bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Iterative Refinement of Cellular Identity from Single-Cell Data Using Online Learning

Chao Gao[1], Joshua D. Welch[1,2]

[1] Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor MI 48109, USA
[2] Department of Computer Science and Engineering, University of Michigan, Ann Arbor MI 48109, USA
gchao@umich.edu, welchjd@umich.edu

Abstract. Recent experimental advances have enabled high-throughput single-cell measurement of gene expression, chromatin accessibility and DNA methylation. We previously used integrative non-negative matrix factorization (iNMF) to jointly learn interpretable low-dimensional representations from multiple single-cell datasets using dataset-specific and shared metagene factors. These factors provide a principled, quantitative definition of cellular identity and how it varies across biological contexts. However, datasets exceeding 1 million cells are now widely available, creating computational barriers to scientific discovery. For instance, it is no longer feasible to use the entire available datasets as inputs to implement standard pipelines on a personal computer with limited memory capacity. Moreover, there is a need for an algorithm capable of iteratively refining the definition of cellular identity as efforts to create a comprehensive human cell atlas continually sequence new cells.

To address these challenges, we developed an online learning algorithm for integrating massive and continually arriving single-cell datasets. We extended previous online learning approaches for NMF to minimize the expected cost of a surrogate function for the iNMF objective. We also derived a novel hierarchical alternating least squares algorithm for iNMF and incorporated it into an efficient online algorithm. Our online approach accesses the training data as mini-batches, decoupling memory usage from dataset size and allowing on-the-fly incorporation of new data as it is generated. The online implementation of iNMF converges much more quickly using a fraction of the memory required for the batch implementation, without sacrificing solution quality. Our new approach enables factorization of 939489 single cells from 9 regions of the mouse brain on a standard laptop in $\sim$ 30 minutes. Furthermore, we construct a multi-modal cell atlas of the mouse motor cortex by iteratively incorporating seven single-cell datasets from three different modalities generated by the BRAIN Initiative Cell Census Network over a period of two years.

Our approach obviates the need to recompute results each time additional cells are sequenced, dramatically increases convergence speed, and allows processing of datasets too large to fit in memory. Most importantly, it facilitates continual refinement of cell identity as new single-cell datasets from different biological contexts and data modalities are generated.

Keywords: Single-Cell Genomics, Multi-Omic Integration, Nonnegative Matrix Factorization, Online Learning.

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

1 Introduction

1.1 Quantitative Definition of Cell Identity from Single-Cell Data

The cell is the building block of life. Defining cellular identity is foundational to a genomic approach to medicine, because discovering what goes wrong in disease requires a reference map of the molecular states of healthy cells. Cells have long been qualitatively characterized by a combination of features such as morphology, presence or absence of cell surface proteins, and high-level function [18]. Recently, high-throughput single-cell sequencing technologies have enabled researchers to profile individual cells using a wide range of measurements, including mRNA, chromatin accessibility and DNA methylation [15]. Multi-modal single-cell data offers tremendous opportunities for unbiased, comprehensive, quantitative definition of discrete cell types and continuous cell states. The resulting catalog of normal cell types will lead to gaining a deeper insight into fields like neuroscience, developmental biology and physiology [13]. More importantly, however, knowing the molecular profiles of individual cells points to biochemical mechanisms by which genetic and environmental factors cause disease, which is crucial for moving toward therapeutic intervention.

Multiple features contribute to cell identity, including gene expression, epigenomic modifications, and spatial location within a tissue, but it is not currently possible to simultaneously measure all of these quantities within the same single cells. Experimental methods for assaying transcriptome and epigenome from the same single cells have been demonstrated, but have not been widely adopted due to significant limitations in data quality and/or scalability. Large-scale gene expression, chromatin accessibility, DNA methylation, chromatin conformation, and spatial transcriptomic measurements of different individual cells are now widely available, but these features have generally been used separately to identify cell clusters representing putative cell types, and it is unclear how these different definitions of molecular cell identity are related. Combining these diverse features into an integrated picture of cellular state is crucial to defining cell identity.

1.2 The Need for Scalable Integration of Single-Cell Data

Single-cell data integration thus represents a crucial step toward enabling quantitative definition of cell identity, but existing computational approaches do not address this need. Three unique aspects make single-cell integration challenging: (1) unlike bulk multi-omic data, only one modality is available from each single cell; (2) both similarities and differences among datasets are of interest, so batch effect removal is not sufficient; and (3) number of samples (n) per dataset is large and rapidly growing. Several recent single-cell data integration approaches have been developed, including Seurat v3, Scanorama, and Conos [15, 6, 1], but these approaches are not designed to integrate multiple data types and/or have difficulty scaling to massive datasets. Furthermore, none of these existing methods can incorporate new data, instead requiring recalculating results each time new data arrive. Currently, scRNA-seq datasets are growing more rapidly than other single-cell data modalities, but we also anticipate rapid growth in the scale of these other data modalities in the near future. Indeed, a recent study assayed more than 100,000 cells with single-cell ATAC-seq [3], and a recent spatial transcriptomic study assayed 1 million cells [12]. A key insight motivating our approach is that techniques for so-called “online learning” [11], in which calculations are performed on-the-fly as new data continuously becomes available (as in many internet applications), provides a path to scalable single-cell data integration.

1.3 Integrative Nonnegative Matrix Factorization

In this paper, we build upon the nonnegative matrix factorization approach at the heart of our recently developed LIGER algorithm [16] to develop an online learning framework, allowing the algorithm to easily scale to millions of cells, overcoming runtime and memory limitations, and avoiding the need to re-compute cell type definitions from scratch each time new data arrive. The intuition behind LIGER is to jointly infer a set of latent factors (”metagenes”) that represent the same biological signals in each dataset, while also retaining the ways in which these signals differ across datasets. These shared and dataset-specific factors can then be used to jointly identify cell types and states, while also identifying and retaining cell-type-specific differences in the metagene features that define cell identities. LIGER takes as inputs two or more single-cell datasets,

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

which may be scRNA-seq experiments across different individuals, time points, or species. The inputs to LIGER may even be measurements from different molecular modalities, such as single-cell epigenome data or spatial transcriptomic data that assay a common set of genes. LIGER relies upon integrated nonnegative matrix factorization (iNMF) [17], which solves the following optimization problem:

$$\min_{\substack{W,V_i,H_i\geq 0\\ i\in 1,..,N}} \sum_{i=1}^{N} \|X_i - (W+V_i)H_i\|_F^2 + \lambda \sum_{i=1}^{N} \|V_i H_i\|_F^2 \tag{1}$$

to jointly factorize N datasets (each consisting of a genes × cells matrix X_i), inferring both shared (W) and dataset-specific (V_i) factors (**Fig. 1**). Each factor, or metagene, represents a distinct pattern of gene co-regulation, often corresponding to biologically interpretable signals–like the genes that define a particular cell type. The dataset-specific metagenes (V_i) allow robust representation of highly divergent datasets; for example, in a cross-species scRNA-seq analysis, dataset-specific factors will capture differences in co-expression patterns of homologous genes. The factorization can also accommodate missing cell types by generating a factor with a very large dataset-specific component.

