Published in final edited form as:
Cell Syst. 2019 June 26; 8(6): 483–493.e7. doi:10.1016/j.cels.2019.05.003.

Geometric Sketching Compactly Summarizes the Single-Cell Transcriptomic Landscape

Brian Hie[*,1], **Hyunghoon Cho**[*,1], **Benjamin DeMeo**[2,3], **Bryan Bryson**[4], and **Bonnie Berger**[†,1,2]

[1]Computer Science and Artificial Intelligence Laboratory, MIT, Cambridge, MA 02139, USA;

[2]Department of Mathematics, MIT, Cambridge, MA 02139, USA;

[3]Department of Biomedical Informatics, Harvard University, Cambridge, MA 02138, USA;

[4]Department of Biological Engineering, MIT, Cambridge, MA 02139, USA

Abstract

Large-scale single-cell RNA-sequencing (scRNA-seq) studies that profile hundreds of thousands of cells are becoming increasingly common, overwhelming existing analysis pipelines. Here, we describe how to enhance and accelerate single-cell data analysis by summarizing the transcriptomic heterogeneity within a dataset using a small subset of cells, which we refer to as a geometric sketch. Our sketches provide more comprehensive visualization of transcriptional diversity, capture rare cell types with high sensitivity, and reveal biological cell types via clustering. Our sketch of umbilical cord blood cells uncovers a rare subpopulation of inflammatory macrophages, which we experimentally validated. The construction of our sketches is extremely fast, which enabled us to accelerate other crucial resource-intensive tasks such as scRNA-seq data integration while maintaining accuracy. We anticipate our algorithm will become an increasingly essential step when sharing and analyzing the rapidly-growing volume of scRNA-seq data and help enable the democratization of single-cell omics.

eTOC

Single-cell RNA-sequencing (scRNA-seq) measures gene expression in millions of cells, providing unprecedented insight into biology and disease. These datasets, however, are becoming too large for conventional analysis methods. Our algorithm, geometric sketching, efficiently samples a small representative subset of cells from massive datasets while preserving biological complexity, highlighting rare cell states, and accelerating complex analyses like dataset

[*]These authors contributed equally.
[†]Lead Contact (bab@mit.edu)
AUTHOR CONTRIBUTIONS
All authors conceived the algorithm. B. Hie and H. Cho performed the computational experiments. B. Bryson performed the scRNA-seq experiments. B. Berger guided the research. All authors wrote the manuscript.

Publisher's Disclaimer: This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

DECLARATION OF INTERESTS
The authors declare no competing interests.

integration. Geometric sketching is an increasingly useful tool as the volume of scRNA-seq data explodes.

INTRODUCTION

Improvements in the throughput of single-cell profiling experiments, especially droplet-based single-cell RNA-sequencing (scRNA-seq), have resulted in datasets containing hundreds of thousands of cells (Angerer et al., 2017; Macosko et al., 2015; Zheng et al., 2017), with hundreds to thousands of gene expression measurements per cell. As these sequencing pipelines become cheaper and more streamlined, experiments profiling tens of millions of cells may become ubiquitous in the near future (Angerer et al., 2017), and consortium-based efforts like the Human Cell Atlas plan to profile billions of cells (Rozenblatt-Rosen et al., 2017). Leveraging this data to improve our understanding of biology and disease will require merging and integrating many cells across diseases and tissues (Hie et al., 2019), resulting in reference datasets with massive numbers of cells. Unfortunately, the sheer volume of scRNA-seq data being generated is quickly overwhelming existing analytic procedures, requiring prohibitive runtime or memory usage to produce meaningful insights (Angerer et al., 2017). This bottleneck limits the utility of these emerging large datasets to researchers with access to expensive computational infrastructure, and makes quick exploratory analyses impossible even for these researchers.

Here, we introduce an approach that intelligently selects a small subset of data (referred to as a “sketch”) that comprehensively represents the transcriptional heterogeneity within the full dataset. Because of their vastly reduced computational overhead, our sketches can be efficiently shared among researchers and be used to quickly identify important patterns in the full dataset to be followed up with in-depth analyses.

Currently, researchers often uniformly downsample a dataset to obtain a small subset to accelerate the initial data analysis (10x Genomics, 2017). Although this simple approach could be used to generate sketches of single-cell datasets, it is highly prone to removing rare cell types and negates the advantage of performing large-scale scRNA-seq experiments in the first place. Alternative sampling approaches that better consider the structure of the data, including *k*-means++ sampling (Arthur and Vassilvitskii, 2007) and spatial random sampling (SRS) (Rahmani and Atia, 2017a), have not yet been applied to the problem of obtaining informative sketches of scRNA-seq data to our knowledge. However, these data-dependent sampling techniques not only lack the ability to efficiently scale to large datasets, but also lack robustness to different experimental settings and produce highly unbalanced sketches that are ill-suited for downstream scRNA-seq analyses as we demonstrate in our experiments.

The key insight behind our sampling approach is that common cell types form dense clusters in the gene expression space, while rarer subpopulations may still inhabit comparably large regions of the space but with much greater sparsity. Rather than sample cells uniformly at random, we sample *evenly across the transcriptomic space*, which naturally removes redundant information within the most common cell types and preserves rare transcriptomic structure contained in the original dataset. We refer to our sampling method as “geometric sketching" because it obtains random samples based on the geometry, rather than the density, of the dataset (Figure 1).

Geometric sketching is extremely efficient, sampling from datasets with millions of cells in a matter of minutes and with an asymptotic runtime that is close to linear in the size of the dataset. We empirically demonstrate that our algorithm produces sketches that more evenly represent the transcriptional space covered by the data. We further show that our sketches enhance and accelerate downstream analyses by preserving rare cell types, producing visualizations that broadly capture transcriptomic heterogeneity, facilitating the identification of cell types via clustering, and accelerating integration of large scRNA-seq datasets. Moreover, we demonstrate how the sensitivity of geometric sketching to rare transcriptional states allows us to identify a previously unknown rare subpopulation of inflammatory macrophages in a human umbilical cord blood dataset, providing insight into a fundamental immunological process. As the size of single-cell data grows, geometric sketching will become increasingly crucial for the democratization of large-scale single-cell experiments, making key analyses tractable even for researchers without expensive computational resources.

RESULTS

Overview of Our Geometric Sketching Algorithm

The overall approach taken by the geometric sketching algorithm is illustrated in Figure 1. Geometric sketching aims to select a subset of cells (i.e., a sketch) from a large scRNA-seq dataset such that the subset accurately reflects the full transcriptomic heterogeneity, where the small size of the sketch enables fast downstream analysis. In order to effectively summarize the diversity of gene expression profiles within a dataset, the first step of our algorithm is to approximate the geometry of the transcriptomic space inhabited by the input data as a union of fixed shapes that admit succinct representation. In our work, we approximate the data with a collection of equal-sized, non-overlapping, axis-aligned boxes (hypercubes), which we refer to as a *plaid covering*. We use boxes instead of spheres to obtain a highly efficient greedy covering algorithm that helps us better cope with the increasing volume of scRNA-seq data. Once the geometry of data is approximated via plaid covering, we sample cells by first spreading the desired total sample count over the covering boxes as evenly as possible (based on a random ordering of the boxes), then choosing the assigned number of samples within each box uniformly at random. This process allows the samples to more evenly cover the gene expression landscape of the data, naturally diminishing the influence of densely populated regions and increasing the representation of rare transcriptional states. A more detailed description and theoretical analysis of our approach is provided in Method Details and a summary illustration of the geometric sketching workflow is provided in Figure 2.

Geometric Sketches Evenly Summarize the Transcriptomic Landscape

We first sought to quantify how well geometric sketching is able to evenly represent the original transcriptomic space by measuring the Hausdorff distance from the full dataset to a geometric sketch (Method Details). Intuitively, a low Hausdorff distance indicates that the points in a sketch are close to all points in the remainder of the dataset within the transcriptomic space, while a high Hausdorff distance indicates that there are some cells in the full dataset that are not well represented within the sketch. We benchmarked geometric sketching against uniform sampling as well as more complex, data-dependent strategies: *k*-means++ sampling (Arthur and Vassilvitskii, 2007) and spatial random sampling (SRS) (Rahmani and Atia, 2017a). Note that, to our knowledge, neither of these non-uniform sampling approaches have been previously considered for the problem of downsampling single-cell datasets. *k*-means++ works by randomly choosing an initial sample, then repeatedly sampling the next point such that more distant points from the current sample set have higher probability. SRS works by projecting the data onto the unit ball, sampling points uniformly across the surface of the ball, and picking the closest example from the dataset to each of those random points.

We used four scRNA-seq datasets of varying sizes and complexities to assess our method (Method Details; Table S1–S4): a 293T/Jurkat mixture with 4,185 cells (Zheng et al., 2017); a PBMC dataset with 68,579 cells (Zheng et al., 2017); a developing and adolescent mouse central nervous system (CNS) dataset with 465,281 cells (Zeisel et al., 2018); and an adult mouse brain dataset with 665,858 cells (Saunders et al., 2018). In all cases, we observed that geometric sketching obtains substantially better improvement under a robust Hausdorff distance measure (Method Details) than uniform sampling and the other data-dependent sampling methods, SRS and *k*-means++ (Figure 3A). The improvement in Hausdorff distance was consistent across sketch sizes ranging from 2% to 10% of the full dataset, providing quantitative evidence that our algorithm more evenly samples over the geometry of the dataset than do other methods.

Visualization of Geometric Sketches Reveals Transcriptional Diversity

We next set out to assess the ability of our geometric sampling approach to improve the low-dimensional visualization of scRNA-seq data, a common exploratory (and often computationally expensive) initial step in single-cell genomic analysis. From our two largest datasets of mouse nervous system, containing 465,281 and 665,858 cells each, we used a 2-dimensional *t*-SNE (Van Der Maaten and Hinton, 2008) to visualize a sketch containing 2% of the total dataset (sampled without replacement) obtained by geometric sketching.

The results, shown in Figure 3B, illustrate that the relative representations of cell types in geometric sketches can have striking differences compared to uniformly downsampled datasets. For instance, when obtaining a sketch of 2% of the dataset of adult mouse neurons (Saunders et al., 2018), clusters of macrophages, endothelial tip cells, and mural cells have only 59, 117, and 336 cells, respectively, in the uniform sample out of 1695, 3818, and 12083 cells in the full data, respectively. In contrast, these cell types have 326, 1022, and 875 cells, respectively, in the geometric sketch of the same size. Although these cell types are rare compared to neurons (428,051 cells in the full dataset), their substantially increased representation in our sketch suggests they inhabit a comparatively large portion of the transcriptional space. Similarly, on a dataset of 465,281 cells from the developing and adolescent mouse central nervous system (CNS) (Zeisel et al., 2018), we also observed a more balanced composition of cell types as determined by the original study’s authors

(Figure 3B). The rarest cell types are also more consistently represented in a geometric sketch than in sketches obtained by SRS or *k*-means++ (Figure S1A; Table S3–4). We also visualize the data with a uniform manifold approximation and projection (UMAP), an alternative method for computing 2-dimensional visualization embeddings (McInnes and Healy, 2018), with similar results as those produced by our *t*-SNE experiments (Figure S1B).

We note that our sampling algorithm is completely unsupervised and has no knowledge of the cell type labels, but naturally obtains a balanced composition of cell types by sampling more evenly across the entire transcriptional space. Indeed, on artificial data in which we controlled the relative volumes and densities of the clusters, geometric sketching samples the clusters proportionally to their relative volumes rather than their frequencies in the full dataset (Figure S2A), suggesting that the composition of different cell types in a geometric sketch more closely reflects the transcriptional variability of individual clusters rather than their frequency in the overall population. Our visualizations therefore reflect a geometric “map” of the transcriptional variability within a dataset, allowing researchers to more easily gain insight into rarer transcriptional states.

