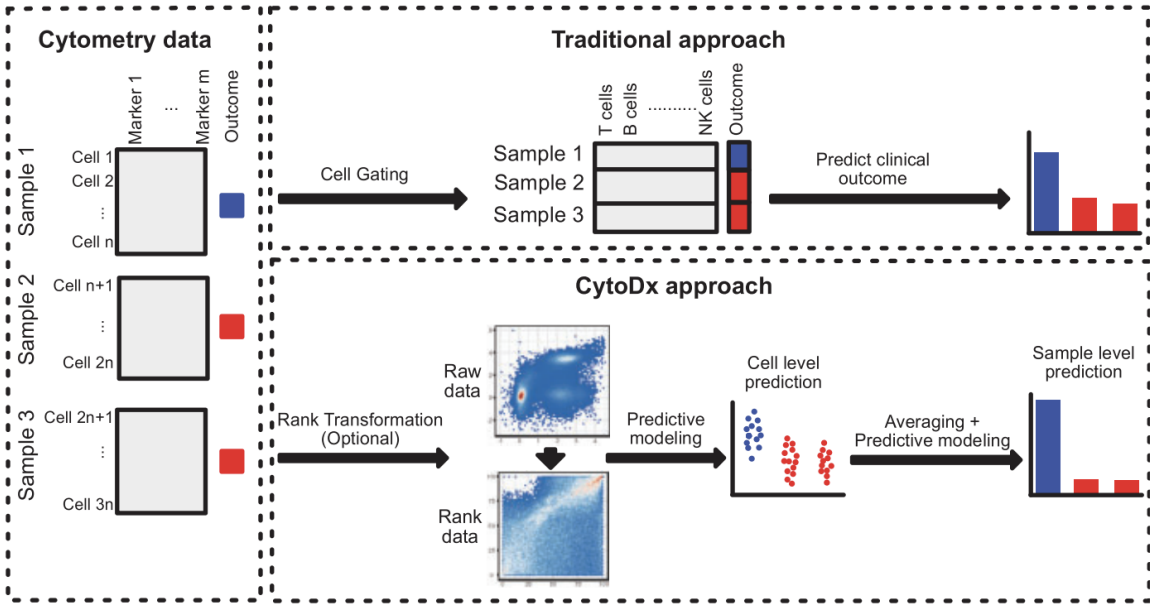


Figure: from Hu *et al.* Bioinformatics. 2018.

Previously Seen in SPADE

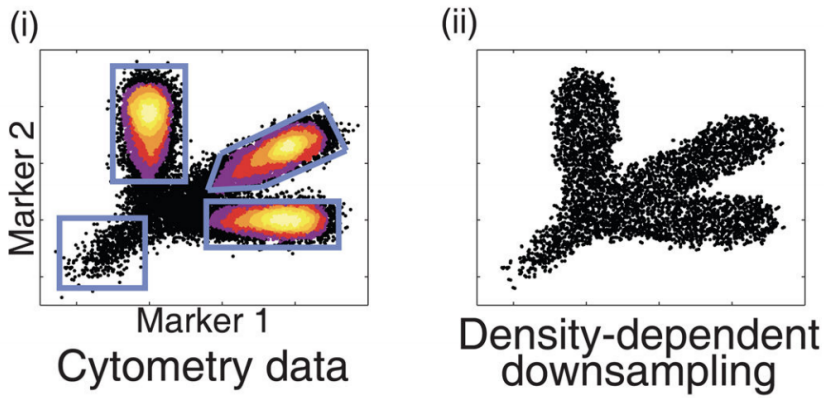


Figure: from Qiu *et al.* Nature Biotech. 2012.

Sampling Effectively Across the Entire Dataset

If you wanted to quickly grab some cells that were well-distributed, what would you do? How would you figure out how many subsampled points you need to cover the entire cellular landscape?

Welcome Geometric Sketching

Instead of randomly selecting a subset of points for visualization, etc., why not do something a bit more smart, according to the inherent data geometry.