1.4 Online Matrix Factorization

Since its proposal by Lee and Seung two decades ago, nonnegative matrix factorization (NMF) has been widely used to learn interpretable representations of high-dimensional data [10]. NMF is a non-convex optimization problem, so the strongest possible convergence guarantee for an NMF algorithm is that it converges to a local minimum of the objective function. However, the widely used multiplicative update algorithm has no such theoretical convergence guarantee and slow empirical convergence. More efficient NMF algorithms based on block coordinate descent, including alternating nonnegative least squares (ANLS) and hierarchical alternating least squares (HALS), have been developed [8], which show strong empirical performance and are guaranteed to converge to a local minimum. However, even these approaches are not able to efficiently handle large and streaming inputs such as images and videos (which often arise in web applications, hence the name "online learning") [2]. As opposed to a batch learning algorithm, an online matrix factorization approach accesses the data only as either single data points or mini-batches and continually updates the basis elements in the metagenes in our context. Subsequently, an online NMF algorithm was developed with guaranteed convergence to a local minimum as other batch NMF methods do. This approach showed strong empirical performance and extremely fast convergence compared to batch NMF [11].

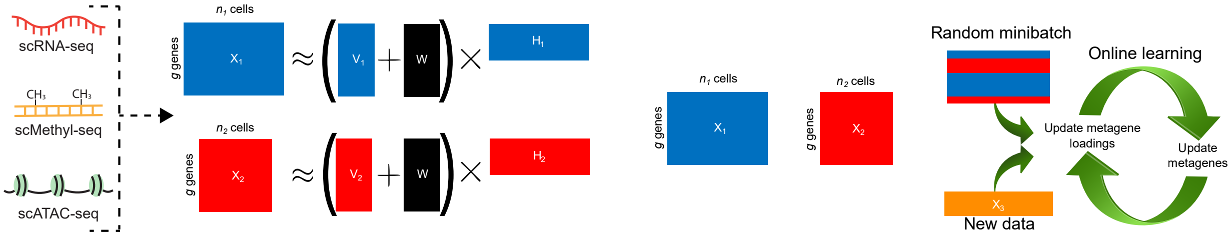

Fig. 1. Overview of the online iNMF approach

1.5 Online iNMF

In this study, we extend the online NMF approach of [11] to make it suitable for iNMF tasks. Online iNMF provides significant advantages in two related settings: (1) integration of large multi-modal datasets by cycling through the data multiple times in small mini-batches and (2) integration of continually arriving datasets, where the entire dataset is not available at any point during training (**Fig. 1**). Our proposed online iNMF method iteratively refines shared and dataset-specific metagene representations using mini-batches from multiple single-cell datasets. In this way, it allows for processing large-scale multi-modal datasets with limited memory capacity as well as integrating continually arriving datasets.

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

2 Methods

2.1 Derivation of Novel HALS Algorithm for iNMF

In the previous implementation of iNMF that we developed [16], we carried out an ANLS algorithm to update H_i, W, and V_i in an iterative process. Briefly, ANLS optimizes the iNMF objective by iteratively solving a nonnegative least squares problem to update each of the matrices (H_i, W, V_i) holding the others fixed. For example, the update for H_i ($i \in \{1, \ldots, N\}$) is:

$$H_i = \underset{H_i \geq \mathbf{0}}{\arg\min} \left\| \begin{pmatrix} W + V_i \\ \sqrt{\lambda} V_i \end{pmatrix} H_i - \begin{pmatrix} X_i \\ \mathbf{0}_{m \times n_i} \end{pmatrix} \right\|_F^2 \tag{2}$$

This is a convex nonnegativity-constrained least squares problem that can be solved efficiently using the block principal pivoting algorithm [9]. This ANLS algorithm is guaranteed to converge to a local minimum, and we previously showed that it outperforms the multiplicative updates in practice [16].

Another type of NMF algorithm, HALS, also provides guaranteed convergence to a local minimum, but often shows more efficient convergence in practice [8]. Thus, we sought to derive a novel HALS algorithm for optimizing the iNMF objective. A HALS derivation proceeds by re-writing the objective function as a sum of rank-one approximations (one for each of the K inner dimensions of the factorization), then deriving a closed-form solution for each of the K basis vectors holding the others fixed. In the case of iNMF, this can be considered as a block coordinate descent strategy with $(2N+1)K$ vector blocks. For example, for $V_{i,j}$, the jth factor in the dataset-specific metagene matrix V_i, we want to solve the optimization problem:

$$\begin{aligned} &\min_{V_{i,j} \geq 0} \sum_{i=1}^{N} \left\| X_i - WH_i - \sum_{\tilde{j} \neq j}^{K} V_{i,\tilde{j}} H_{i,\tilde{j}} - V_{i,j} H_{i,j} \right\|_F^2 + \lambda \left\| V_i H_i \right\|_F^2 \\ &= \sum_{i=1}^{N} \left\| V_{i,j} H_{i,j} - R_{i,j} \right\|_F^2 + \lambda \left\| V_{i,j} H_{i,j} + \sum_{\tilde{j} \neq j}^{K} V_{i,\tilde{j}} H_{i,\tilde{j}} \right\|_F^2 \end{aligned} \tag{3}$$

where $R_{i,j} = X_i - WH_i - \sum_{\tilde{j} \neq j}^{K} V_{i,\tilde{j}} H_{i,\tilde{j}}$.

Taking the derivative with respect to kth element in $V_{i,j}$ gives:

$$\frac{\partial}{\partial (V_{i,j})_k} = 2 \left[(1+\lambda) \left\| H_{i,j} \right\|_F^2 (V_{i,j})_k - H_{i,j}^\top (R_{i,j})_{k\cdot} \right] \tag{4}$$

We then set the derivative equal to 0 and solve for $(V_{i,j})_k$ subject to nonnegativity constraints. Applying the same process to all elements in $V_{i,j}$ yields the following update for jth column for V_i:

$$(V_i)_{\cdot j} = \left[\frac{1}{1+\lambda} (V_i)_{\cdot j} + \frac{(X_i H_i^\top)_{\cdot j} - (W + V_i)(H_i H_i^\top)_{\cdot j}}{(1+\lambda)(H_i H_i^\top)_{jj}} \right]_+ \tag{5}$$

where $[x]_+ = \max\{10^{-16}, x\}$. Similar derivations for H_i and W give:

$$W_{\cdot j} = \left[W_{\cdot j} + \frac{\sum_{i=1}^{N} (X_i H_i^\top)_{\cdot j} - (W + V_i)(H_i H_i^\top)_{\cdot j}}{\sum_{i=1}^{N} (H_i H_i^\top)_{jj}} \right]_+ \tag{6}$$

$$(H_i)_{j\cdot} = \left[\frac{[(W+V_i)^\top (W+V_i)]_{jj}}{[(W+V_i)^\top (W+V_i)]_{jj} + \lambda (V_i^\top V_i)_{jj}} (H_i)_{j\cdot} + \frac{[(W+V_i)^\top X_i]_{j\cdot} - [(W+V_i)^\top (W+V_i) H_i]_{j\cdot}}{[(W+V_i)^\top (W+V_i)]_{jj} + \lambda (V_i^\top V_i)_{jj}} \right]_+ \tag{7}$$

To optimize the iNMF objective using HALS, we randomly initialize the factor matrices and iteratively update each of the K factors of H_i, then W, then V_i. The HALS updates are very fast to compute because they involve only sparse matrix operations (see Results).