Rare Cell Types Are Better Preserved Within Geometric Sketches

As suggested by the above results, one of the key advantages of our algorithm is that it naturally increases the representation of rare cell types with sufficient transcriptomic heterogeneity in the subsampled data. Using the four datasets mentioned above, which include cell type labels provided by the original study authors, we evaluated the ability of our method to preserve the rarest cell type within each dataset. In particular, we focused on 28 293T cells (0.66% of the total number of cells in the dataset) in the 293T/Jurkat mixture, 262 dendritic cells (0.38%) in the PBMC dataset, 1695 macrophages (0.25%) among the adult mouse brain cells, and 2777 ependymal cells (0.60%) among the mouse CNS cells. In all datasets, the rare cell types are substantially more represented in the sketch obtained by our algorithm compared to other sampling techniques (Figure 3C). For example, a sketch that is 2% the size of the 665,858 mouse brain cells contains an average of 281 macrophages compared to only 31 cells from uniform sampling. Geometric sketching is able to better preserve rare cell types because the extent of transcriptional variation among rare cells is similar to that of common cells. To this end, we used the differential entropy of a multivariate Gaussian fit to each cell type as a proxy to its transcriptional diversity (Method Details; Table S1–S4). We also note that, within the geometric sketch, almost all of the rare cell types in each dataset have increased representation compared to the full data, where the representation of rare cell types gradually converges to that of uniform sampling as the sketch size increases (Figure S2B).

Clustering of Geometric Sketches Better Recapitulates Biological Cell Types

Since the samples produced by our algorithm consist of a more balanced composition of cell types, including rare cell types, we also reasoned that clustering analyses should be able to better distinguish these cell types within a geometric sketch compared to uniform downsampling. To assess this capability, we first clustered the sketches using the standard graph-based Louvain clustering algorithm (Blondel et al., 2008). Then, we transferred cluster labels to the rest of the dataset via *k*-nearest-neighbor classification and assessed the agreement between our unsupervised cluster labels and the biological cell type labels provided by the original studies (Method Details). We quantified the clustering accuracy via balanced adjusted mutual information (BAMI), our proposed metric for evaluating clustering quality when the ground truth clusters are highly imbalanced, which is often the case for scRNA-seq datasets. BAMI balances the terms in adjusted mutual information (Vinh et al., 2010) to equally weight each of the ground truth clusters, preventing rare cell types from having only negligible contribution to the performance metric. We also provide results for adjusted mutual information, without our balancing technique, which are largely consistent with our comparisons based on BAMI (Figure S2C).

On a variety of real scRNA-seq datasets, our algorithm's sketches recapitulate the biological cell types consistently better than uniform sampling (Figure 3D). Although two other data-dependent sampling methods, SRS and *k*-means++, achieve performance comparable to our method in a few cases, only geometric sketching obtains competitive performance across all datasets, suggesting that our method is reasonably robust to different experimental settings. Notably, because our sketches are drawn without replacement, clustering scores can become closer to those of uniform samples as the size of the sketch increases; this may explain the diminishing performance of our method with increasing sketch size on the mixture of 293T cells and Jurkat cells (Figure 3D). Still, we note our substantial advantage even on this dataset using very small sketches that select as low as 2% of the full dataset. Moreover, the overall improvement in clustering consistency could become more pronounced as more fine-grain clusters become available as ground truth in light of the enhanced representation of rare transcriptional states within geometric sketches.

Geometric Sketching Assists in the Discovery of a Rare Population of Inflammatory Macrophages

Because geometric sketching of large datasets highlights rare transcriptional states, certain subpopulations of cells that are difficult to identify when analyzing the full dataset may become discoverable within a geometric sketch. To test this in practice, we analyzed a dataset of 254,941 cells taken from human umbilical cord blood without cell type labels (Method Details). We computed a geometric sketch of 20,000 cells and clustered the sketch via the Louvain community detection algorithm. Among the putative macrophage clusters with elevated expression of macrophage-specific marker genes, including *CD14* and *CD68* (Khazen et al., 2005), we found a comparatively rare cluster of macrophages defined by the marker genes *CD74*, *HLA-DRA*, *B2M*, and *JUNB* (AUROC > 0.90; Method Details) (Figure 4). We hypothesized that this cluster corresponds to *inflammatory* macrophages, since each of its marker genes has been implicated in macrophage activation in response to inflammatory stimuli: *CD74* encodes the receptor for macrophage migration inhibitory factor (MIF) (Leng et al., 2003), a pro-inflammatory signal (Morand et al., 2006; Santos and Morand, 2009); HLA-DR has elevated expression in classically pro-inflammatory M1-macrophages (Helm et al., 2014); increased *B2M* has been demonstrated in murine bone marrow derived macrophages after LPS stimulation (Tanaka et al., 2017); and *JUNB* has been implicated as a key transcriptional modulator of macrophage activation (Fontana et al., 2015) and is upregulated by MIF (Calandra and Roger, 2003). We did not observe major differences in the number of unique genes between this rare cluster and the rest of the macrophages (Figure S3), so these differences in gene expression are most likely not an artifact of variable data sparsity or dropout.

We sought further confirmation of this rare expression signature in macrophages by conducting a separate scRNA-seq study of an *in vitro* model of macrophage inflammation in which human CD14+ monocytes were polarized with GM-CSF to induce an inflammatory response (Method Details). We compared this data to a similar scRNA-seq dataset of macrophages but with M-CSF stimulation (Hie et al., 2019) to induce an anti-inflammatory polarization. Expression of all four marker genes we identified (*CD74*, *HLA-DRA*, *B2M*, and *JUNB*) was significantly elevated in GM-CSF-derived ($n = 354$ cells) macrophages compared to the M-CSF-derived ($n = 1107$ cells) macrophages (one-sided Welch's t-test $P =$ 4e-34 for *CD74*, $P =$ 1e-29 for *HLA-DRA*, $P =$ 3e-46 for *B2M*, and $P =$ 1e-13 for *JUNB*), increasing our confidence in these marker genes as a signature of inflammation. Additional *in vivo* confirmation of these markers, along with more in-depth study of macrophage subpopulations, will help reveal insight into inflammation and ways to modulate inflammatory processes in response to disease.

When we applied the same clustering procedure to either the full dataset or a uniform subsample, the clustering algorithm did not assign a separate cluster to inflammatory macrophages but rather placed all macrophages into a single cluster, likely because of the relative scarcity of this cell type compared to the large cluster of inactive macrophages. These results provide additional evidence that geometric sketches contain a richer variety of transcriptional states and can therefore assist researchers in identifying interesting but rare biological structure.

Geometric Sketching Has Significantly Better Scalability to Large Datasets Than Other Sophisticated Sampling Strategies

Not only does geometric sketching lead to more informative sketches of the single-cell data, it is also dramatically faster than other non-uniform sampling methods, which is imperative since researchers stand to gain the most from sketches of very large datasets. Geometric sketching has an asymptotic runtime that is close to linear in the size of the dataset (Method Details) and, when benchmarked on real datasets, is more than an order of magnitude faster than non-uniform methods and has a negligible dependence on the number of samples specified by the user, unlike k-means++ and SRS (Figure 5A). On our largest dataset of 665,858 cells, our sampling algorithm takes an average of around 5 minutes (Figure 5A); in contrast, k-means++ takes 3 hours and spatial random sampling (SRS) takes 5.5 hours when subsampling 10% of the cells. On a simulated benchmark dataset of 10 million data points (Method Details), geometric sketching subsamples 20,000 cells after an average time of 67 minutes, demonstrating practical scalability to datasets with hundreds of millions of cells (Figure 5A). Although uniform sampling is trivially the most efficient technique since it does not consider any properties of the underlying dataset, our algorithm is both efficient and produces high quality samples that more accurately represent the underlying transcriptomic space as we demonstrated above. Notably, our runtime comparison does not include the standard preprocessing step of (randomized) principal component analysis

(PCA), which we uniformly applied to all methods and whose runtime as well as scalability are comparable to our geometric sketching step (Method Details; Figure S4A).

Geometric Sketching Accelerates scRNA-seq Data Integration

In addition to being efficient by itself, geometric sketching can also accelerate other downstream algorithms for scRNA-seq analysis. One such problem involves integration of multiple scRNA-seq datasets across different batches or conditions (Butler et al., 2018; Haghverdi et al., 2018; Hie et al., 2019; Korsunsky et al., 2018). Here, we consider an approach to accelerating scRNA-seq data integration by applying the integration algorithm only to geometric sketches instead of the full datasets. Then, we use the integrated values of the sketch to learn a nonlinear transformation that is applied to the full dataset to place it on the same integrated landscape (Method Details). Since the integration step is more computationally intensive than the latter interpolation step, our geometric sketch-based integration offers a speedup that becomes especially dramatic when integrating large numbers of cells. Moreover, because geometric sketching better preserves rare transcriptional states, as demonstrated above, rare cell types are also more likely to be integrated during the procedure compared to using sketches from other sampling approaches.

We applied geometric sketch-based acceleration to two existing algorithms, Scanorama (Hie et al., 2019) and Harmony (Korsunsky et al., 2018), for scRNA-seq data integration (Figure 5B). However, we note that our acceleration procedure is agnostic to the underlying integration method and can easily interface with similar algorithms (Butler et al., 2018; Haghverdi et al., 2018). We benchmarked the runtime improvement using geometric sketching on a dataset of 534,253 human immune cells from two different tissues (umbilical cord blood and adult bone marrow). On this data, Scanorama and Harmony require 2.1 and 1.9 hours of computation, respectively, to obtain integrations that remove tissue-specific differences. In contrast, the integration procedure with geometric sketching (which includes finding the geometric sketches, integrating the sketches, and then transforming the full datasets based on the sketches) requires just 8 minutes of computation with either Scanorama or Harmony. Moreover, using geometric sketching-based acceleration has integration performance comparable to the full integration (Figure 5B) and better than sketch-based integration using other sampling strategies (Figure S4B), providing yet another example of how geometric sketching can be used to accelerate other algorithms for large-scale scRNA-seq analysis.

DISCUSSION

Geometric sketching provides an efficient algorithm for obtaining subsamples of large scRNA-seq datasets such that the subsample contains as much of the transcriptional heterogeneity from the original dataset as possible. Our algorithm’s sketches require less bandwidth to transfer and can be more easily shared among researchers. Geometric sketches can be inputted into computationally intensive downstream analysis tools designed for smaller datasets, including those that learn complex low-dimensional embeddings (Ding et al., 2018), 2-dimensional visualization coordinates (Cho et al., 2018; McInnes and Healy, 2018), or that fit complex models for a variety of tasks including pseudo-temporal trajectory analysis (Qiu et al., 2017), rare cell type discovery (Grün et al., 2015; Jiang et al., 2016), gene regulatory network reconstruction (Van Dijk et al., 2018), or robust differential expression analysis (Kharchenko et al., 2014). While our method does not distinguish between transcriptional structure due to biological or technical variation (e.g., batch effects), our sampling algorithm could be applied separately to datasets from different batches and then integrated or batch corrected using other methods (Butler et al., 2018; Hie et al., 2019).