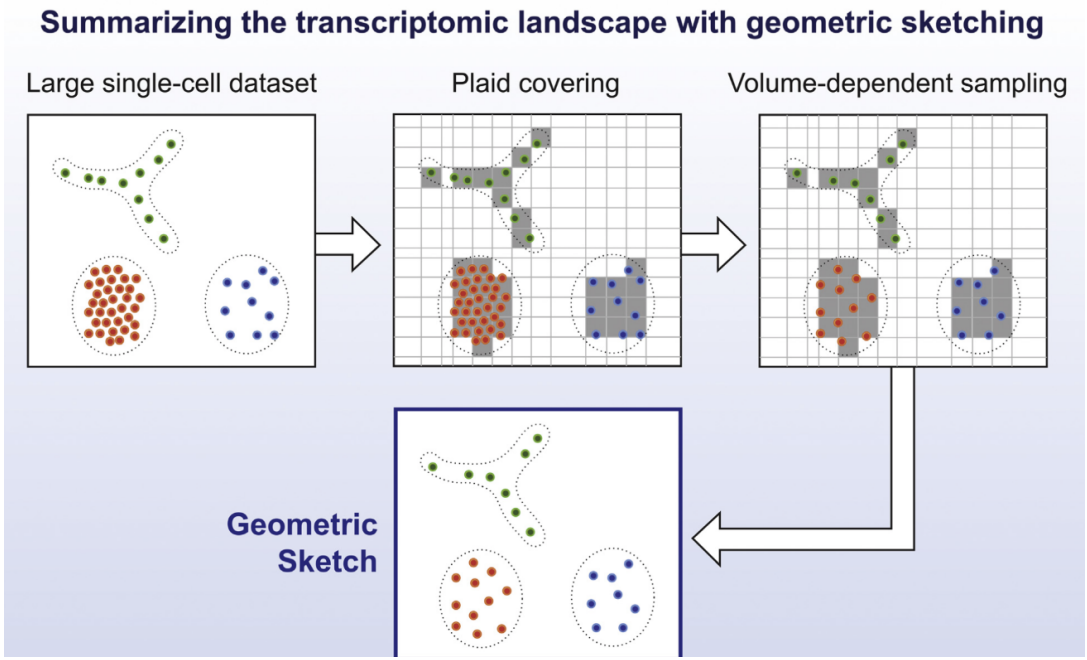


Figure: from Hie *et al.* Cell Systems. 2019

Sketching Algorithms

In general a sketching algorithm takes a dataset and compresses it, such that you can still effectively carry out a query that you wanted to do on the original data.