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

2.2 Optimizing a Surrogate Function for iNMF

We developed an online learning algorithm for integrative nonnegative matrix factorization by adapting a previously published strategy for online NMF [11]. The key innovation that makes it possible to perform online learning is to optimize a "surrogate function" that asymptotically converges to the same solution as the original iNMF objective. We can formulate NMF using the following objective function: $f(W,H) = \|X - WH\|_F^2$ where W and H are constrained to be nonnegative. The original online NMF paper proved that the following surrogate function converges almost surely to a local minimum as $t \to \infty$:

$$\hat{f}_t(W,H) = \frac{1}{t}\sum_{i=1}^{t} \|x_i - Wh_i\|_F^2 \tag{8}$$

where H and W are constrained to be nonnegative. We can then perform NMF in an online fashion by iteratively minimizing the expected cost $\hat{f}_t(H,W)$ as new data points x_t (or points randomly sampled from a large fixed training set) arrive. Intuitively, this strategy allows online learning because it expresses a formula for incorporating a new observation x_t given the factorization result $(W^{(t-1)}, H^{(t-1)})$ for previously seen data points. Thus, we can iterate over the data points one-by-one or in "mini-batches"—and also rapidly update the factorization when new data points arrive. For iNMF, where we have N data matrices $X_1, \ldots, X_N$ and data points x_i, the iNMF objective function is given by (1). The corresponding surrogate function is:

$$\hat{f}_t(W, V_1, ..., V_N, H_1, ..., H_N) = \frac{1}{t}\sum_{i=1}^{t} \|x_i - (W + V_{d_i})h_{d_i}\|_F^2 + \lambda \|h_{d_i} V_{d_i}\|_F^2 \tag{9}$$

where d_i indicates which dataset the ith data point belongs to.

For a new data point (or mini-batch of new data points) x_t, we first compute the corresponding metagene loading values h_{d_t}. In the original online NMF paper [11], the authors used a least angle regression algorithm (LARS). We chose to use the ANLS update (2) instead because it is highly efficient and solves the subproblem exactly in a single iteration. To update the shared (W) and dataset-specific (V_i) factors, we use the HALS updates (5) and (6), which are analogous to the updates used by Mairal et al [11]. Because the updates for W and V_i depend on all of the previously seen data points X_i and their cell factor loadings H_i, a naive implementation would require storing all of the data and cell factor loadings in memory. However, the HALS updates (5) and (6) depend on X_i and H_i only through the matrix products $A = h_i h_i^\top$ and $B = x_i h_i^\top$. These matrix products have only K^2 and mK elements respectively, allowing efficient storage, and can be computed incrementally with the incorporation of each new data point or mini-batch x_t:

$$\begin{aligned} A_i^{(t)} &= A_i^{(t-1)} + h_i^{(t)} h_i^{(t)\top} \\ B_i^{(t)} &= B_i^{(t-1)} + x_i^{(t)} h_i^{(t)\top} \end{aligned} \tag{10}$$

2.3 Implementation

We implemented online iNMF according to Algorithm 1 below. We used our previous Rcpp implementation of the block principal pivoting algorithm to calculate the ANLS updates for h_i. We implemented the HALS updates for W and V_i using native R, since the updates carry out only sparse matrix operations, which are highly optimized in R. Because the online algorithm does not require all of the data on each iteration (only a single data point or fixed-size mini-batch), we used the rhdf5 package [5] to load each mini-batch from disk on the fly. By creating HDF5 files with chunk size no larger than the mini-batch size, we were able to create an efficient implementation that never loads more than a single mini-batch of the data from disk at once. The mini-batch size for the dataset with the most samples is set to 5,000. For the others, the mini-batch size p_i is proportional to its data size. For a mini-batch size of 5,000 cells, we found that reading each mini-batch from disk added minimal overhead (less than 0.35 seconds per iteration) (**Fig. S1**). We also employed two heuristics that were used in the original online NMF paper: (1) we initialize the data-specific metagenes using K cells randomly sampled from the corresponding training data and (2) we remove the information older than two epochs from matrices A and B.

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Algorithm 1 Online learning for Integrative Nonnegative Matrix Factorization

```
Require: $(X_1)_{m\times n_1}, \dots, (X_N)_{m\times n_N}$
Initialize $A_i^{(0)} \in \mathbf{0}^{K\times K}$, $B_i^{(0)} \in \mathbf{0}^{M\times K}$, $i \in 1, \dots, N$
Initialize elements in $W$ using unif(0,2) ,$V_i$ using random sampled columns from $X_i$, $i \in 1, \dots, N$
for $t = 1$ to $T$ do
    Sample a mini-batch $x_i$ of size $p_i$ from $X_i$, $i \in 1, \dots, N$
    Compute $h_i$ using ANLS, $i \in 1, \dots, N$
        $h_i = \underset{H_i \geq \mathbf{0}}{\arg\min} \left\| \begin{pmatrix} W + V_i \\ \sqrt{\lambda} V_i \end{pmatrix} h_i - \begin{pmatrix} x_i \\ \mathbf{0}_{m\times p_i} \end{pmatrix} \right\|_F^2 \quad (11)$
    $A_i^{(t)} \leftarrow A_i^{(t-1)} + \frac{1}{p_i} h_i^{(t)} h_i^{(t)\top}$        ▷ Discard information older than 2 epochs
    $B_i^{(t)} \leftarrow B_i^{(t-1)} + \frac{1}{p_i} x_i^{(t)} h_i^{(t)\top}$        ▷ Discard information older than 2 epochs
    Update $W$ and $V_i$ using HALS
        $W_{\cdot j} = \left[ (W)_{\cdot j} + \frac{\sum_{i=1}^{N} (B_i^{(t)})_{\cdot j} - (W + V_i)(A_i^{(t)})_{\cdot j}}{\sum_{i=1}^{N} (A_i^{(t)})_{jj}} \right]_+ \quad (12)$
        $(V_i)_{\cdot j} = \left[ \frac{1}{1+\lambda} (V_i)_{\cdot j} + \frac{(B_i^{(t)})_{\cdot j} - (W + V_i)(A_i^{(t)})_{\cdot j}}{(1+\lambda)(A_i^{(t)})_{jj}} \right]_+ \quad (13)$
end for
Solve for $H_i$ using ANLS with updated $W$ and $V_i$, $i \in 1, \dots, N$
return $W, V_i, H_i$, $i \in 1, \dots, N$
```

3 Results

3.1 Online iNMF Converges Efficiently Without Loss of Accuracy Compared to Batch iNMF

We benchmarked the performance of our online iNMF algorithm using a large scRNA-seq dataset from the frontal cortex (194027 cells) and posterior cortex (116437 cells) of the adult mouse with selected 1369 highly variable genes. Because the online algorithm optimizes the expected cost, we tracked the value of the iNMF objective on both the training data (80% of the entire dataset) and a held-out testing set (20% of the entire dataset) not seen during training. For this experiment, we set the number of factors (K) and the tuning parameter λ to 40 and 5, respectively. Batch methods including multiplicative updates (Multiplicative), alternating nonnegative least squares (ANLS), hierarchical alternating least squares (HALS) were also tested for comparison. The $H_{frontal}$ and $H_{posterior}$ for the testing set were calculated using the W,$V_{frontal}$ and $V_{posterior}$ learned on the training set and then the testing objective was obtained.