Our work is distinct from but complementary to techniques that aim to find representative summaries of gene expression within clusters of cells (Baran et al., 2018; Iacono et al., 2018; Saunders et al., 2018), which output aggregate expression profiles that are not observed in the original dataset. Applying geometric sketching as a preprocessing step would mostly likely accelerate these complex methods for gene expression aggregation while preserving representation of rare transcriptional states. Moreover, we note that because all of the elements within a geometric sketch correspond to actual observations from the original data, researchers have the flexibility to apply any existing downstream method designed for single-cell RNA-seq datasets, unlike methods that modify the gene expression values.

We note that our algorithm should be used in conjunction with other tools for scRNA-seq quality control. To limit artifacts arising due to dropout and data sparsity, it is common to apply a minimum unique gene cutoff, which we also do in our experiments; filtering steps with a linear time complexity in the size of the dataset are unlikely to be a substantial bottleneck for single-cell methods. Another potential artifact common to droplet-based scRNA-seq experiments are doublets, which, due to their more complex transcriptional signatures, may also be more likely to appear within a geometric sketch. Many methods have been developed for computational doublet detection (DePasquale et al., 2018; Kang et al., 2018; McGinnis et al., 2018; Wolock et al., 2018), which can be applied to the sketch to remove these potential sources of confounding variation. We also note that more advanced quality control methods, including those for normalization (Bacher et al., 2017; Lun et al., 2016; Vallejos et al., 2017), highly variable gene filtering (Yip et al., 2018), and imputation (Van Dijk et al., 2018; Li and Li, 2018; Ronen and Akalin, 2018) can naturally be applied to a geometric sketch before further analysis.

While it is possible for individuals to download large datasets and independently run geometric sketching, we envision laboratories that generate large-scale single-cell omics datasets would also compute and provide geometric sketches alongside the full data. These sketches would then be available to download for users with more limited computational resources or those wishing to run quick exploratory analyses on a subset of the data. In this spirit, we have computed small geometric sketches of a number of large, publicly-available scRNA-seq datasets containing hundreds of thousands of cells or even millions of cells, which are available for download at http://geosketch.csail.mit.edu. We also provide implementations of geometric sketching and the other sampling algorithms used in our benchmarking experiments in the geosketch Python package (https://github.com/brianhie/geosketch). Finally, we note that our techniques can be applied beyond single-cell transcriptomics, or even biological datasets, to any setting in which compact, geometric summaries of the data would prove useful.

STAR METHODS

CONTACT FOR REAGENT AND RESOURCE SHARING

Further information and requests for resources and reagents should be directed to and will be fulfilled by the Lead Contact, Bonnie Berger (bab@mit.edu).

EXPERIMENTAL MODEL AND SUBJECT DETAILS

Macrophages—Human monocytes were isolated from human buffy coats purchased from the Massachusetts General Hospital blood bank using a standard Ficoll gradient and subsequent CD14+ cell positive selection (Stemcell Technologies). Selected monocytes were cultured in ultra low-adherence flasks (Corning) for 6 days with RPMI media (Invitrogen) supplemented with 10% FBS (Invitrogen) and 50 ng/mL human GM-CSF (Biolegend) before profiling with single-cell RNA-sequencing.

METHOD DETAILS

Geometric Sketching Problem—We first give a mathematical formulation of the sketching problem to elucidate the theoretical insights underlying our approach. Let $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be a representation of a single-cell dataset, consisting of m-dimensional measurements $\mathbf{x}_i \in \mathbb{R}^m$ from n individual cells. In the case of very large n (e.g., millions of cells) (Macosko et al., 2015; Zheng et al., 2017), it is often desirable to construct a *sketch* $\mathcal{S} \subset \mathcal{X}$ (i.e., a downsampled dataset), which can be more easily shared with other researchers and be used to quickly understand the salient characteristics of $\mathcal{X}$ without paying the full computational price of analyzing $\mathcal{X}$.

Drawing insight from computational geometry, we measure the quality of a sketch $\mathcal{S}$ with respect to a dataset $\mathcal{X}$ via the *Hausdorff distance* d_H (Hausdorff, 1937) defined as

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

where d denotes the distance function of the underlying metric space (i.e., a notion of dissimilarity between two cells). Intuitively, d_H measures the distance of the cell in the original dataset that is farthest away from any of the cells included in the sketch. The lower this distance, the more comprehensively our sketch covers the original dataset.

We are interested in developing an efficient algorithm for obtaining $\mathcal{S}$ of a predetermined size k (i.e., $|S| = k$) that minimizes $d_H(\mathcal{X}, \mathcal{S})$. A key property of our approach is that it is *agnostic to local density of data points*, since only the maximum distance is taken into account. As a result, our sketches more evenly cover the space of gene expression spanned by the original dataset. In contrast, approaches based on uniform sampling or distance-based sampling [e.g., k-means++ (Arthur and Vassilvitskii, 2007)] are biased toward selecting more cells in densely populated regions at the expense of other regions of interest with fewer data points, as we demonstrate in our experiments.

Theoretical Connection to Covering Problems—Our problem of finding a high-quality sketch $\mathscr{S}$ of size *k* that minimizes $d_H(\mathscr{X}, \mathscr{S})$ is closely related to the concept of covering numbers in information theory and combinatorics. Informally, *internal covering number* is defined as the smallest number of equal-sized shapes (e.g., spheres or boxes) centered at individual data points that, together, "cover" all points in a dataset. To relate our covering to the Hausdorff distance, we provide the following lemma:

Lemma 1: Let $\mathscr{X} = \left\{\mathbf{x}_1, \ldots, \mathbf{x}_n\right\}$ be a representation of a single-cell dataset, consisting of *m*-dimensional measurements $\mathbf{x}_i \in \mathbb{R}^m$ from *n* individual cells. Let d_H^* be the minimal Hausdorff distance $d_H(\mathscr{X}, \mathscr{S})$ obtained by a sketch $\mathscr{S} \subset \mathscr{X}$ where $|\mathscr{S}| = k$. Then, $d_H^* = N_{\text{int}}^{-1}(k)$, where $N_{\text{int}}^{-1}(k) := \min\left\{r : N_{\text{int}}(\mathscr{X}, r) \le k\right\}$.

Proof of Lemma 1: Since d_H bounds the maximum distance of a data point from $\mathscr{S}$, placing a sphere of radius d_H at every point in $\mathscr{S}$ gives a covering of $\mathscr{X}$, which implies $N_{\text{int}}\left(\mathscr{X}, d_H^*\right) \le k$. Thus, $N_{\text{int}}^{-1}(k) \le d_H^*$. If $N_{\text{int}}^{-1}(k) < d_H^*$, then there exists a cover with *k* spheres of radius $d' < d_H^*$ Taking the center points of this cover as our sketch $\mathscr{S}'$, we obtain $d_H(\mathscr{X}, \mathscr{S}') \le d_H^*$, a contradiction. Hence, $d_H^* = N_{\text{int}}^{-1}(k)$.■

Lemma 1 shows that the *minimum radius* for covering spheres that gives an internal covering number of at most *k* on a given dataset is in fact equal to the optimal Hausdorff distance achievable by a sketch of size *k*. An important insight given by this observation is that the problem of finding a high-quality sketch reduces to finding a minimum-cardinality cover of a dataset given a certain radius. In particular, if one were to have access to an oracle that could find the optimal covering of a dataset for any radius, our problem could be solved by finding the minimum radius that gives the desired number of covering spheres (e.g., via binary search). Unfortunately, finding the minimum-cardinality cover is NP-complete (Attali et al., 2016), and although algorithms for a variety of simplified settings have been studied (Ahn et al., 2011; Al et al., 2006; Chan and Hu, 2015; Chvatal, 1979), none scales to the high-dimensional and large-scale data that we need to handle in single-cell genomics. Given the hardness of the covering problem, we aimed to devise an approximate covering algorithm that readily scales to large-scale single-cell data while maintaining good sketch quality.

Our Geometric Sketching Algorithm—At the core of our geometric sketching algorithm is a *plaid covering*, which approximates the geometry of the given single-cell data as a union of equal-sized boxes. To enable scalability to large datasets, we restricted our attention to covering the data points with a simple class of covering sets—plaids—whose structure is amenable to fast computation. Formally, we define a length-ℓ *plaid cover*) $\mathscr{C}$ of a dataset $\mathscr{X}$ as a collection of points $\mathbf{c}_1, \ldots, \mathbf{c}_k \in \mathbb{R}^m$ such that:

i. Either $c_{ij} = c_{i'j}$ or $\left|c_{ij} - c_{i'j}\right| \geq \ell$ for all $i, i' \in [j]$ and $j \in [m]$, and

ii. $\mathscr{X} \subset \cup_{i=1}^{k} R\left(\mathbf{c}_i, \ell\right)$, where $R\left(\mathbf{c}_i, \ell\right) = \left[c_{i1}, c_{i1} + \ell\right] \times \cdots \times \left[c_{im}, c_{im} + \ell\right]$.

Intuitively, $\mathscr{C}$ represents a collection of m-dimensional square boxes of side length ℓ covering $\mathscr{X}$ that can be generated by placing a grid (with potentially uneven intervals) over the space and selecting a subset of grid cells. An example plaid cover is illustrated in Figure 1. Our greedy algorithm for finding a plaid cover of a given dataset is shown in Algorithm 1. To see that the plaid cover found by our algorithm uses the smallest number of intervals in each coordinate (although it may not achieve the smallest cardinality overall) consider the following proof:

Proof that Algorithm 1 is optimal in each dimension separately: Fix a dimension $d \in [n]$, and consider covering the projection $\pi_d(\mathscr{X}) = \left\{x_{1d}, x_{2d}, \ldots, x_{nd}\right\} \subset \mathbb{R}$ with a one-dimensional plaid cover of length ℓ. Let $Q = \{q_1, \ldots, q_k\}$ be any such cover, and let $Y = \{y_1, \ldots, y_m\}$ denote the cover produced by our algorithm on iteration d. We show that $k \geq m$, i.e., Y has the smallest size of any length-ℓ cover.

Assume without loss of generality that $q_1 < q_2 < \ldots < q_k$ and $y_1 < y_2 < \ldots < y_m$. Let z_i denote the i^{th}-smallest element of $\pi_d(\mathscr{X})$. Our algorithm sets $y_1 = z_1$. We must have $q_1 \leq z_1$, or else z_1 is not covered by Q. Thus, $q_1 \leq y_1$. Proceeding inductively, we see that:

$$q_{i+1} \leq \min\left\{z_i : z_i > q_i + \ell\right\} \leq \min\left\{z_i : z_i > y_i + \ell\right\} = y_{i+1}$$

where the final equality holds because our algorithm defines y_{i+1} exactly this way. Thus, we have $q_i \leq y_i$ for all $i \in 1, 2, \ldots, \min(k, m)$. If $|Q| \leq |Y|$, then y_{m-1} and y_m are both greater than all elements in Q. But because Q covers all the points z_i, this implies that y_m covers no points, a contradiction because our algorithm does not construct empty covering sets. Thus, we must have $|Q| \geq |Y|$, and because Q is arbitrary, Y has the smallest possible size. ■

The main intuition behind our choice of plaid pattern is that it generalizes grid-based approximation of geometric shapes while maintaining computational efficiency in assigning points to their respective covering box. Note our plaid covering algorithm has time complexity in each dimension of $O(n \log n)$ in general—the main bottleneck being the sorting of each coordinate—and uses $O(n)$ space. In practical scenarios where each coordinate requires only a small constant number of intervals to cover, we achieve $O(n)$ time complexity by taking linear scans to find the next interval without sorting. This is a substantial improvement over other approaches for tackling the covering problem, which typically require $O(n^2)$ time for all pairwise distance calculations. A greedy approach to building a cover could require only $O(kn)$ pairwise distance calculations where k is the number of covering objects (Chvatal, 1979), yet k is still typically much larger than $\log n$ for our applications in single-cell analysis.

