A hybrid deep clustering approach for robust cell type profiling using single-cell RNA-seq data

SUHAS SRINIVASAN,[1] ANASTASIA LESHCHYK,[2] NATHAN T. JOHNSON,[3,4] and DMITRY KORKIN[1,2,5]

[1]Data Science Program, Worcester Polytechnic Institute, Worcester, Massachusetts 01609, USA
[2]Bioinformatics and Computational Biology Program, Worcester Polytechnic Institute, Worcester, Massachusetts 01609, USA
[3]Laboratory of Systems Pharmacology, Harvard Program in Therapeutic Science, Harvard Medical School, Boston, Massachusetts 02115, USA
[4]Breast Tumor Immunology Laboratory, Dana Farber Cancer Institute, Boston, Massachusetts 02215, USA
[5]Department of Computer Science, Worcester Polytechnic Institute, Worcester, Massachusetts 01609, USA

ABSTRACT

Single-cell RNA sequencing (scRNA-seq) is a recent technology that enables fine-grained discovery of cellular subtypes and specific cell states. Analysis of scRNA-seq data routinely involves machine learning methods, such as feature learning, clustering, and classification, to assist in uncovering novel information from scRNA-seq data. However, current methods are not well suited to deal with the substantial amount of noise that is created by the experiments or the variation that occurs due to differences in the cells of the same type. To address this, we developed a new hybrid approach, deep unsupervised single-cell clustering (DUSC), which integrates feature generation based on a deep learning architecture by using a new technique to estimate the number of latent features, with a model-based clustering algorithm, to find a compact and informative representation of the single-cell transcriptomic data generating robust clusters. We also include a technique to estimate an efficient number of latent features in the deep learning model. Our method outperforms both classical and state-of-the-art feature learning and clustering methods, approaching the accuracy of supervised learning. We applied DUSC to a single-cell transcriptomics data set obtained from a triple-negative breast cancer tumor to identify potential cancer subclones accentuated by copy-number variation and investigate the role of clonal heterogeneity. Our method is freely available to the community and will hopefully facilitate our understanding of the cellular atlas of living organisms as well as provide the means to improve patient diagnostics and treatment.

Keywords: scRNA-seq; clustering; machine learning; single-cell; transcriptomics

INTRODUCTION

Despite the centuries of research, our knowledge of the cellular architecture of human tissues and organs is still very limited. Microscopy has been conventionally used as a fundamental method to discover novel cell types, study cell function and cell differentiation states through staining and image analysis (Carpenter et al. 2006). However, this approach is not able to identify heterogeneous subpopulations of cells, which might look similar, but perform different functions. Recent developments in single-cell RNA sequencing (scRNA-seq) have enabled harvesting the gene expression data from a wide range of tissue types, cell types, and cell development stages, allowing for a fine-grained discovery of cellular subtypes and specific cell states (Tanay and Regev 2017). Single-cell RNA sequencing data have played a critical role in the recent discoveries of new cell types in the human brain (Darmanis et al. 2015), gut (Grün et al. 2015), lungs (Treutlein et al. 2014), and immune system (Villani et al. 2017), as well as in determining cellular heterogeneity in cancerous tumors, which could help improve prognosis and therapy (Patel et al. 2014; Tirosh et al. 2016). Single-cell experiments produce data sets that have three main characteristics of big data: volume (number of samples and number of transcripts per each sample), variety (types of tissues and cells), and veracity (missing data, noise, and dropout events) (Brennecke et al. 2013). Recently emerging large initiatives, such as the Human Cell Atlas (Regev et al. 2017), rely on single-cell sequencing technologies at an unprecedented scale, and have generated data sets obtained from hundreds of thousands and even

Corresponding authors: ssrinivasan@wpi.edu, dkorkin@wpi.edu
Article is online at http://www.rnajournal.org/cgi/doi/10.1261/rna.074427.119.

millions of cells. The high numbers of cells, in turn, allow to account for data variability due to cellular heterogeneity and different cell cycle stages. As a result, there is a critical need to automate the processing and analysis of scRNA-seq data. For instance, for the analysis of large transcriptomics data sets, computational methods are frequently employed that find patterns associated with the cellular heterogeneity or cellular development, and group cells according to these patterns.