As Fig. 2 shows, online iNMF using a mini-batch size of 5000 converges much faster than ANLS and Multiplicative, achieving a similar iNMF objective on the training set. Online iNMF performed even more favorably in both convergence time and minimization of the objective on the held-out set, confirming its capability, by design, of minimizing the expected loss. Reassuringly, we also find that the convergence behavior of the online algorithm on both training and test sets is relatively insensitive to the batch size. For batch sizes from 500 to 10,000, the convergence behavior is nearly identical. As the batch size approaches the dataset size (50,000 or 155,222), the first few iterations take considerably longer, slowing the convergence time, but the final objective remains unchanged.

Next, we investigated the effect of increasing dataset size n_i on the convergence time and memory usage of online iNMF with mini-batch size 5,000, in comparison with the batch methods. The Seurat v3 anchors method (based on Canonical correlation analysis) [15] was also included for comparison. We used the same adult mouse frontal and posterior cortex scRNA-seq datasets to generate training sets with different sizes, from 20,000 cells to 200,000 cells, while using a fixed testing set of size 10,000. We observed that, for a fixed test set, the length of time (number of mini-batches) needed to achieve convergence remains relatively constant once the number of cells exceeds some minimum threshold (around 60,000, in this case). The result highlights a key advantage of our online approach: convergence time appears to grow sub-linearly

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

with the number of cells added (**Fig. 3**). This behavior likely occurs because, for a cell population of fixed complexity (for example, a tissue containing 12 cell types), only some fixed number of observations is required to effectively learn the metagenes. Thus, using the entire dataset to update the shared and data-specific metagenes at each iteration becomes increasingly inefficient as the dataset size exceeds the minimum threshold size needed to learn the metagenes.

In contrast with the online approach, the convergence time grows rapidly with dataset size for multiplicative updating, ANLS, HALS, and CCA. Furthermore, by construction, the memory usage of online iNMF depends only on the mini-batch size, which can be determined for training based on the available computing resources, independent of dataset size. For example, the peak memory usage for online iNMF using a mini-batch of 5,000 samples is about 500MB, regardless of the dataset size. In contrast, the peak memory usage for the other batch methods increases significantly with n_i, because they have to load the entire dataset into memory. In fact, we were unable to run Seurat v3 with less than 32GB of RAM for datasets larger than 100,000 cells. Having established the efficiency of online iNMF in terms of time and memory usage, we next evaluated the quality of our new approach with respect to clustering and dataset alignment. We applied both online iNMF and ANLS (the implementation from our previous LIGER paper) to three benchmark scRNA-seq datasets from human PBMCs, human pancreas, and mouse cortex. We created UMAP visualizations of the resulting factor loadings, colored by either dataset labels or published cluster labels (**Fig. 4**). These plots allow visual and qualitative assessment of dataset alignment and cluster preservation. For all three datasets, online iNMF yields qualitatively similar visualizations with high dataset alignment and accurate preservation of the original data structure. We also compared the batch and online iNMF results by calculating the alignment metric as defined in our previous LIGER paper. Both approaches showed high

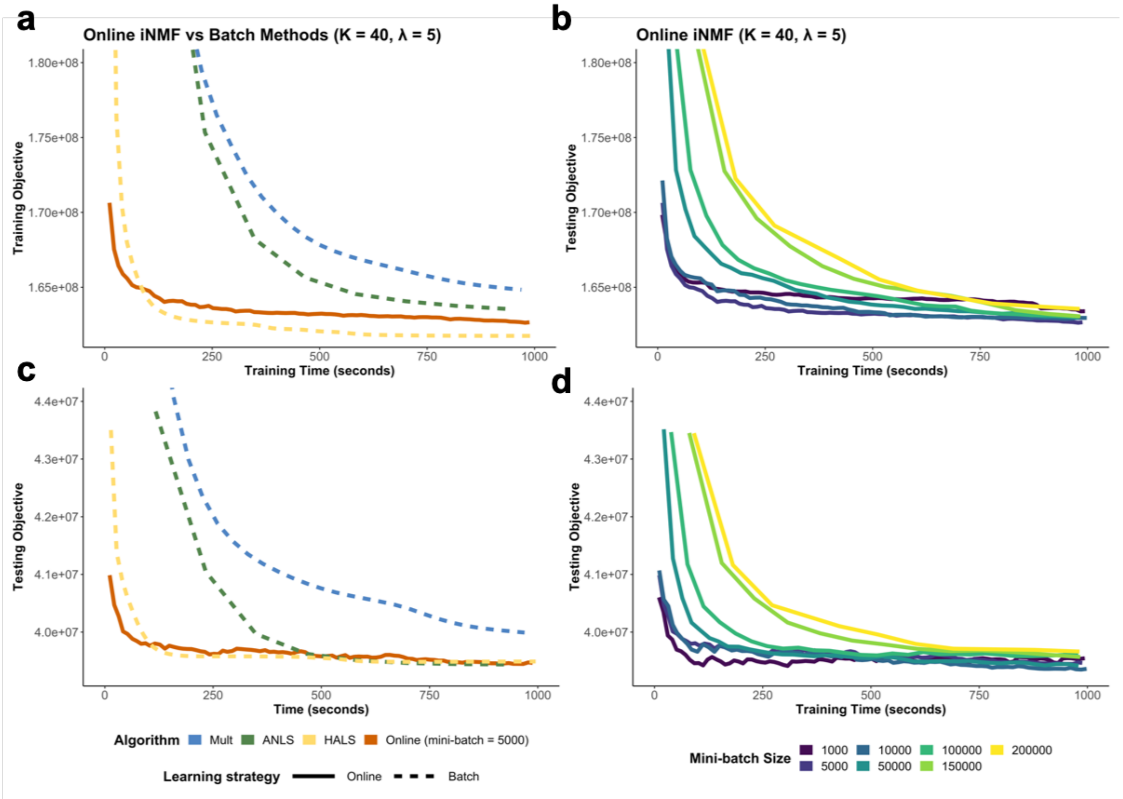

Fig. 2. Comparison of convergence behavior for online iNMF and three batch iNMF algorithms on scRNA-seq data from the adult mouse cortex Training set objective are plotted against the training time for multiplicative updates (Multiplicative), alternating nonnegative least squares (ANLS), hierarchical alternating least squares (HALS) and online iNMF.**(a)-(b)** compare the performance between online iNMF and batch methods. **(c)-(d)** compare the performance of online iNMF with different choices of mini-batch size.

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

alignment metrics, with online iNMF slightly exceeding batch iNMF in some cases (**Table 1**). Online and batch iNMF both accurately preserved the published cluster assignments for each dataset, with purity scores approaching the maximum possible value of 1 (**Table 1**).