Algorithm 1:

Greedy Plaid Cover

```
Data: Dataset $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ where $\mathbf{x}_i \in \mathbb{R}^m$, length $\ell$
Result: Length-$\ell$ plaid cover $\mathcal{C}$ of $\mathcal{X}$
$\mathbf{y}_i \leftarrow 0 \in \mathbb{R}^m, \forall i \in [n]$
for $j \in [m]$ do
    $z_1, \ldots, z_n \leftarrow$ Sort($\{x_{1j}, \ldots, x_{nj}\}$) /* In ascending order. */
    $p \leftarrow 1$
    while $z_p + \ell < z_n$ do
        Find smallest $i > p$ where $z_p + \ell < z_i$
        $y_{i'j} \leftarrow z_p, \forall i' \in \{p, \ldots, i-1\}$
        $p \leftarrow i$
    end
    $y_{i'j} \leftarrow z_p, \forall i' \in \{p, \ldots, n\}$
end
return $\{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$ /* Only unique points are returned. */
```

The cardinality of the cover returned by our plaid cover algorithm generally decreases as the length parameter ℓ increases, although pathological cases that deviate from this pattern exist. We empirically confirmed the near-monotonic relationship between number of covering boxes and ℓ on all our single-cell benchmark datasets (Figure S5A). Based on this observation, we perform binary search (with graceful handling of potential exceptions) to find the value of ℓ that approximately produces a desired number of covering boxes. By default, we choose the same number of boxes as the desired sketch size k. A sketch is then constructed by sampling the boxes in a plaid cover and choosing a point at random from each box. The quality of our sketch is given by the following theorem:

<u>**Theorem 1:**</u> Given a dataset $\mathcal{X}$ of n points in m dimensions, let $N_{\text{plaid}}(\ell)$ be the number of boxes in the plaid cover returned by our algorithm as a function of length parameter ℓ. Let $N^{-1}_{\text{plaid}}(k) = \inf\left\{\ell : N_{\text{plaid}}(\ell) \leq k\right\}$. Let k be a desired sketch size and assume $k = N_{\text{plaid}}\left(N^{-1}_{\text{plaid}}(k)\right)$ for simplicity (if not take a nearby k where this holds). Let $\mathcal{S}_{\text{plaid}}(k)$ be a sketch of size k obtained by randomly choosing appoint from each box in the plaid cover. Let $d^*_H(k) = \min_{S : |S| = k} d_H(\mathcal{X}, \mathcal{S})$. Then, the following holds:

$$\frac{1}{2}N^{-1}_{\text{plaid}}\left(2^m \cdot k\right) \leq d^*_H(k) \leq d_H\left(\mathcal{X}, \mathcal{S}_{\text{plaid}}(k)\right).$$

Proof of Theorem 1: For the first inequality, Let $\mathscr{P} = P_1, P_2, \ldots, P_N$ be any covering by plaid sets of side length $2d_H^*(k)$, such that all covering sets contain at least one point. We show that $\mathscr{P}$ has cardinality at most 2^m k.

Let $\mathscr{B}$ be a covering of $\mathscr{X}$ by k balls $B_1, B_2, \ldots, B_k$, each with radius $d_H^*(k)$. The definition of d_H^* ensures that such a covering exists. Define

$$I_{\mathscr{P}}\left(B_i\right) = \left|\left\{P_j : P_j \cap B_i \neq \varnothing\right\}\right|.$$

That is, $I_{\mathscr{P}}\left(B_i\right)$ is the number of sets in $\mathscr{P}$ that intersect B_i.

Because $\mathscr{P}$ and $\mathscr{B}$ are both covering sets, each plaid square in $\mathscr{P}$ is intersected by at least one ball in $\mathscr{B}$. Therefore,

$$|\mathscr{P}| \leq \sum_{i=1}^{k} I_{\mathscr{P}}\left(B_i\right).$$

On the other hand, we see that $I_{\mathscr{P}}\left(B_i\right)$ is bounded above by 2^m, because any ball overlaps at most two plaid intervals in each dimension. Thus,

$$\left|\mathscr{P}\right| \leq 2^m k$$

as desired. The second inequality is immediate, because $d_H^*(k)$ is an infimum of Hausdorff distances of all sets of size k with $\mathscr{X}$, and $\mathscr{S}_{\text{plaid}}(k)$ is such a set. ■

Theorem 1 provides a theoretical insight into the quality of a sketch obtained via plaid covering. In particular, it gives a bound on the optimal Hausdorff distance relative to the solution obtained by plaid covering.

In order to reduce the dimensionality of the problem for scalability as well as robustness to noise, we first project the data down to a relatively low-dimensional space (100 dimensions for single-cell data) using a fast random projection-based PCA (Halko et al., 2011) before applying our plaid covering algorithm. We note that much work has been done in obtaining algorithms for computing an approximate PCA of very large datasets with provable bounds on approximation error that are also highly efficient in runtime and memory (Halko et al., 2011; Ross et al., 2008); obtaining the top 100 principal components (PCs) of our largest benchmark dataset with 665,858 cells requires about 10 minutes of additional computation time with linear-time scalability in the size of the dataset (Figure S4A).

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 ($\lvert\mathscr{C}\rvert$)	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.

Baseline Sampling Methods—We benchmark our algorithm against a number of existing sampling methods:

i. *Uniform sampling* returns a random sample of the cells, where every cell is given equal probability. We use the random choice function provided by the numpy Python package (Oliphant, 2006).

ii. *Spatial random sampling* (SRS) (Rahmani and Atia, 2017b) first projects the data points onto the unit hypersphere, then each sample is obtained by uniformly sampling a point on the unit hypersphere and selecting the closest point in the projected dataset according to the cosine distance.

iii. *k-means++ sampling* (Arthur and Vassilvitskii, 2007) randomly chooses an initial sample, then repeatedly samples the next point by giving each point a weight proportional to the minimum distance from previous samples. This procedure continues until the desired number of samples have been obtained. We used the *k*-means++ implementation from the scikit-learn package (Pedregosa and Varoquaux, 2011).

We also run our experiments for SRS and *k*-means++ sampling using the same lower dimensional embeddings (top 100 PCs) used as input to geometric sketching.

Benchmark Datasets—We used the following datasets for our benchmarking experiments:

i. *293T and Jurkat mixture*. We obtained a mixture of 293T cells and Jurkat cells from 10X Genomics (Zheng et al., 2017) containing a much smaller number of 293T cells than Jurkat cells, where cell types are computationally inferred based on consensus clustering and marker genes. We removed cells below a cutoff of 500 unique genes, normalized each cell by the total expression and reduced the dimensionality to 100 PCs. The resulting data contained 4,185 cells in total.

ii. *Peripheral blood mono-nuclear cells (PBMCs)*. We obtained a dataset of PBMCs from 10X Genomics (Zheng et al., 2017) and used the computationally curated cell type labels as well as the cell quality filtering steps from the original study. We then normalized each cell by the total expression and reduced the dimensionality to 100 PCs. The resulting data contained 68,579 cells.

iii. *Adult mouse brain*. We obtained scRNA-seq data from different regions of the mouse brain from Saunders *et al.* (Saunders et al., 2018) and used the computationally curated cell type labels as well as the cell quality filtering steps from the original study, including removal of doublet and outlier cells. We then normalized each cell by the total expression and reduced the dimensionality to 100 PCs. The resulting data contained 665,858 cells.

iv. *Developing and adolescent mouse central nervous system (CNS)*. We obtained scRNA-seq data from different regions of the mouse CNS from Zeisel *et al.* (Zeisel et al., 2018), removed cells below a cutoff of 500 unique genes, and used the computationally curated cell type labels and additional cell quality filtering steps from the original study. We then normalized each cell by the total expression and reduced the dimensionality to 100 PCs. The resulting data contained 465,281 cells.

Robust Hausdorff Distance Computation—The classical Hausdorff distance (HD) (Hausdorff, 1937), according to our problem formulation, is computed as $d_H(\mathcal{X}, \mathcal{S}) = \max_{\mathbf{x} \in \mathcal{X}}\left\{\min_{\mathbf{s} \in \mathcal{S}} d(\mathbf{x}, \mathbf{s})\right\}$ where $\mathcal{X}$ denotes the point set corresponding to the full scRNA-seq dataset and $\mathcal{S}$ denotes the point set corresponding to a sketch, where $\mathcal{S} \subseteq \mathcal{X}$. Because the classical HD measure is highly sensitive to even a few number of outliers (Huttenlocher et al., 1993; Sim et al., 1999), we use a robust HD measure proposed by Huttenlocher et al. called the partial HD measure, defined $d_{HK}(\mathcal{X}, \mathcal{S}) = K^{\text{th}}_{\mathbf{x} \in \mathcal{X}}\left\{\min_{\mathbf{s} \in \mathcal{S}} d(\mathbf{x}, \mathbf{s})\right\}$ where $K^{\text{th}}_{\mathbf{x} \in \mathcal{X}}$ denotes the K^{th} largest value; partial HD requires a parameter $q = 1 - K/|\mathcal{X}|$ between 0 and 1, inclusive, which is equivalent to classical HD when $q = 0$ (Huttenlocher et al., 1993). We set $q = 1\text{e-}4$, which obtains a measurement that is very close to the value obtained by classical HD but is robust to the most extreme outliers. We achieved similar results for different values of q (Figure S5C).

Data Visualization—To visualize the subsampled data based on different sampling methods, we used a 2-dimensional *t*-distributed stochastic neighbor embedding (*t*-SNE) with a perplexity of 500, a learning rate of 200, and 500 training iterations. We used the implementation provided by the Multicore-TSNE Python package (https://github.com/DmitryUlyanov/Multicore-TSNE).

Simulation Analysis of Data with Known Volume and Density—To obtain datasets for which the volume (transcriptional diversity) and density of each cell type is known *a priori*, we considered different ways to duplicate and transform a dataset composed entirely of 293T cells (Zheng et al., 2017). To obtain a dataset with clusters of equal volume and variable density, we uniformly subsampled the 293T cells by a factor of 10 or 100 to create two new clusters, where each new cluster is translated such that none of the clusters overlap. Likewise, to obtain a dataset with clusters of equal number of data points but variable volume, we projected down to 3-dimensions using PCA and rescaled the components by a factor of 10 or 100 to create two new clusters, which are similarly translated to avoid overlap. Using a lower dimensionality in our simulations allowed us to better reason about the expected change in volume and is close to the effective fractal dimension of the dataset

(Figure S5B). We then sketched these datasets and assessed the density-dependence by computing the Kullback-Leibler (KL) divergence $\sum_{i=1}^{k} p_i \log \frac{c_i}{np_i}$ where p_i denotes the normalized volume of cell type i such that $\sum_i p_i = 1$, c_i denotes the number of cells in cluster i, k is the number of clusters, and n is the total number of cells in the dataset. Lower values of the KL divergence indicate a sampling that better reflects the volume of each of the clusters.

Differential Entropy of Cell Types—To obtain a rough estimate of the transcriptional variability represented by each cell type i, we fit a multivariate Gaussian distribution to each cell type to obtain an estimate of the covariance $\widehat{\boldsymbol{\Sigma}}_i$; we then computed the differential entropy $\frac{m}{2} + \frac{m}{2}\ln(2\pi) + \frac{1}{2}\ln\left(\left|\widehat{\boldsymbol{\Sigma}}_i\right|\right)$ where m is the dimensionality of the data. We fit the distribution using the GaussianMixture class from scikit-learn (Pedregosa and Varoquaux, 2011).