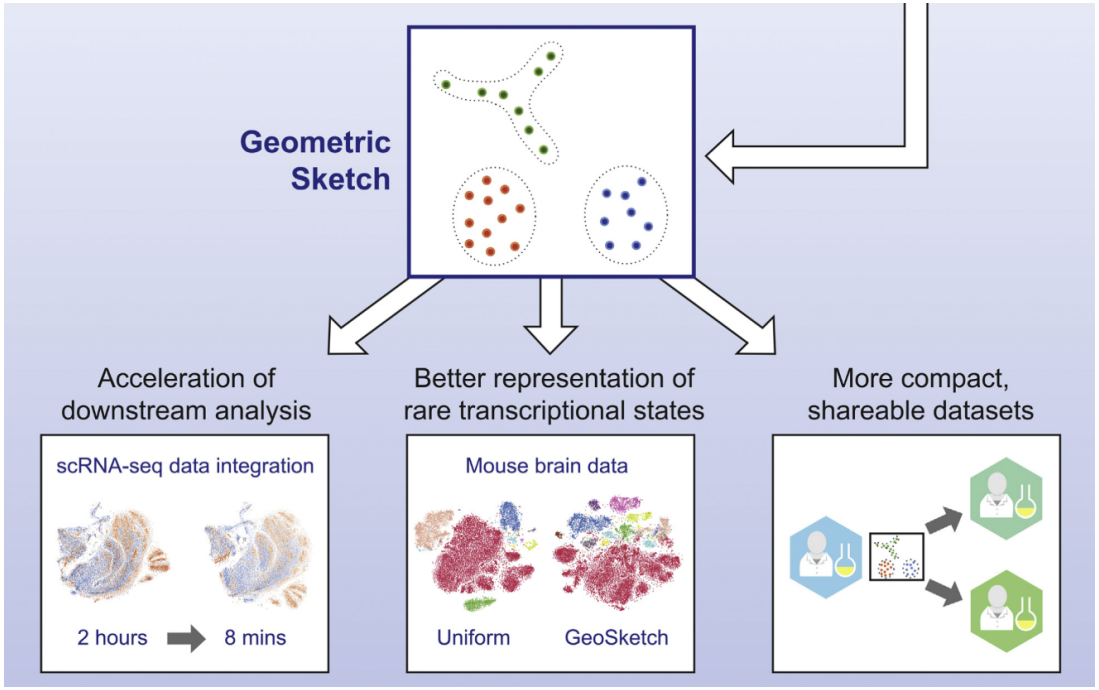


Figure: from Fig. 2 of Hie *et al.* Cell Systems. 2019. Perhaps we need an accurate estimate of cell-population frequencies, etc.

Sketching Algorithms

For a brilliant tutorial and description of sketching, refer to the following tutorial, `https://nips.cc/virtual/2020/public/tutorial_7bef20627bb50052e352b9653c3bca53.html`

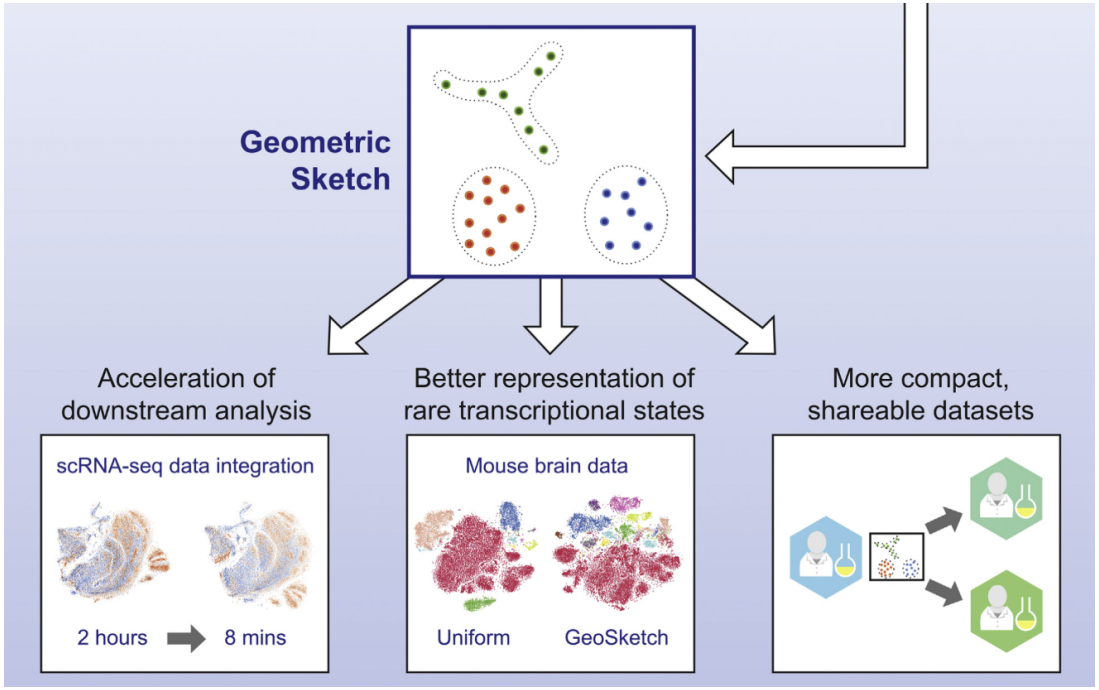


Figure: from Fig. 2 of Hie *et al.* Cell Systems. 2019.