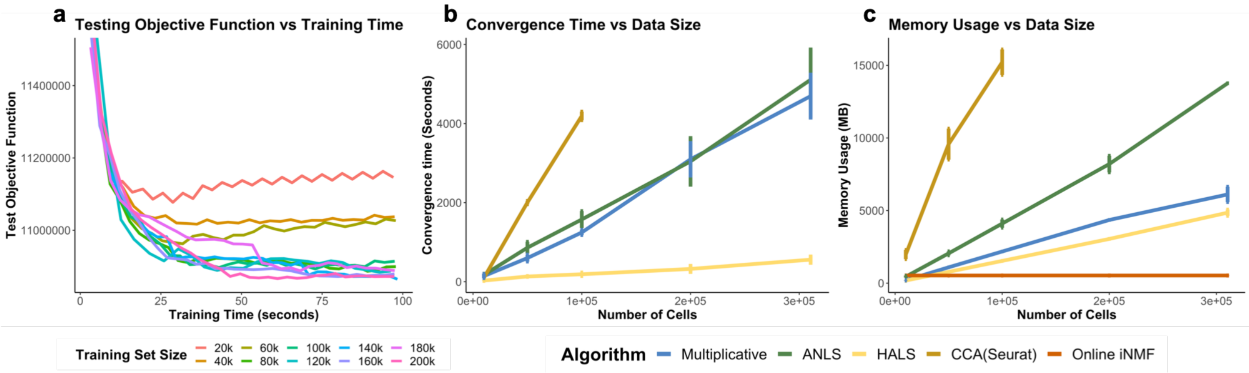

Fig. 3. Convergence time and peak memory usage of different algorithms in the presence of more training samples. **(a)** With the mini-batch size fixed at 5000, the time needed for convergence does not differ significantly among various sizes of the training set. **(b)** The convergence time of batch methods grows as more cells included in the training set. **(c)** In contrast to the fixed memory usage of online iNMF, more memory is requested for batch methods given a larger traning set.

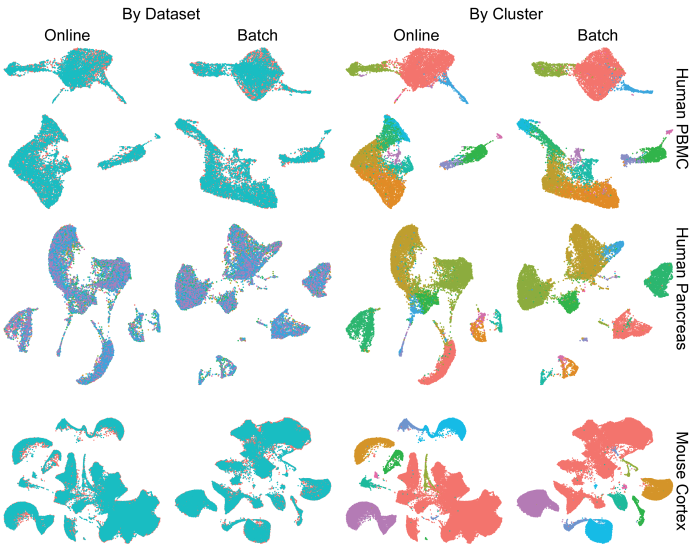

Fig. 4. UMAP visualizations comparing batch method and online iNMF results on human PBMC, human pancreas and mouse cortex datasets. Online iNMF is comparable to ANLS in preserving the data structure.

3.2 Online iNMF Rapidly Factorizes Whole Mouse Brain Dataset Using Fixed Memory

To demonstrate the scalability of online iNMF, we analyzed the scRNA-seq dataset recently published by Saunders et al. [14], which contains 939489 cells sampled from 9 regions (9 datasets) spanning the entire mouse brain. Using a mini-batch size of 5,000, $k = 50$ and $\lambda = 5$, we performed 5 epochs of training. Our online iNMF approach factorized the entire dataset in $\sim$ 30 minutes on a MacBook Pro with an Intel i7

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

processor using approximately 500 MB of RAM. We note that the published analysis by Saunders et al. did not analyze all 9 tissues simultaneously due to computational limitations. Furthermore, we estimate (based on the data in Fig. 2) that performing this analysis using our previous batch iNMF approach would have taken 4-5 hours and required over 40 GB of RAM.

We used this factorization to cluster the cells into 50 groups by assigning each cell to the factor on which it has the largest loading, a common procedure using NMF approaches. We then investigated the biological properties of our factorization. Visualizing the iNMF factors using UMAP showed that the factors clearly preserved the broad cell classes of the brain, as identified by Saunder et al (**Fig. 5**). Inspection of the UMAP plots further indicates substantial variation in the cell type composition of each brain region. For example, the "neurogenesis" cell class appears in a portion of the visualization that is almost exclusively occupied by cells from the hippocampus, consistent with the known presence of adult neural stem cells in the dentate gyrus. Reassuringly, our cluster assignments largely represent subtypes within the broad cell classes and do not span class boundaries (**Fig. 5**; cluster purity of 0.975). As expected, neurons show by far the most diversity with 11 subclusters. In contrast, choroid plexus, ependymal cells, macrophages, and mitotic cells each have only a single cluster. We also examined differences in the regional proportions of each cell cluster (**Fig. 5**). Neurons and oligodendrocytes showed by far the most regional variation in composition, consistent with previous analyses [19]. The total proportion of oligodendrocytes varied by region, but individual subtypes of oligodendrocytes were not region-specific, as expected. In contrast, individual subtypes of neurons were highly region-specific, reflecting highly diverse regional specializations in neuronal function.

Table 1. Alignment quality and cluster purity metrics for batch and online iNMF on three datasets.

	Alignment		**Purity**	
Dataset	**Batch**	**Online**	**Batch**	**Online**
Human PBMC	0.937	0.948	0.867	0.832
Human Pancreas	0.990	0.990	0.926	0.921
Mouse Cortex	0.828	0.843	0.986	0.984

3.3 Online iNMF Allows Iterative Refinement of Cell Atlas from Mouse Motor Cortex

One of the most appealing properties of our online learning algorithm is the ability to rapidly incorporate new data points as they arrive. This capability is especially useful for large, distributed collaborative efforts to construct comprehensive cell atlases [4, 7, 13]. These cell atlas projects involve multiple research groups asynchronously generating experimental data with constantly evolving protocols, making the ultimate cell type definition a moving target that needs to be constantly updated. To demonstrate the utility of online iNMF for iteratively refining cell type definitions, we used data generated by the BRAIN Initiative Cell Census Network (BICCN), an NIH-funded consortium that aims to identify all of the cell types in the mouse and human brains. During a pilot phase starting in 2018, the BICCN generated single-cell datasets from a single region of mouse brain (primary motor cortex) spanning 3 modalities (single-cell RNA-seq, and single-nucleus RNA-seq) and totaling more than 600,000 cells. These datasets have been publicly released on the BICCN data portal (https://nemoarchive.org/).Over the past 2 years, the 4 funded research groups have sequentially generated datasets, re-running the experiments as additional replicates and new protocols become available. Thus, this dataset provides an ideal case study to demonstrate how online iNMF can refine a cell atlas as additional cells are sequenced.

We used online iNMF to incorporate the datasets in chronological order, refining the factorization with each additional dataset (**Fig. 6**). These datasets represent a sort of historical record reflecting the rapid development of single-cell experimental techniques, with the first dataset generated using SMART-seq, the dominant protocol before the advent of droplet-based protocols. Subsequent datasets reflect newer technologies, including two versions of the 10X Genomics scRNA-seq protocol (v2 and v3); droplet-based single-nucleus RNA-seq; and droplet-based single-nucleus ATAC-seq. We used a fixed mini-batch size of 5,000 cells, $k = 40$, $\lambda = 5$, and performed a single epoch of training (each cell participates in exactly one mini-batch). When adding a new dataset, we incorporated a new dataset-specific factor matrix V_i and randomly initialized it. We did not use the previously seen data to refine the factors after the initial single epoch per dataset.