Clustering Analysis—We quantify the ability for clustering analyses on a subsample of a full dataset to recapitulate a set of “ground truth” labels, in this case, the cell type labels assigned by the original study authors.

For the Louvain clustering analysis, we constructed the nearest neighbors graph on which we applied the Louvain community detection algorithm (Blondel et al., 2008). We use the graph construction and Louvain implementation with default parameters provided by scanpy (Wolf et al., 2018), which leverages the louvain-igraph package (https://github.com/vtraag/louvainigraph). Louvain cluster labels were applied to the full dataset based on the most common label of the five nearest neighbors within the sketch (ties broken randomly). We quantified agreement between the unsupervised cluster labels and the previous study labels using the adjusted mutual information (AMI) score (Vinh et al., 2010) implemented by the scikit-learn Python package (Pedregosa and Varoquaux, 2011) based on a resampled dataset where the relative frequencies of the ground truth clusters are set to uniform to equally consider the clusters regardless of their abundance in the full dataset. We refer to this metric as balanced AMI (BAMI). The correction factor in AMI for chance agreement is updated accordingly to account for the balanced distribution. We repeat the analysis using three different Louvain resolution parameters (0.5, 1, and 2) and take the maximum BAMI score across these parameter settings for each sampling algorithm.

Immune Cell Analysis—254,941 cells from umbilical cord blood were obtained from the Human Cell Atlas (https://preview.data.humancellatlas.org). The dataset was filtered for cells containing more than 500 unique genes, normalized by the total expression for each cell, and then natural log transformed after adding a pseudo-count of 1. Data was projected to 100 PCs using the randomized PCA implementation provided by the fbpca Python package (https://github.com/facebook/fbpca). Unsupervised clustering was performed by running the Louvain community detection algorithm with the default parameters (resolution of 1, 15-nearest neighbors graph) of the scanpy framework (Wolf et al., 2018).

Macrophage Polarization scRNA-seq Experiment and Analysis—scRNA-seq data of M-CSF-derived macrophages was obtained from the study by Hie *et al.* (2019). We repeated the same experiment but instead polarized with an inflammatory stimulus, GM-CSF. Human monocytes were isolated and polarized as described above. SeqWell analysis was performed as previously described (Gierahn et al., 2017). Briefly, after 6 days, cells were detached using trypsin, spun down, and counted. Approximately 12,000 cells were loaded on each array for each timepoint and condition to minimize doublet-loading. The arrays were sealed with a semi-permeable membrane prior to cell lysis and hybridization to single-cell beads. Beads were subsequently pooled for reverse transcription and whole transcriptome amplification.

Read alignment and transcript quantification were performed as in Macosko *et al.* (2015b). Briefly, raw sequencing data was converted to demultiplexed FASTQ files using bcl2fastq2 based on Nextera N700 indices corresponding to individual samples/arrays. Reads were then aligned to the hg19 genome using the Galaxy portal maintained by the Broad Institute for Drop-Seq alignment using standard settings. Individual reads were tagged according to the 12-bp barcode sequence and the 8-bp UMI contained in Read 1 of each fragment. Following alignment, reads were binned onto 12-bp cell barcodes and collapsed by their 8-bp UMI. Digital gene expression matrices for each sample were obtained from quality filtered and mapped reads, with an automatically determined threshold for cell count. Analysis was done on cells that were filtered for a minimum cutoff of 500 unique genes, normalized by the total expression of each cell, and then natural log transformed after adding a pseudo-count of 1.

Runtime Benchmarking—We benchmarked the runtime of geometric sketching to obtain 20,000 cells from a dataset of 1, 2.5, 5, or 10 million cells, where we obtained each full dataset by resampling the cells from the mouse CNS dataset (Zeisel et al., 2018) to reach the desired cell count. We timed the algorithm using Python's time module. All experiments were done on a 2.30 GHz Intel Xeon E5–2650v3 CPU.

Geometric Sketching-Accelerated Integration—We assume an integration function that takes in a list of datasets and returns modifications to the datasets that removes differences to due batch effect etc. Let $\mathbf{X} \in \mathbb{R}^{n \times m}$ denote one of the datasets, $\mathbf{X}_{\mathcal{S}} \in \mathbb{R}^{|\mathcal{S}| \times m}$ denote the subset of $\mathbf{X}$ obtained by geometric sketching, and $\mathbf{X}'_{\mathcal{S}} \in \mathbb{R}^{|\mathcal{S}| \times m}$ denote the modified version of $\mathbf{X}_{\mathcal{S}}$ returned by the integration function. Our goal is to apply a transformation to $\mathbf{X}$ that puts it into the same integrated space as $\mathbf{X}'_{\mathcal{S}}$. At a high level, we use a nearest-neighbors-based method to compute alignment vectors from $\mathbf{X}$ to $\mathbf{X}_{\mathcal{S}}$, we use Gaussian smoothing to combine these alignment vectors into translation vectors, and then we apply thetranslation to $\mathbf{X}$ to obtain an "integrated" full dataset $\mathbf{X}'$.

Formally, for each cell in $\mathbf{X}_{\mathcal{S}}$, we find its k nearest neighbors in $\mathbf{X}$ and we denote the set of all matches between a cell in $\mathbf{X}_{\mathcal{S}}$ and $\mathbf{X}$ as $\mathcal{M}$ where $|\mathcal{M}| = k\left|\mathbf{X}_{\mathcal{S}}\right|$. Now we define the alignment vectors as the rows of the matrix $\mathbf{X}^{(\text{match})} - \mathbf{X}_{\mathcal{S}}^{(\text{match})}$ where the rows of $\mathbf{X}^{(\text{match})}, \mathbf{X}_{\mathcal{S}}^{(\text{match})} \in \mathbb{R}^{|\mathcal{M}| \times m}$ correspond to the pairs of matching cells in $\mathcal{M}$. We want to combine these alignment vectors to obtain our translation vectors, which we do using Gaussian smoothing. We compute weights via a Gaussian kernel as

$$[\mathbf{\Gamma}]_{a,b} = \exp\left(-\frac{\sigma}{2}\left\|[\mathbf{X}]_{a,:} - \left[\mathbf{X}^{(\text{match})}\right]_{b,:}\right\|_2^2\right)$$

where $\mathbf{\Gamma} \in \mathbb{R}^{n \times |M|}$ and $[\cdot]_{a,b}$ denotes the value in the a^{th} row and b^{th} column of a matrix and $[\cdot]_{a,:}$ denotes the a^{th} row of a matrix. Finally, we construct the translation vectors as an average of the alignment vectors with Gaussian-smoothed weights, where

$$\mathbf{v}_a = \frac{[\mathbf{\Gamma}]_{a,:}\left(\mathbf{X}^{(\text{match})} - \mathbf{X}_{\mathcal{S}}^{(\text{match})}\right)}{\sum_{b \in [|\mathscr{M}|]} [\mathbf{\Gamma}]_{a,b}}$$

and we translate

$$[\mathbf{X}']_{a,:} = [\mathbf{X}]_{a,:} + \mathbf{v}_a$$

for all $a \in [n]$ where $[n]$ denotes the set of all natural numbers up to n. We repeat this for all datasets integrated by the “black-box” integration function; in our study, we used the Scanorama (Hie et al., 2019) and Harmony (Korsunsky et al., 2018) algorithms for integration.

We use geometric sketches of size 4000 (around 1% of the total data) and parameters $k = 3$ and $\sigma = 15$. We used Harmony version 0.0.0.9000 and Scanorama version 1.0. For all methods, we measured the runtime required for integration and translation, not including the initial PCA step for computing low dimensional embeddings (100 PCs). We quantify dataset mixing by clustering the integrated embeddings using k-means, varying the number of clusters, and computing the average negative Shannon entropy normalized to a maximum value of 1 on the dataset labels averaged across all clusters, an approach taken by recent work (Park et al., 2018).

QUANTIFICATION AND STATISTICAL ANALYSIS

Immune Cell Analysis—Marker genes for inflammation were selected using a nominal AUROC cutoff of 0.9 for separation of the inflammatory cluster from the remaining clusters of macrophages. Validation of marker genes in GM-CSF-versus M-CSF-polarized macrophages using a one-sided Welch’s *t*-test (for unequal population sizes) using the scipy Python package (Oliphant, 2007).

DATA AND SOFTWARE AVAILABILITY

Our code and data (including the above datasets) are available at http://geosketch.csail.mit.edu and at https://github.com/brianhie/geosketch.

ADDITIONAL RESOURCES

We provide precomputed sketches of large-scale, publicly-available benchmark scRNA-seq datasets at http://geosketch.csail.mit.edu.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

ACKNOWLEDGEMENTS

We thank Y.W. Yu for discussion and feedback. B. Hie and H. Cho are partially supported by NIH grant R01 GM081871 (to B. Berger). A preliminary version of this manuscript has been accepted for presentation at the 2019 Annual International Conference on Research in Computational Molecular Biology (RECOMB).

GLOSSARY

Sketch
A smaller subset of elements from a larger dataset. Typically used to accelerate a given analysis while preserving the accuracy of the analysis results.

Transcriptomic space
A multidimensional space in which the location of a point (i.e., cell) within this space is determined by gene expression.

Cover, covering
In the geometric sketching setting, a set of shapes in the transcriptomic space that collectively contains all of the cells in a dataset.

Hypercube
A generalization of a cube (with equal side lengths) to many dimensions.

Clustering
Methods that assign cells to groups, or “clusters,” based on some notion of similarity, where more similar cells are assigned to the same cluster.

REFERENCES

10x Genomics (2017). Transcriptional Profiling of 1.3 Million Brain Cells with the Chromium Single Cell 3’ Solution.

Ahn HK, Bae SW, Demaine ED, Demaine ML, Kim SS, Korman M, Reinbacher I, and Son W (2011). Covering points by disjoint boxes with outliers. Comput. Geom. Theory Appl.

Alt H, Arkin EM, Brönnimann H, Erickson J, Fekete SP, Knauer C, Lenchner J, Mitchell JSB, and Whittlesey K (2006). Minimum-cost coverage of point sets by disks. Proc. Twenty-Second Annu. Symp. Comput. Geom. - SCG ‘06.

Angerer P, Simon L, Tritschler S, Wolf FA, Fischer D, and Theis FJ (2017). Single cells make big data: New challenges and opportunities in transcriptomics. Curr. Opin. Syst. Biol

Arthur D, and Vassilvitskii S (2007). K-Means++: the Advantages of Careful Seeding. Proc ACM-SIAM Symp. Discret. Algorithms

Attali D, Nguyen T-B, and Sivignon I (2016). Epsilon-covering is NP-complete. In European Workshop on Computational Geometry (EuroCG), p.

Bacher R, Chu LF, Leng N, Gasch AP, Thomson JA, Stewart RM, Newton M, and Kendziorski C (2017). SCnorm: Robust normalization of single-cell RNA-seq data. Nat. Methods

Baran Y, Sebe-pedros A, Lubling Y, Giladi A, Chomsky E, and Meir Z (2018). MetaCell: analysis of single cell RNA-seq data using k-NN graph partitions. BioRxiv.

Blondel VD, Guillaume JL, Lambiotte R, and Lefebvre E (2008). Fast unfolding of communities in large networks. J. Stat. Mech. Theory Exp.

Butler A, Hoffman P, Smibert P, Papalexi E, and Satija R (2018). Integrating single-cell transcriptomic data across different conditions, technologies, and species. Nat. Biotechnol 4096.