Formulation of Sketching Problem

- Let $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be a collection of m-dimensional cells ($\mathbf{x}_i \in \mathbb{R}^m$).
- We seek a down-sampled subset, $S \subset \mathcal{X}$, which can be used for downstream analysis, specifically to understand the salient characteristics of $\mathcal{X}$

Formulation of Sketching Problem

You can measure the quality of a particular sketch, $\mathcal{S}$ wrt a dataset, $\mathcal{X}$ through Hausdorff Distance as,

$$d_H(\mathcal{X}, \mathcal{S}) = \max_{\mathbf{x} \in \mathcal{X}} \left\{ \min_{\mathbf{s} \in \mathcal{S}} d(\mathbf{x}, \mathbf{s}) \right\}$$

Here, d is any distance or dissimilarity measure that you can compute between $\mathbf{x}$ and $\mathbf{s}$.

A Robust Formuation of Hausdorff Distance

The regular formulation of Hausdorff distance is sensitive to outliers. An alternative partial HD measure is defined as follows,

$$d_{HK}(\mathcal{X}, \mathcal{S}) = K^{\text{th}}_{\mathbf{x}\in\mathcal{X}} \left\{ \min_{\mathbf{s}\in\mathcal{S}} d(\mathbf{x}, \mathbf{s}) \right\}$$

This looks at the Kth largest distance. Define $q = 1 - K/|\mathcal{X}|$ The q can therefore be varied.

Plaid Covering

The plaid covering, $\mathcal{C}$ represents a collection of m-dimensional equal volume hypercubes. The same number of cells are then sampled from each hypercube. In practice, they define the same number of hypercubes as desired cells (k) and therefore sample 1 cell per cube.

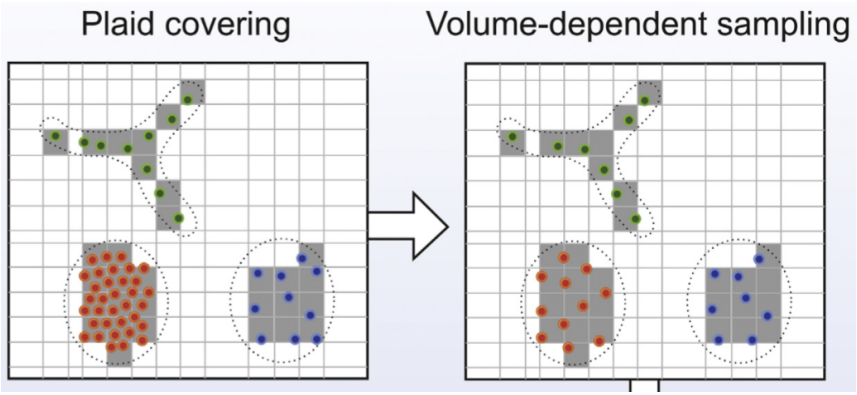


Figure: from Fig. 2

Parameters for Sketching

Geometric Sketching Algorithm Parameters

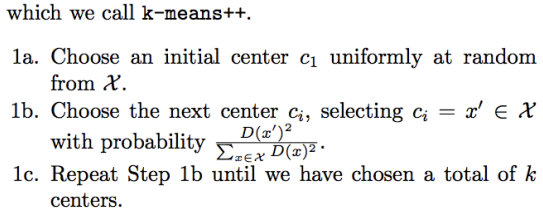
Parameter	Type	Default Value	Notes
Sketch size (k)	Integer between 0 and total number of cells, inclusive	N/A	The desired sketch size is chosen depending on the amount of compute resources available and the algorithmic complexity of downstream analyses; smaller sketches omit more cells but will accelerate analysis while preserving much of the transcriptional heterogeneity.
Number of covering boxes ($\lvert\mathcal{C}\rvert$)	Integer between 1 and total number of cells, inclusive	Equal to desired sketch size k	Converges to uniform sampling as parameter increases; a number of covering boxes less than k may yield a coarser picture of the transcriptional space, including overrepresentation of rare cell types, at the cost of an increased Hausdorff distance.