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

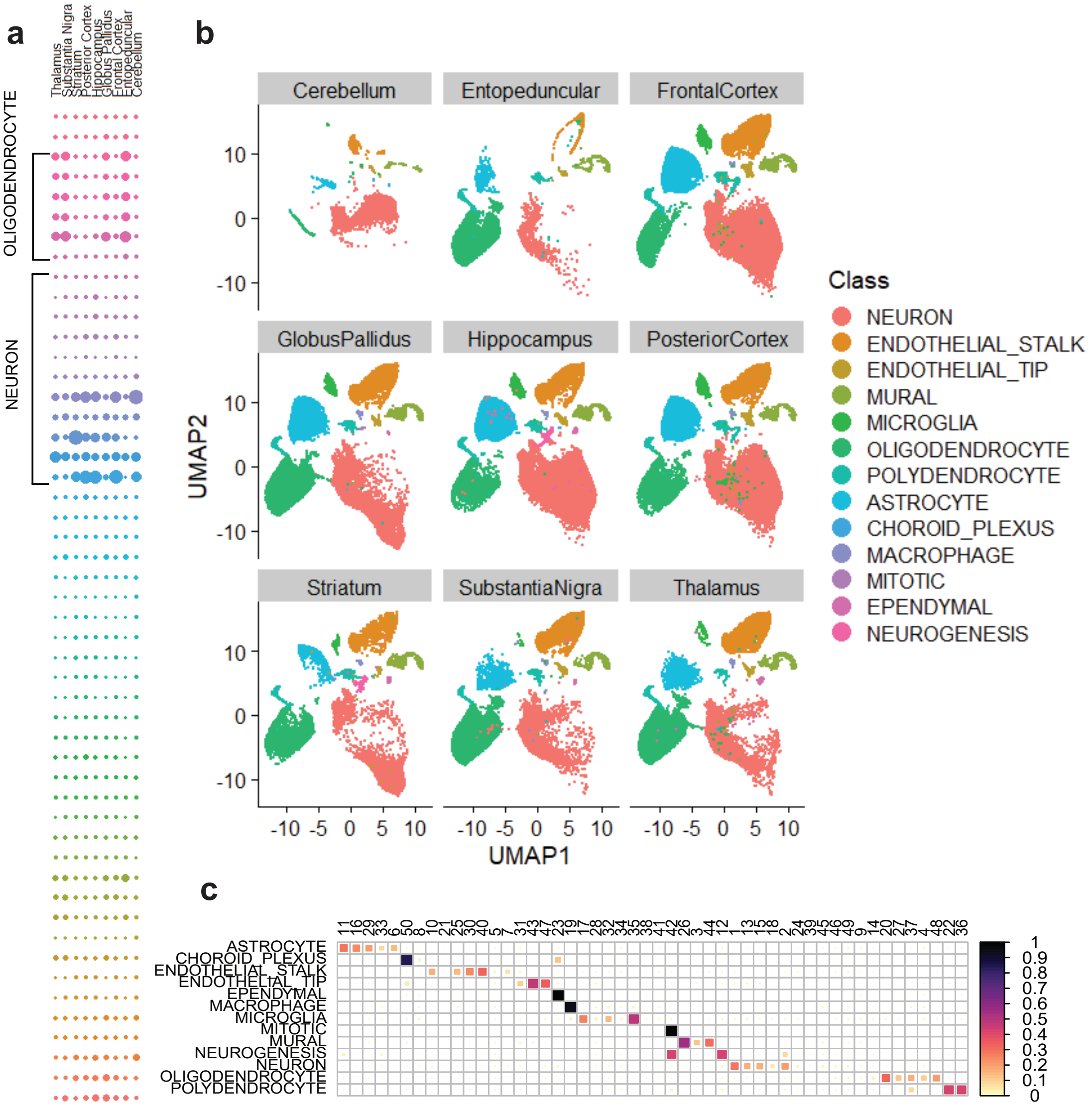

Fig. 5. Online iNMF analysis of 939489 single cells from 9 regions of mouse brain. **(a)** Dot plot showing the proportion of each of 50 clusters inferred from iNMF in each brain region. **(b)**UMAP visualization of the iNMF factors for learned for each brain region, colored by the published cell class labels of Saunders et al.

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

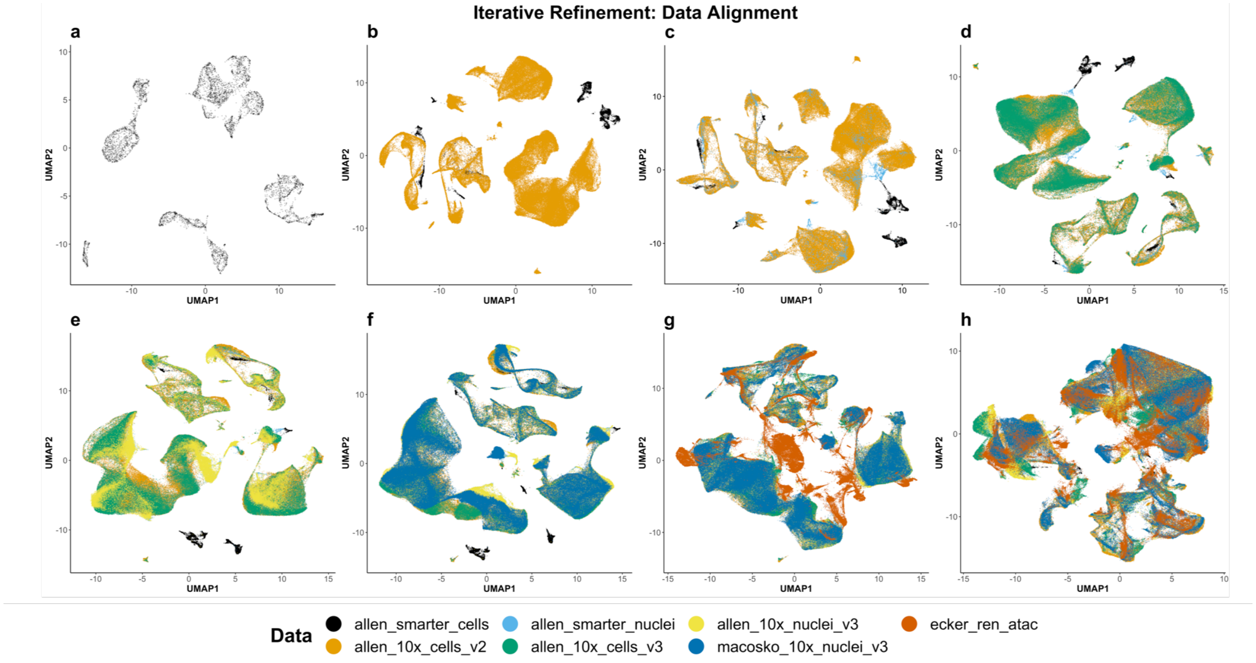

Fig. 6. UMAP visualization of iterative refinement of cell identity using data from mouse primary motor cortex. Each of panels **(a)-(f)** shows a successive step in iterative refinement. Panel **(g)** shows the result after the final dataset is added and quantile normalization is performed. Panel **(h)** shows the result from running online iNMF for 5 epochs using all of the datasets at once.