Calandra T, and Roger T (2003). Macrophage migration inhibitory factor: A regulator of innate immunity. Nat. Rev. Immunol

Chan TM, and Hu N (2015). Geometric red-blue set cover for unit squares and related problems. Comput. Geom. Theory Appl.

Cho H, Berger B, and Peng J (2018). Generalizable and Scalable Visualization of Single-Cell Data Using Neural Networks. Cell Syst.

Chvatal V (1979). A Greedy Heuristic for the Set-Covering Problem. Math. Oper. Res

DePasquale EAK, Schnell DJ, Valiente I, Blaxall BC, Grimes HL, Singh H, and Salomonis N (2018). DoubletDecon: Cell-State Aware Removal of Single-Cell RNA-Seq Doublets. BioRxiv.

Van Dijk D, Sharma R, Nainys J, Wolf G, Krishnaswamy S, Pe'er Correspondence D, and Gene GA (2018). Recovering Gene Interactions from Single-Cell Data Using Data Diffusion In Brief Population Analysis Archetypal Analysis Gene Interactions. Cell 174, 716–729.e27. [PubMed: 29961576]

Ding J, Condon A, and Shah SP (2018). Interpretable dimensionality reduction of single cell transcriptome data with deep generative models. Nat. Commun

Fontana MF, Baccarella A, Pancholi N, Pufall MA, Herbert DR, and Kim CC (2015). JUNB Is a Key Transcriptional Modulator of Macrophage Activation. J. Immunol

Gierahn TM, Wadsworth MH, Hughes TK, Bryson BD, Butler A, Satija R, Fortune S, Christopher Love J, and Shalek AK (2017). Seq-Well: Portable, low-cost RNA sequencing of single cells at high throughput. Nat. Methods 14, 395–398. [PubMed: 28192419]

Grün D, Lyubimova A, Kester L, Wiebrands K, Basak O, Sasaki N, Clevers H, and Van Oudenaarden A (2015). Single-cell messenger RNA sequencing reveals rare intestinal cell types. Nature.

Haghverdi L, Lun A, Morgan M, and Marioni J (2018). Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. Nat. Biotechnol 4091.

Halko N, Martinsson P-G, and Tropp J (2011). Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions. SIAM Rev. 53, 217–288.

Hausdorff F (1937). Set Theory.

Helm O, Held-Feindt J, Schäfer H, and Sebens S (2014). M1 and M2: There is no "good" and "bad"-How macrophages promote malignancy-associated features in tumorigenesis. Oncoimmunology.

Hie B, Bryson B, and Berger B (2019). Efficient integration of heterogeneous single-cell transcriptomes using Scanorama. Nat. Biotechnol In press.

Huttenlocher DP, Klanderman GA, and Rucklidge WJ (1993). Comparing Images Using the Hausdorff Distance. IEEE Trans. Pattern Anal. Mach. Intell

Iacono G, Mereu E, Guillaumet-Adkins A, Corominas R, Cusco I, Rodríguez-Esteban G, Gut M, Pérez-Jurado LA, Gut I, and Heyn H (2018). Bigscale: An analytical framework for big-scale single-cell data. Genome Res.

Jiang L, Chen H, Pinello L, and Yuan GC (2016). GiniClust: Detecting rare cell types from single-cell gene expression data with Gini index. Genome Biol.

Kang HM, Subramaniam M, Targ S, Nguyen M, Maliskova L, McCarthy E, Wan E, Wong S, Byrnes L, Lanata CM, et al. (2018). Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. Nat. Biotechnol 36, 89–94. [PubMed: 29227470]

Kharchenko PV, Silberstein L, and Scadden DT (2014). Bayesian approach to single-cell differential expression analysis. Nat. Methods

Khazen W, M'Bika JP, Tomkiewicz C, Benelli C, Chany C, Achour A, and Forest C (2005). Expression of macrophage-selective markers in human and rodent adipocytes. FEBS Lett.

Korsunsky I, Fan J, Slowikowski K, Zhang F, Wei K, Baglaenko Y, Brenner M, Loh P-R, and Raychaudhuri S (2018). Fast, sensitive, and accurate integration of single cell data with Harmony. BioRxiv

Leng L, Metz CN, Fang Y, Xu J, Donnelly S, Baugh J, Delohery T, Chen Y, Mitchell RA, and Bucala R (2003). MIF Signal Transduction Initiated by Binding to CD74. J. Exp. Med

Li WV, and Li JJ (2018). An accurate and robust imputation method scImpute for single-cell RNA-seq data. Nat. Commun

Lun ATL, Bach K, and Marioni JC (2016). Pooling across cells to normalize single-cell RNA sequencing data with many zero counts. Genome Biol.

Van Der Maaten LJP, and Hinton GE (2008). Visualizing high-dimensional data using t-SNE. J. Mach. Learn. Res 9, 2579–2605.

Macosko EZ, Basu A, Satija R, Nemesh J, Shekhar K, Goldman M, Tirosh I, Bialas AR, Kamitaki N, Martersteck EM, et al. (2015). Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. Cell.

McGinnis CS, Murrow LM, and Gartner ZJ (2018). DoubletFinder: Doublet detection in single-cell RNA sequencing data using artificial nearest neighbors. BioRxiv.

McInnes L, and Healy J (2018). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. ArXiv 1802.03426.

Morand EF, Leech M, and Bernhagen J (2006). MIF: A new cytokine link between rheumatoid arthritis and atherosclerosis. Nat. Rev. Drug Discov.

Oliphant TE (2006). A guide to NumPy (Trelgol Publishing).

Oliphant TE (2007). SciPy: Open source scientific tools for Python. Comput. Sci. Eng 9, 10–20.

Park J-E, Polaski K, Meyer K, and Teichmann SA (2018). Fast Batch Alignment of Single Cell Transcriptomes Unifies Multiple Mouse Cell Atlases into an Integrated Landscape. BioRxiv.

Pedregosa F, and Varoquaux G (2011). Scikit-learn: Machine learning in Python.

Qiu X, Mao Q, Tang Y, Wang L, Chawla R, Pliner HA, and Trapnell C (2017). Reversed graph embedding resolves complex single-cell trajectories. Nat. Methods 14, 979–982. [PubMed: 28825705]

Rahmani M, and Atia GK (2017a). Spatial Random Sampling: A Structure-Preserving Data Sketching Tool. IEEE Signal Process. Lett 24, 1398–1402.

Rahmani M, and Atia GK (2017b). Spatial Random Sampling: A Structure-Preserving Data Sketching Tool. IEEE Signal Process. Lett

Ronen J, and Akalin A (2018). netSmooth: Network-smoothing based imputation for single cell RNA-seq. F1000Research 7, 8. [PubMed: 29511531]

Ross DA, Lim J, Lin RS, and Yang MH (2008). Incremental learning for robust visual tracking. Int. J. Comput. Vis

Rozenblatt-Rosen O, Stubbington MJT, Regev A, and Teichmann SA (2017). The Human Cell Atlas: From vision to reality. Nature 550, 451–453. [PubMed: 29072289]

Santos LL, and Morand EF (2009). Macrophage migration inhibitory factor: A key cytokine in RA, SLE and atherosclerosis. Clin. Chim. Acta

Saunders A, Macosko EZ, Wysoker A, Goldman M, Krienen FM, de Rivera H, Bien E, Baum M, Bortolin L, Wang S, et al. (2018). Molecular Diversity and Specializations among the Cells of the Adult Mouse Brain. Cell 174, 1015–1030.e16. [PubMed: 30096299]

Sim DG, Kwon OK, and Park RH (1999). Object matching algorithms using robust Hausdorff distance measures. IEEE Trans. Image Process.

Tanaka A, To J, O'Brien B, Donnelly S, and Lund M (2017). Selection of reliable reference genes for the normalisation of gene expression levels following time course LPS stimulation of murine bone marrow derived macrophages. BMC Immunol.

Vallejos CA, Risso D, Scialdone A, Dudoit S, and Marioni JC (2017). Normalizing single-cell RNA sequencing data: Challenges and opportunities. Nat. Methods 14, 565–571. [PubMed: 28504683]

Vinh N, Epps J, and Bailey J (2010). Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. J. Mach. Learn. Res

Wolf FA, Angerer P, and Theis FJ (2018). SCANPY: Large-scale single-cell gene expression data analysis. Genome Biol. 19.

Wolock SL, Lopez R, and Klein AM (2018). Scrublet: computational identification of cell doublets in single-cell transcriptomic data. BioRxiv.

Yip SH, Sham PC, and Wang J (2018). Evaluation of tools for highly variable gene discovery from single-cell RNA-seq data. Brief. Bioinform

Zeisel A, Hochgerner H, Lönnerberg P, Johnsson A, Memic F, van der Zwan J, Häring M, Braun E, Borm LE, La Manno G, et al. (2018). Molecular Architecture of the Mouse Nervous System. Cell.

Zheng GXY, Terry JM, Belgrader P, Ryvkin P, Bent ZW, Wilson R, Ziraldo SB, Wheeler TD, McDermott GP, Zhu J, et al. (2017). Massively parallel digital transcriptional profiling of single cells. Nat. Commun 8.

PRIMER

Large-scale scRNA-seq analysis:

Current approaches and challenges.

Single-cell RNA-sequencing (scRNA-seq) experiments routinely profile hundreds of thousands of cells, with billions of cells likely to be profiled in the near future. Deriving biological insights from single-cell datasets requires computationally intensive operations such as clustering, visualization, and nonlinear data integration. Clustering analyses assign more similar cells to groups, or clusters, that may correspond to biologically meaningful structure. Visualization lets researchers develop an intuition about variation in a dataset by highlighting important variability within an interpretable, usually two-dimensional, plot. Data integration requires searching for similar transcriptional structure across two or more datasets and removing confounding differences like batch effects. Performing these analyses on very large datasets is already not feasible for many researchers without expensive computational infrastructure, and is still time consuming for researchers with enough compute power. Instead, researchers often perform initial analysis on a random subset of cells chosen with uniform probability for each cell, which is prone to removing rare cell types and negates the advantage of performing large-scale experiments.

A geometric interpretation of single-cell datasets.

Throughout this paper, we understand a single-cell dataset as a collection of points in a multidimensional “transcriptomic space.” Each point in a dataset corresponds to a single cell and its location is determined by measuring gene expression. The abstraction of points within a multidimensional space enables us to reason about the “geometry” of a scRNA-seq dataset, including the particularly useful concepts of distance and volume. Cells with closer distances in the transcriptomic space have greater transcriptomic similarity. Similarly, a shape that occupies a greater volume of the transcriptomic space represents greater transcriptomic variation.

Overview of geometric sketching.

Here we introduce an approach for intelligently choosing a smaller subsample of a dataset that aims to represent as much of the transcriptional heterogeneity as possible. The key insight is that rare transcriptomic states (e.g., rare cell types) may have enough variation to occupy a similar volume of the transcriptomic space as that of common states (e.g., common cell types), but where cells belonging to common states more densely occupy the transcriptomic space. Instead of uniformly subsampling with equal probability for each cell, our algorithm subsamples more evenly over the volume occupied by the dataset, which we refer to as *geometric sketching* (Figure 1). Geometric sketching approximates the geometry of a scRNA-seq dataset by using equal-volume boxes within the transcriptomic space to “cover” all of the cells in the dataset, where each box contains at least once cell. Once we have obtained the covering, we use the covering boxes to sample evenly across the transcriptomic space. This approach naturally preserves the representation of cells from rarer transcriptional states that still occupy large regions of the transcriptomic space. This approach is also designed to be extremely efficient so that complex downstream analyses like clustering, visualization, and integration can be orders of magnitude more efficient when applied to a geometric sketch, instead of the full data, while maintaining accuracy.