Figure: from Hie *et al.* Cell Systems. 2019.

Baseline Downsampling Method, k-means++

- Ref : `https://theory.stanford.edu/~sergei/papers/kMeansPP-soda.pdf`

k-means++ is an approach to better initialize k-means. The downsampling aspect is to find appropriate cluster centers. Cluster centers are added, such that, each addition is different from the previous center added.



which we call **k-means++**.

1a. Choose an initial center c_1 uniformly at random from $\mathcal{X}$.

1b. Choose the next center c_i, selecting $c_i = x' \in \mathcal{X}$ with probability $\frac{D(x')^2}{\sum_{x \in \mathcal{X}} D(x)^2}$.

1c. Repeat Step 1b until we have chosen a total of k centers.

Figure: $D(\cdot)$ is the smallest distance from a point and to any of the previously chosen cluster centers.

Baseline Downsampling Method, Spatial Random Sampling (SRS)

- Ref : `https://arxiv.org/pdf/1705.03566.pdf`
- A limited number of points are sampled based on their proximity to randomly sampled points on the unit sphere.

Results: Hausdorff Distance

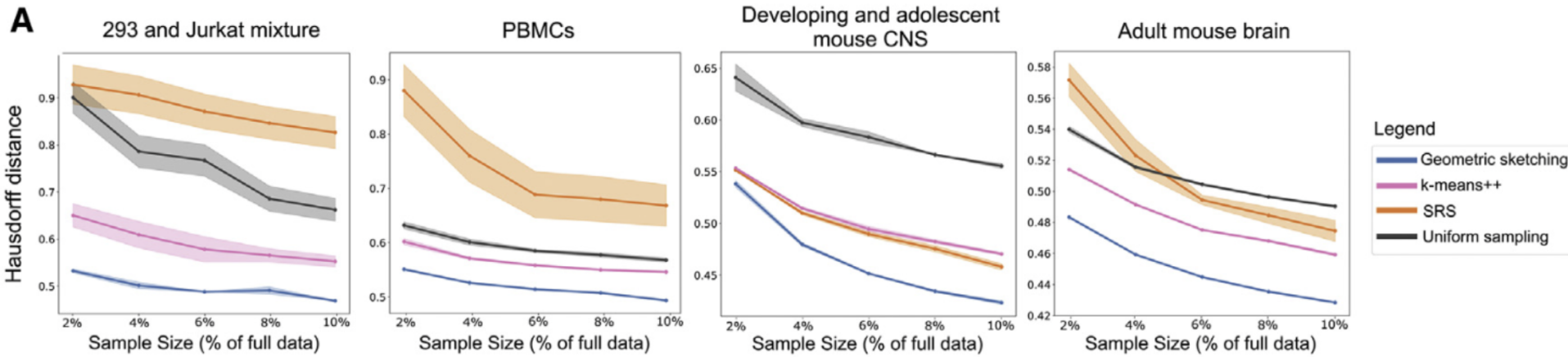


Figure: from Fig. 3. Note the superior performance of geometric sketching!

Using Robust Hausdorff

Recall the interplay between q and k for computing the robust hausdorff distance.