If one assumes that all cellular types or stages extractable from a single-cell transcriptomics experiment have been previously identified, it is possible to apply a supervised learning classifier. The supervised learning methods are trained on the data extracted from the individual cells whose types are known. The previously developed approaches for supervised cell type classification have leveraged data from image-based screens (Jones et al. 2008) and flow cytometry experiments (Chen et al. 2016). There has also been a recent development of supervised classifiers for single-cell transcriptomic data (Demšar et al. 2013), including methods that implement neural networks trained on a combination of transcriptomic data and protein interaction data (Lin et al. 2017). While a supervised learning approach is expected to be more accurate in identifying the previously observed cellular types, its main disadvantage is the limited capacity in discovering new cell types or identifying the previously known cell types whose RNA-seq profiles differ from the ones observed in the training set.

Another popular technique for scRNA-seq data analysis is unsupervised learning, or clustering. In this approach, no training data are provided. Instead, the algorithm looks to uncover intrinsic similarities shared between cells of the same type and not shared between cells of different types (Menon 2018). Often, clustering analysis is coupled with a feature learning method to filter out thousands of unimportant features extracted from the scRNA-seq data. In a recent study, the principal component analysis (PCA) approach was used on gene expression data from scRNA-seq experiments profiling neuronal cells (Usoskin et al. 2015). With the goal of identifying useful gene markers that underlie specific cell types in the dorsal root ganglion of mice, 11 distinct cellular clusters were discovered. Other approaches have also adopted this strategy of combining a simple, but efficient feature learning method with a clustering algorithm, to detect groups of cells that could be of different subtypes or at different stages in cellular development (Huang et al. 2008; Wang et al. 2017). One challenge faced by such an approach is due to scRNA-seq data exhibiting complex high-dimensional structure, and such complexity cannot be accurately captured by fewer dimensions when using simple linear feature learning methods.

A nonlinear method frequently used in scRNA-seq data analysis for clustering and visualization is t-distributed stochastic neighbor embedding (t-SNE) (Maaten and Hinton 2008). While t-SNE can preserve the local clusters, preserving the global hierarchical structure of clusters is often problematic (Wattenberg et al. 2016). Furthermore, the conventional feature learning methods may not be well suited for scRNA-seq experiments that have considerable amount of both experimental and biological noise or the occurrence of dropout events (Kolodziejczyk et al. 2015; Stegle et al. 2015). To address this problem, two recent methods have been introduced, pcaReduce (Yau 2016) and SIMLR (Wang et al. 2017). pcaReduce integrates an agglomerative hierarchical clustering with PCA to generate a hierarchy where the cluster similarity is measured in subspaces of gradually decreasing dimensionalities. The other approach, SIMLR, learns different cell-to-cell distances through by analyzing the gene expression matrix; it then performs feature learning, clustering, and visualization. The computational complexity of the denoising technique in SIMLR prevents its application on the large data sets. Therefore, a different pipeline is used to handle large data, where the computed similarity measure is approximated, while the diffusion approach to reduce the effects of noise is not used. In addition to the dimension reduction methods, K-means is a popular clustering method used in single-cell transcriptomics analysis. While being arguably the most popular divisive clustering algorithm it has several limitations (Jain 2010; Celebi et al. 2013).