HIGHLIGHTS

- Method to subsample massive scRNA-seq datasets while preserving rare cell states
- Resulting “sketch” accelerates clustering, visualization, and integration analyses
- Highlighting rare cells helps uncover a rare subtype of inflammatory macrophages
- Sketches can boost the utility of single-cell data for labs with limited resources

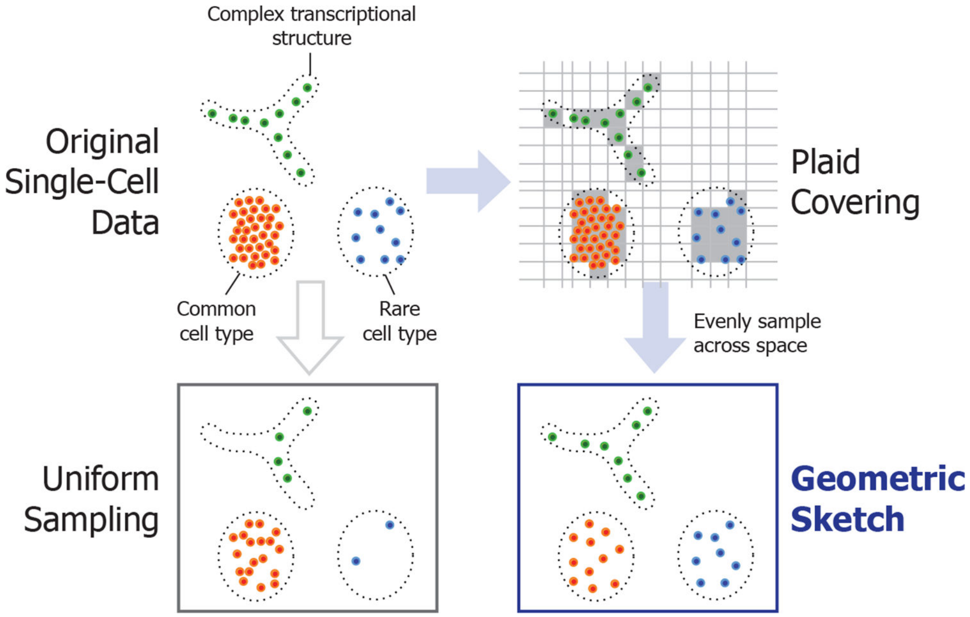
Complex transcriptional structure
Original Single-Cell Data
Common cell type
Rare cell type
Plaid Covering
Evenly sample across space
Uniform Sampling
Geometric Sketch

Figure 1. Illustration of Geometric Sketching

We first cover the data points with equal-sized boxes (which we refer to as a *plaid covering*) to approximate their geometry, then sample data points by first spreading the desired total sample count over the boxes as evenly as possible, then choosing the assigned number of samples within each box uniformly at random. The resulting sketch more evenly covers the landscape of the data compared to uniform sampling of points, where the latter is more prone to omitting rare cell types or transcriptional patterns.

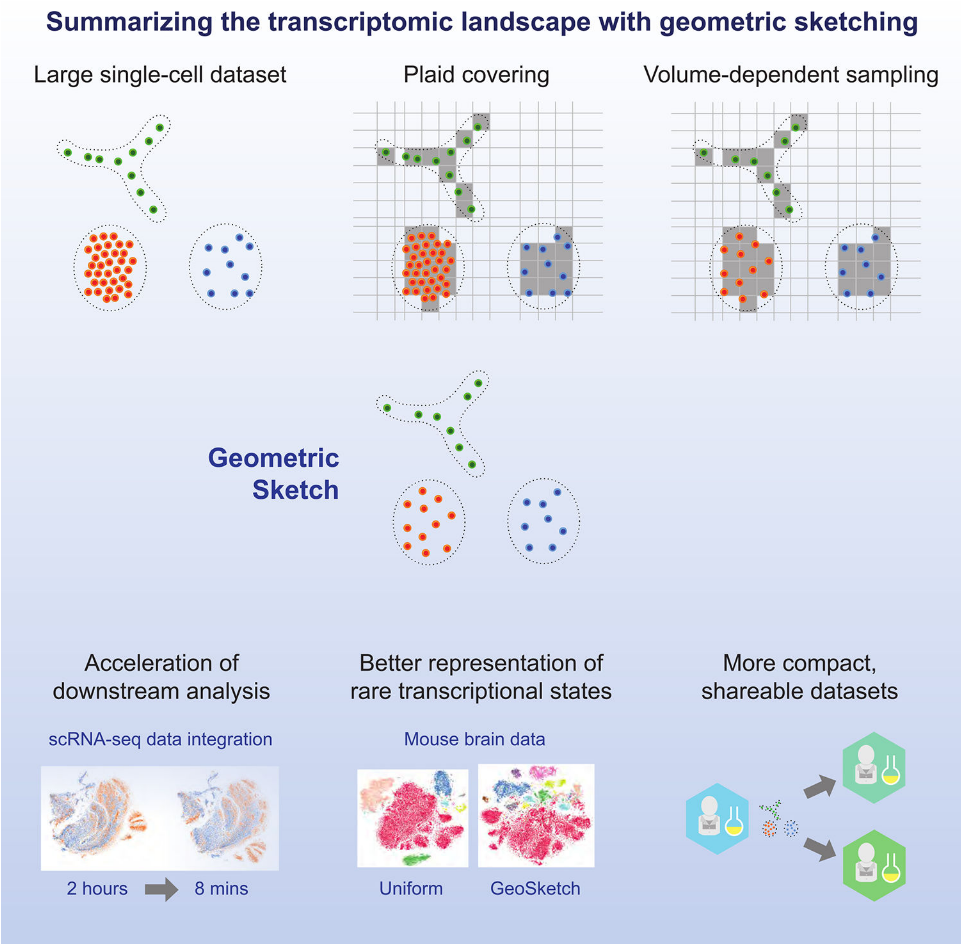
Summarizing the transcriptomic landscape with geometric sketching
Large single-cell dataset
Plaid covering
Volume-dependent sampling
Geometric Sketch
Acceleration of downstream analysis
Better representation of rare transcriptional states
More compact, shareable datasets
scRNA-seq data integration
2 hours
8 mins
Mouse brain data
Uniform
GeoSketch

Figure 2. Geometric Sketching Workflow

A large single cell dataset is covered with equal-volume hypercubes assigned via a “plaid covering” (Method Details). The covering is used to evenly sample cells across the transcriptomic space to obtain a geometric sketch. The sketch construction is extremely efficient and can be used to accelerate downstream analyses. Sampling according the geometry of the dataset also naturally preserves rare cell types, improving the information content obtained by clustering analyses of the geometric sketch compared to other sampling methods. Geometric sketches can also be more efficiently shared with other researchers.