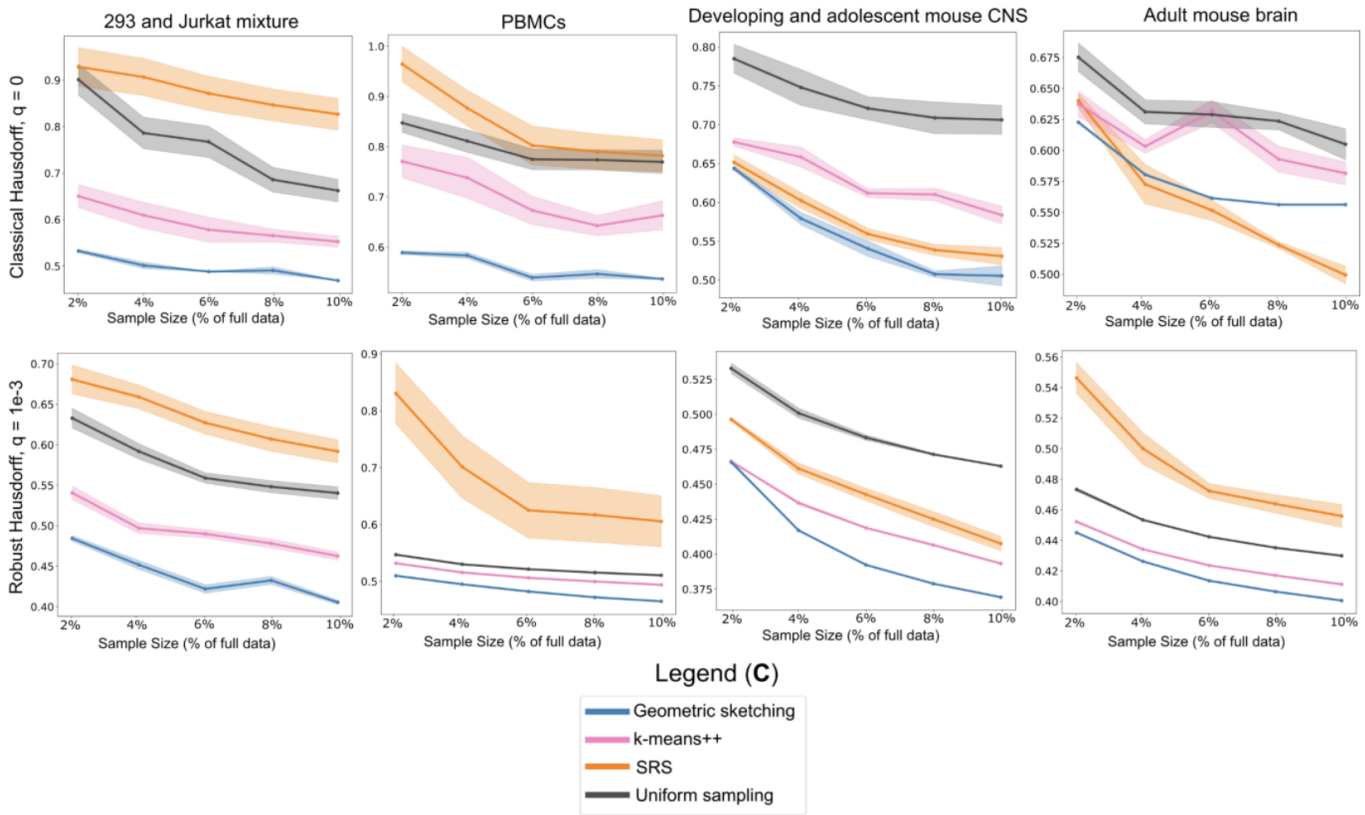


Figure: from Supplementary Figure S5.