The results indicate that our approach is able to successfully incorporate each new RNA-seq dataset (**Fig. 6a-f**). Although no ground truth labels or published clustering assignments are available for this dataset, UMAP visualizations indicate that the structure of the datasets is iteratively refined with each successive dataset that is added. Even without performing quantile normalization (as previously described in LIGER [16]), we observed high alignment between datasets already seen and each new RNA dataset that we added (**Fig. 6a-f**). However, we found that the single-nucleus ATAC-seq data, the last dataset to arrive, did not align as well as the RNA datasets (**Fig. 6g**). We reasoned that this may be because each cell is seen only once in this setting. Consistent with this hypothesis, we observed much better dataset alignment of the snATAC-seq data when we sampled mini-batches from the whole dataset for 5 epochs (**Fig. 6h**). This suggests that incorporating unseen datasets can prove challenging when the new datasets are very different from those seen so far. Additionally, we visualized the shared metagene values W to see how they change during training. The metagene values change with each new dataset that is incorporated, indicating refinement in the gene signatures defining each cell type (**Fig. S2**). The magnitude of these changes decreases markedly with training, indicating that the signatures stabilize as training progresses. Overall, we conclude that the iterative refinement capability of online iNMF shows great promise on this real-world cell atlas dataset.

4 Discussion

In summary, our online iNMF algorithm achieves faster convergence on the objective than previously developed batch algorithms such as ANLS and Multiplicative updating rules. Although the performance of HALS on minimizing the objective is comparable to the proposed method, it requires the input data to be loaded in the memory. On the contrary, by specifying a reasonable mini-batch size, the online algorithm can be implemented with fixed memory usage regardless of the size of the input datasets, which are saved on the disk. This characteristic allows users to carry out analyses on a regular laptop without worrying about the size of the data. Additionally, online iNMF can iteratively refine cellular identity by processing new data

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

points either from the same type of datasets or different ones, as we demonstrated on a cell atlas of the mouse primary motor cortex. Future effort is needed for improving the performance of online iNMF when integrating a dissimilar type of dataset (such as snATAC-seq) on top of the previous integrative analysis result.

References

1. Barkas, N., Petukhov, V., Nikolaeva, D., Lozinsky, Y., Demharter, S., Khodosevich, K., Kharchenko, P.V.: Joint analysis of heterogeneous single-cell RNA-seq dataset collections. Nat. Methods **16**(8), 695–698 (Aug 2019)
2. Bottou, L., Bousquet, O.: The tradeoffs of large scale learning. In: Platt, J.C., Koller, D., Singer, Y., Roweis, S.T. (eds.) Advances in Neural Information Processing Systems 20, pp. 161–168. Curran Associates, Inc. (2008)
3. Cusanovich, D.A., Hill, A.J., Aghamirzaie, D., Daza, R.M., Pliner, H.A., Berletch, J.B., Filippova, G.N., Huang, X., Christiansen, L., DeWitt, W.S., Lee, C., Regalado, S.G., Read, D.F., Steemers, F.J., Disteche, C.M., Trapnell, C., Shendure, J.: A Single-Cell atlas of in vivo mammalian chromatin accessibility. Cell **174**(5), 1309–1324.e18 (Aug 2018)
4. Ecker, J.R., Geschwind, D.H., Kriegstein, A.R., Ngai, J., Osten, P., Polioudakis, D., Regev, A., Sestan, N., Wickersham, I.R., Zeng, H.: The BRAIN initiative cell census consortium: Lessons learned toward generating a comprehensive brain cell atlas. Neuron **96**(3), 542–557 (Nov 2017)
5. Fischer, B., Pau, G., Smith, M.: rhdf5-HDF5 interface for R. R# Package Version; RCoreTeam: Vienna, Austria **2** (2015)
6. Hie, B., Bryson, B., Berger, B.: Efficient integration of heterogeneous single-cell transcriptomes using scanorama. Nat. Biotechnol. **37**(6), 685–691 (Jun 2019)
7. HuBMAP Consortium: The human body at cellular resolution: the NIH human biomolecular atlas program. Nature **574**(7777), 187–192 (Oct 2019)
8. Kim, J., He, Y., Park, H.: Algorithms for nonnegative matrix and tensor factorizations: a unified view based on block coordinate descent framework. J. Global Optimiz. **58**(2), 285–319 (Feb 2014)
9. Kim, J., Park, H.: Fast nonnegative matrix factorization: An Active-Set-Like method and comparisons. SIAM J. Sci. Comput. **33**(6), 3261–3281 (Nov 2011)
10. Lee, D.D., Seung, H.S.: Algorithms for non-negative matrix factorization. In: Leen, T.K., Dietterich, T.G., Tresp, V. (eds.) Advances in Neural Information Processing Systems 13, pp. 556–562. MIT Press (2001)
11. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online learning for matrix factorization and sparse coding. J. Mach. Learn. Res. **11**(Jan), 19–60 (2010)
12. Moffitt, J.R., Bambah-Mukku, D., Eichhorn, S.W., Vaughn, E., Shekhar, K., Perez, J.D., Rubinstein, N.D., Hao, J., Regev, A., Dulac, C., Zhuang, X.: Molecular, spatial, and functional single-cell profiling of the hypothalamic preoptic region. Science **362**(6416) (Nov 2018)
13. Regev, A., Teichmann, S.A., Lander, E.S., Amit, I., Benoist, C., Birney, E., Bodenmiller, B., Campbell, P., Carninci, P., Clatworthy, M., Clevers, H., Deplancke, B., Dunham, I., Eberwine, J., Eils, R., Enard, W., Farmer, A., Fugger, L., Göttgens, B., Hacohen, N., Haniffa, M., Hemberg, M., Kim, S., Klenerman, P., Kriegstein, A., Lein, E., Linnarsson, S., Lundberg, E., Lundeberg, J., Majumder, P., Marioni, J.C., Merad, M., Mhlanga, M., Nawijn, M., Netea, M., Nolan, G., Pe'er, D., Phillipakis, A., Ponting, C.P., Quake, S., Reik, W., Rozenblatt-Rosen, O., Sanes, J., Satija, R., Schumacher, T.N., Shalek, A., Shapiro, E., Sharma, P., Shin, J.W., Stegle, O., Stratton, M., Stubbington, M.J.T., Theis, F.J., Uhlen, M., van Oudenaarden, A., Wagner, A., Watt, F., Weissman, J., Wold, B., Xavier, R., Yosef, N., Human Cell Atlas Meeting Participants: The human cell atlas. Elife **6** (Dec 2017)
14. Saunders, A., Macosko, E.Z., Wysoker, A., Goldman, M., Krienen, F.M., de Rivera, H., Bien, E., Baum, M., Bortolin, L., Wang, S., Goeva, A., Nemesh, J., Kamitaki, N., Brumbaugh, S., Kulp, D., McCarroll, S.A.: Molecular diversity and specializations among the cells of the adult mouse brain. Cell **174**(4), 1015–1030.e16 (Aug 2018)
15. Stuart, T., Satija, R.: Integrative single-cell analysis. Nat. Rev. Genet. **20**(5), 257–272 (May 2019)
16. Welch, J.D., Kozareva, V., Ferreira, A., Vanderburg, C., Martin, C., Macosko, E.Z.: Single-Cell multi-omic integration compares and contrasts features of brain cell identity. Cell **177**(7), 1873–1887.e17 (Jun 2019)
17. Yang, Z., Michailidis, G.: A non-negative matrix factorization method for detecting modules in heterogeneous omics multi-modal data. Bioinformatics **32**(1), 1–8 (Jan 2016)
18. Ye, Z., Sarkar, C.A.: Towards a quantitative understanding of cell identity. Trends Cell Biol. **28**(12), 1030–1048 (Dec 2018)
19. Zeisel, A., Hochgerner, H., Lönnerberg, P., Johnsson, A., Memic, F., van der Zwan, J., Häring, M., Braun, E., Borm, L.E., La Manno, G., Codeluppi, S., Furlan, A., Lee, K., Skene, N., Harris, K.D., Hjerling-Leffler, J., Arenas, E., Ernfors, P., Marklund, U., Linnarsson, S.: Molecular architecture of the mouse nervous system. Cell **174**(4), 999–1014.e22 (Aug 2018)