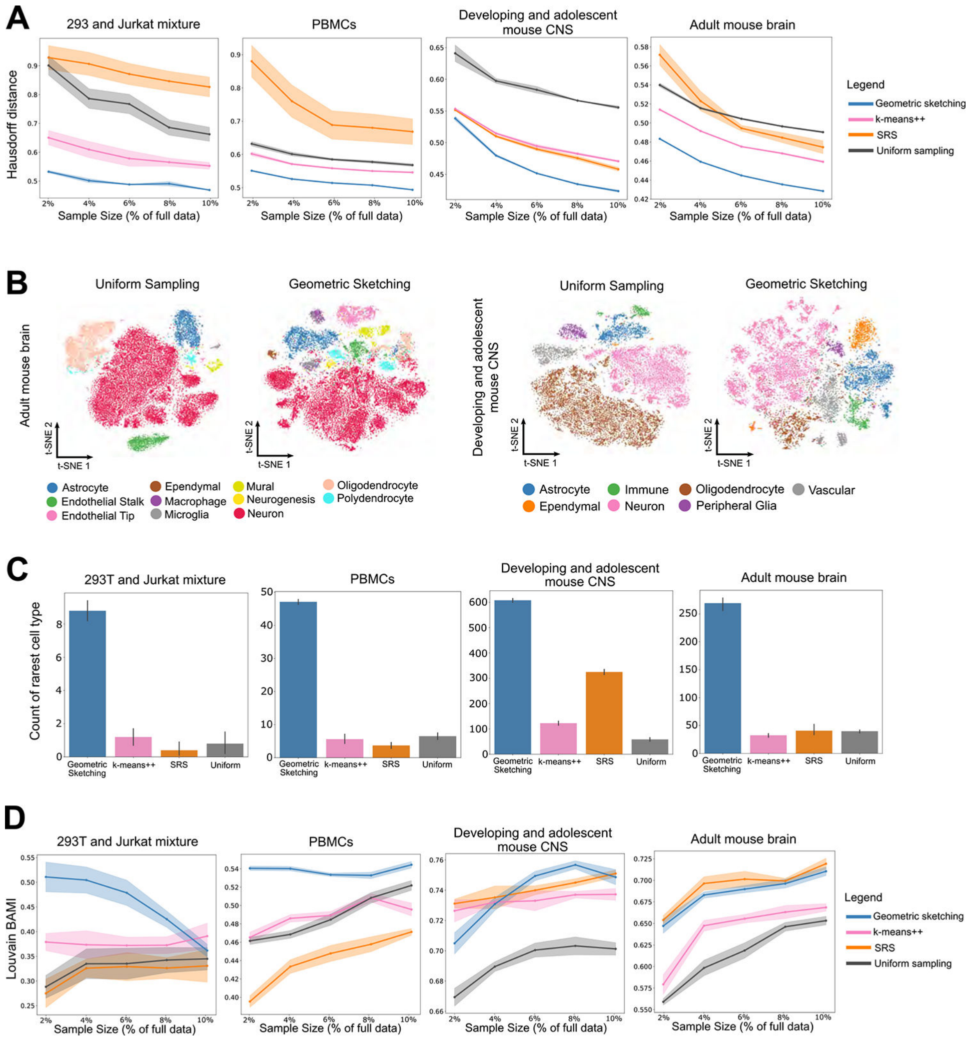

Figure 3. Geometric Sketching Outperforms Existing Sampling Approaches

(A) Geometric sketching yields more even coverage of the transcriptomic space. In our experiments, the Hausdorff distance measures the maximum distance from any point in the dataset to its closest point in the sketch; a lower Hausdorff distance indicates that the points represented by a sketch are in general closer to all of the points in the remainder of the dataset. Geometric sketching results in consistently lower Hausdorff distances than other sampling methods across a large number of sketch sizes and datasets. We use a robust Hausdorff distance that is less sensitive to small numbers of outlier observations (Method Details). Solid lines indicate means and shaded areas indicate standard error across 10 random trials for geometric sketching and uniform sampling and 4 random trials for *k*-means ++ and SRS (due to long runtimes). **(B)** Geometric sketches contain more balanced summaries of the transcriptional landscape. *t*-SNE visualizations of sketches containing 2% of the cells from the adult mouse brain (Saunders et al., 2018) and from the developing and adolescent mouse CNS (Zeisel et al., 2018) using uniform random sampling and geometric sketching, with increased representation of rare cell types in the geometric sketch. Numbers of cells from each cell type are given in Tables S3–S4. Uniform sampling, which does not evenly consider the transcriptional space, produces visualizations that are poor at capturing transcriptional heterogeneity. Geometric sketching substantially underrepresents oligodendrocytes in both datasets compared to uniform sampling, which is expected given the low transcriptional heterogeneity among oligodendrocytes as quantified by differential entropy (Method Details; Tables S3–S4). Visualizations based on other sampling approaches as well as a different visualization method are provided in Figure S1. (**C**) Geometric sketches preserve rare cell types in the subsampled data. In sketches containing 2% of the total dataset, we counted the number of cells that belong to the rarest cell type in each dataset: 293T cells (0.66% of total cells) in a 293T/Jurkat mixture, dendritic cells (0.38% of total) in a dataset of 68k PBMCs, macrophages (0.25% of total) in a dataset of adult mouse brain cells, and ependymal cells (0.60% of total) in a dataset of developing and adolescent mouse CNS cells. Higher count indicates increased representation of the rare cell type in the sketch. Bar height indicates means and error bars indicate standard error across 10 random trials for geometric sketching and uniform sampling and 4 random trials for *k*-means++ and SRS (due to long runtimes). Comparison of rare cell type representation over different sketch sizes is shown in Figure S2B. (**D**) Geometric sketching is consistently effective at distinguishing biological cell types via clustering. Louvain clustering was applied to a subsample of the dataset, cluster labels were transferred to the full dataset using a *k*-nearest-neighbor classifier fit to the sketch, and the balanced adjusted mutual information (BAMI) was measured between the unsupervised cluster labels and the labels corresponding to biological clusters provided by each previous study (Method Details). Higher score indicates greater agreement between unsupervised clustering and biological cell type labels. Solid lines indicate means and shaded areas indicate standard error across 10 random trials for geometric sketching and uniform sampling and 4 random trials for *k*-means++ and SRS (due to long runtimes). Unsupervised clustering of geometric sketches consistently recapitulates biological cell types better than clustering results obtained by uniform sampling. Other non-uniform sampling methods, *k*-means++ and SRS, show performance comparable to ours in a few cases, but only geometric sketching obtains competitive performance across all settings. Because samples are drawn without replacement, clustering accuracy may approach that of uniform sampling as the sketch size increases, as is the case in the 293T/Jurkat mixture experiments.

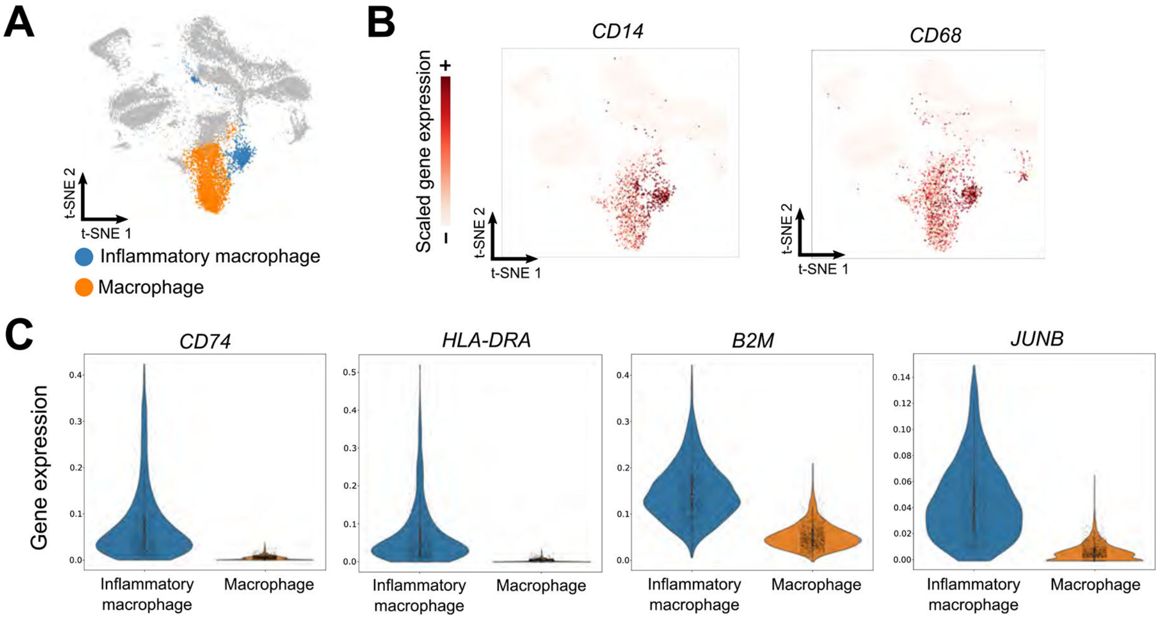

Figure 4. Identification of Subpopulation of Inflammatory Macrophages Identified Using Geometric Sketching

A geometric sketch of 20,000 cells was obtained from a full dataset of 254,941 cells from human umbilical cord blood. Analysis of clusters obtained by the Louvain community detection algorithm reveals multiple clusters of macrophages **(A)**, defined by *CD14* and *CD68* marker gene expression **(B)**. A rare subpopulation of these macrophages is in turn defined by inflammatory marker gene expression (*CD74*, *HLA-DRA*, *B2M*, and *JUNB*) (**C**), providing insight into an important but comparatively rarer immunological process.

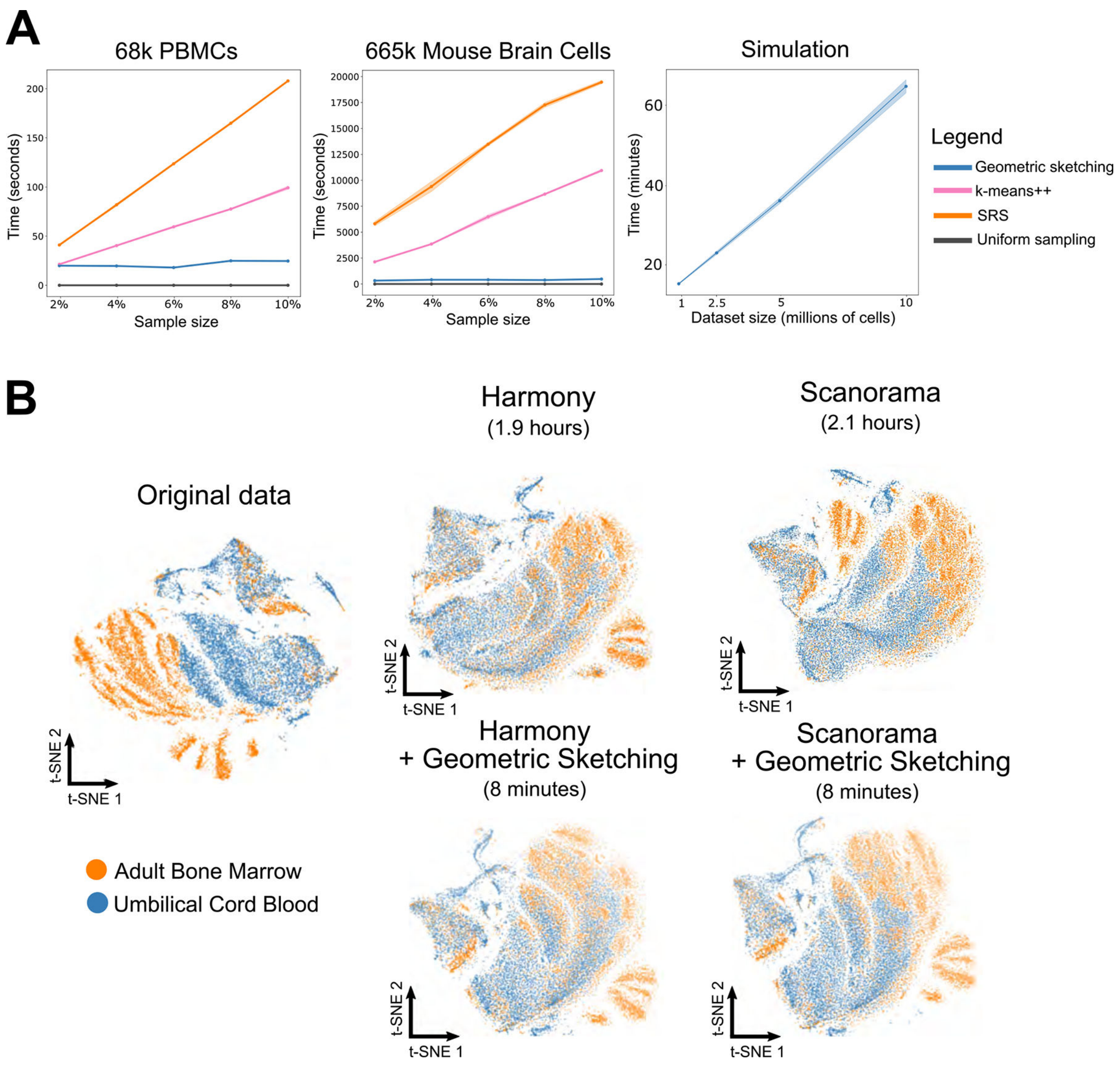

Figure 5. Geometric Sketching Efficiently Scales to Large Single-Cell Datasets

(A) Geometric sketching is substantially more efficient than other data-dependent subsampling approaches, SRS and *k*-means++. Although uniform sampling is fastest because it does not consider any properties of the dataset, geometric sampling obtains a sketch that preserves transcriptional heterogeneity while running in close to linear time in the size of the data, largely independent of the requested number of samples. Solid lines indicate means and shaded areas indicate standard error across 10 random trials for geometric sketching and uniform sampling and 4 random trials for *k*-means++ and SRS (due to long runtimes). Geometric sketching has a practical runtime of around 67 minutes when sampling 20,000 cells from a simulated dataset with 10 million cells, which was obtained by resampling from a dataset of mouse CNS cells (Zeisel et al., 2018). (**B**) Geometric sketching accelerates single-cell data integration tools. Geometric sketching can help accelerate existing tools for scRNA-seq data integration. We use two existing algorithms for scRNA-seq integration, namely Harmony (Korsunsky et al., 2018) and Scanorama (Hie et al., 2019), but note that our approach works for other integrative algorithms as well. Learning alignment vectors among geometric sketches, which are then used to transform the full datasets to remove tissue-specific differences (Method Details), decreases integration time of 534,253 human immune cells from hours to minutes while achieving comparable integration quality (Figure S4B).

KEY RESOURCES TABLE

REAGENT or RESOURCE	**SOURCE**	**IDENTIFIER**
Deposited Data		
Raw sequence data	NCBI SRA	Uploaded, pending SRA approval
293T cell expression matrices	10x Genomics	https://support.10xgenomics.com/single-cell-gene-expression/datasets
293T and Jurkat cell mixture expression matrices	10x Genomics	https://support.10xgenomics.com/single-cell-gene-expression/datasets
Human PBMC expression matrices	10x Genomics	https://support.10xgenomics.com/single-cell-gene-expression/datasets
Developing and adolescent mouse CNS expression matrices	(Zeisel et al., 2018)	http://mousebrain.org
Adult mouse brain cell expression matrices	(Saunders et al., 2018)	http://dropviz.org/
M-CSF macrophage expression matrices	(Hie et al., 2019)	http://scanorama.csail.mit.edu
GM-CSF macrophage expression matrices	This paper	http://geosketch.csail.mit.edu
Biological Samples		
Human buffy coats	Massachusetts General Hospital	N/A
Software and Algorithms		
geosketch Python package	This paper	https://github.com/brianhie/geosketch
Scanorama	(Hie et al., 2019)	https://github.com/brianhie/scanorama
Harmony	(Korsunsky et al., 2018)	https://github.com/immunogenomics/harmonv
SCANPY	(Wolf et al.,2018)	https://scanpv.readthedocs.io/en/latest/