Downsampled Cells Facilitate Faster Downstream Tasks

for example : batch effect correction

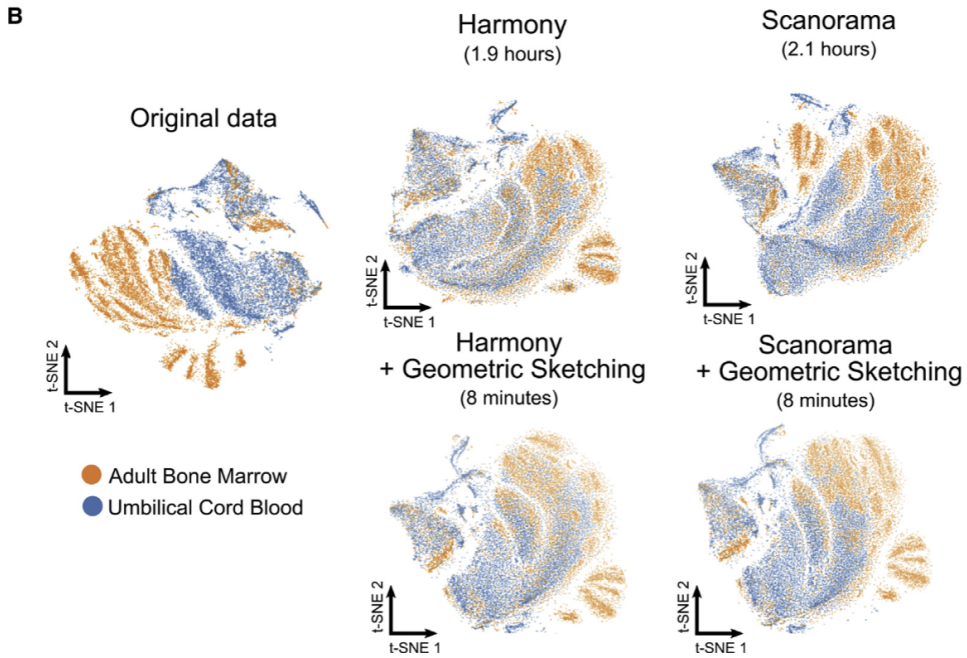


Figure: from Fig. 5

Rarer Populations are Sufficiently Represented

Check out (for example) Ependymal (cells colored brown)

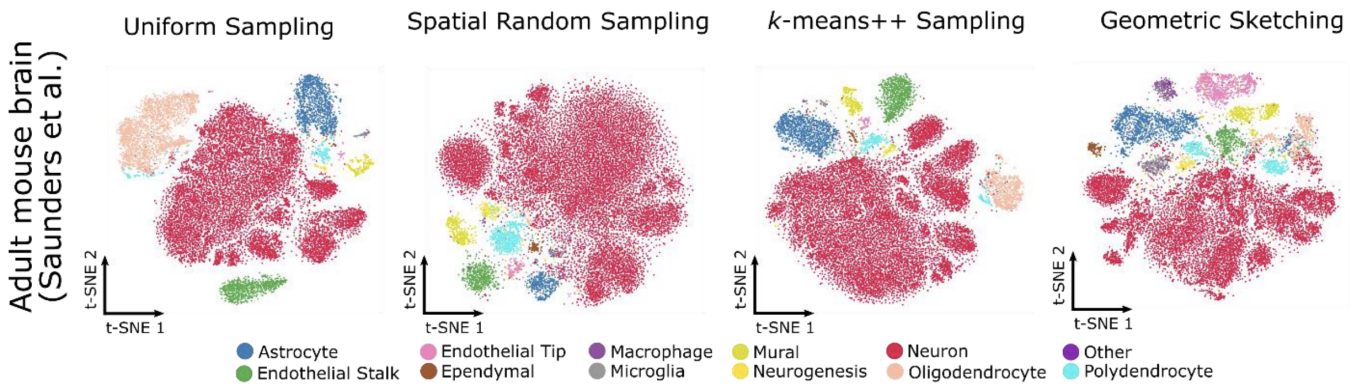


Figure: from Supplementary Figure S1

Counts of Rarest Cell Type

In sketches containing 2% of the dataset, the methods were compared in terms of their representation of the rarest cell-type.