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

Supplementary

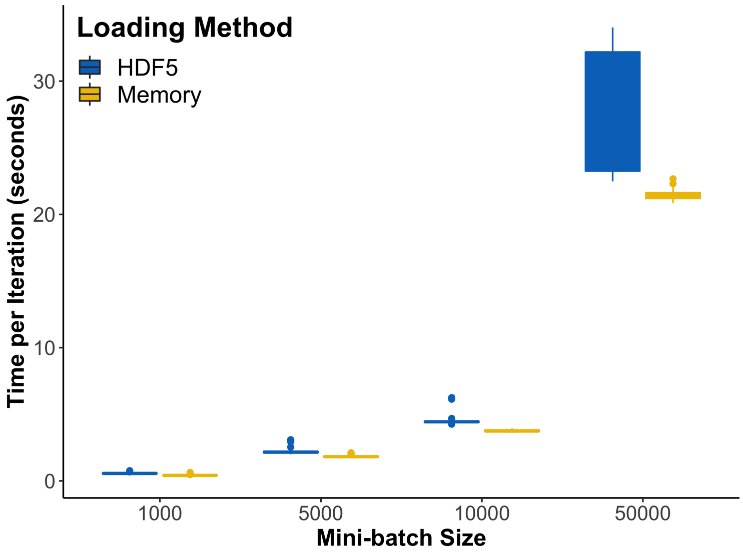

Fig. S1. The effect of data loading method on the implementation of online iNMF. In this study, each chunk in HDF5 files stores 1000 samples. Pulling data from the disk does not add significant overhead compared to loading the data from memory, as long as the size of the mini-batch size is close to the specified chunk size.

bioRxiv preprint doi: https://doi.org/10.1101/2020.01.16.909861; this version posted January 17, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

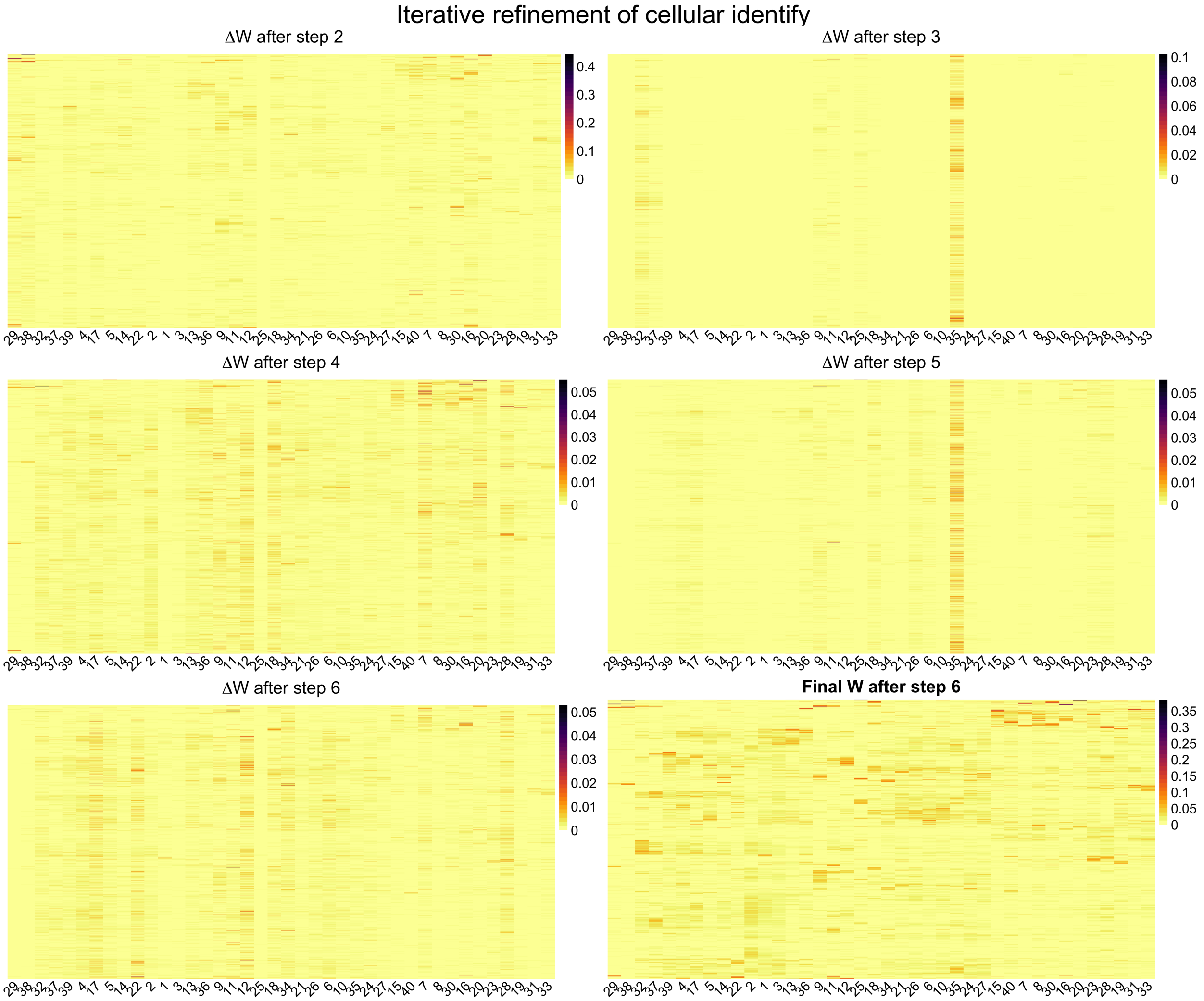

Fig. S2. Shared metagenes W are iteratively refined using online iNMF. Heatmaps showing the shared metagenes W after each new dataset is incorporated (see Fig. 6). Columns of W after each new data are normalized to unit norm. The value of each gene in each column represents its relative contribution to this metagene. ΔW is the absolute difference before and after the update upon arrival of new dataset.