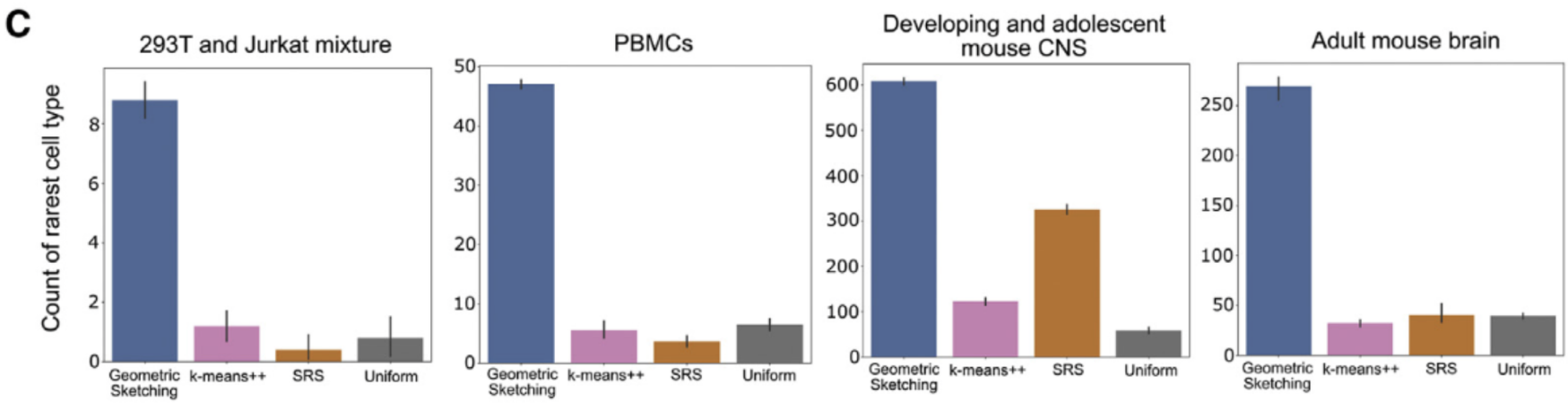


Figure: from Fig. 3

Interpretation wrt Clustering

Louvain on downsampled cells produced the ability to differentiate between inflammatory macrophage and macrophage, and to observe appropriate gene expression differences.

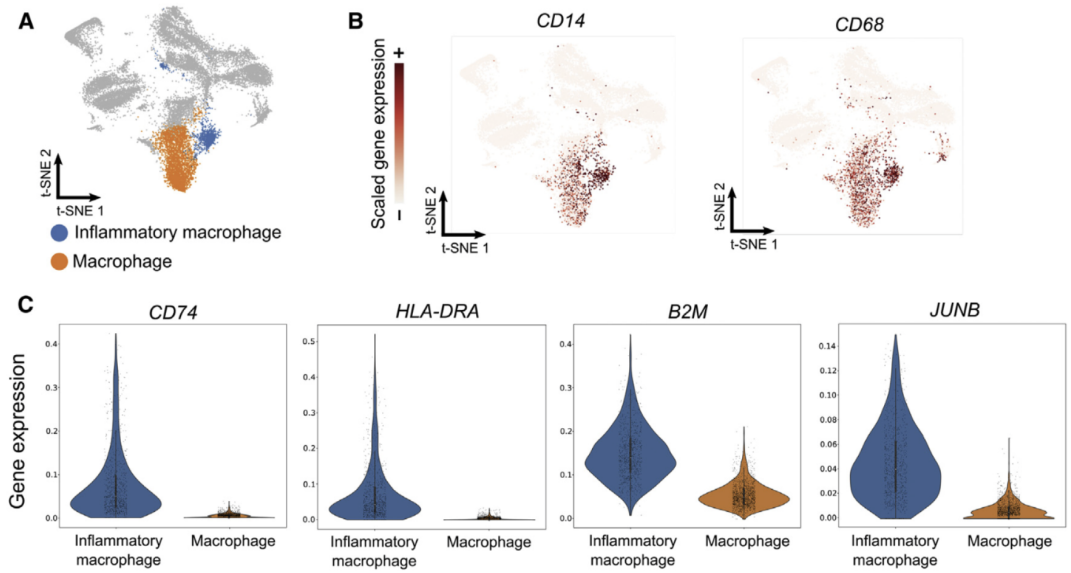


Figure: from Fig. 4

Recap and Unsolicited Opinions

- Beautiful concept. Similar to what we have seen with Cydar
- Through the plaid covering, a subset of cells can be sampled to create a quality sketch (in terms of Hausdorff Distance)
- The reduced set of cells has good representation across many cell types, including those that are rare.
- **Still Missing:** ??