In this work, we looked at the possibility to leverage an unsupervised deep learning approach (Baldi 2012) to handle the complexities of scRNA-seq data and overcome the above limitations of the current feature learning methods. It has been theoretically shown that the multilayer feed-forward artificial neural networks, with an arbitrary squashing function and sufficient number of hidden units (latent features) are universal approximators (Hornik et al. 1989) capable of performing dimensionality reduction (Cybenko 1989). A recently published method, scVI, implemented unsupervised neural networks to overcome specific problems of the library size and batch effects during single-cell sequencing (Lopez et al. 2018). However, scVI underfits the data when the number of sequenced genes exceeds the number of sampled cells. Therefore such phenomenon is likely to be observed in experiments that sample a few thousand cells while quantifying tens of thousands of genes. Here, we propose the use of denoising autoencoder (DAE) (Vincent et al. 2008), an unsupervised deep learning architecture that has previously proven successful for several image classification (Vincent et al. 2010) and speech recognition (Lu et al. 2013) tasks, by reducing noise. DAEs are different from other deep learning architectures in their ability to handle noisy data and construct robust features. We add a novel extension to the DAE called denoising autoencoder with neuronal approximator (DAWN), which decides the number of latent features that are required to represent efficiently any given data set. To overcome the limitations of K-means clustering, we integrate our DAWN approach with the expectation-maximization (EM) clustering algorithm (Do and Batzoglou 2008). We use the features generated by DAWN as an input to the EM clustering algorithm and show that our hybrid approach has higher accuracy when compared to the traditional feature learning and clustering algorithms discussed above. In particular, we can recover clusters from the original study without using any knowledge about the tissue-specific or cell type specific markers. As a result, our hybrid approach, deep unsupervised single-cell clustering (DUSC), helps to overcome the noise in the data, captures features that are representative of the true patterns, and improves the clustering accuracy. In an application to triple-negative breast cancer, DUSC clustering results were integrated with copy-number variation data to understand the role of clonal evolution, and specifically clonal heterogeneity, in triple-negative breast cancer.

RESULTS

Data sets

For the assessment of our approach, we chose five single-cell RNA-seq data sets (Fig. 1): embryonic data set-1 (E1), embryonic data set-2 (E2), sensory neurons (SN), mouse cortex (MC), and malignant melanoma (MM) (Biase et al. 2014; Usoskin et al. 2015; Zeisel et al. 2015; Goolam et al. 2016; Tirosh et al. 2016). These data sets were selected to represent areas where scRNA-seq technology had a significant impact (Shapiro et al. 2013). The areas included embryonic development, cellular heterogeneity in the nervous system and cellular heterogeneity in a disease (cancer). The data sets originated from a model organism (mouse) and human. In total, 8055 single-cell samples were analyzed (Fig. 1A). All data sets were downloaded in a quantified format from publicly available sources listed in the studies (Langmead et al. 2009; Li and Dewey 2011; Trapnell et al. 2012; Anders et al. 2015). To test the scalability and robustness of the proposed method, the data sets were chosen such that they exhibited variability across multiple parameters: (a) number of sequenced cells (from 56 cells to 4139 cells), (b) number of genes quantified (from ∼19,000 to ∼41,000 genes), (c) different sequencing and quantification pipelines, (d) cellular heterogeneity during development or disease, and (e) varying cellular hierarchy and number of cell types (with one to three levels of hierarchy and five to 12 cell types/subtypes). Many of the cellular types include subpopulations corresponding to the cellular subtypes (Fig. 1B). Specifically, the cellular subtypes in SN and MM data sets are hierarchically organized; SN has a three-level hierarchy, while MM has two levels. The distributions of number of genes quantified per cell varied significantly: for E1 and E2, the distribution was centered around ∼13,000 genes, and for SN, MC and MM the distributions were centered around ∼4000 genes (Supplemental Fig. 1). Using the normalized Shannon entropy, H_{NORM}, we found that the distribution balance of samples across cell types also varied, with the first level of SN being the most balanced and the second level of MM being the most unbalanced sets, correspondingly (see also Effects of data balance on accuracy subsection).

Comparison with clustering and classification algorithms

We first evaluated the overall performance of our clustering approach, DUSC, and its most critical part, a new feature learning method DAWN. To test if DUSC could improve the discovery of cell type clusters in scRNA-seq data, we compared the clustering of our hybrid approach with (i) clustering that had no feature selection, and (ii) the same clustering methods that now employed the classical and state-of-the-art unsupervised feature learning methods. We expected that clustering with no feature selection would perform the worst, thus establishing a baseline for comparative analysis. We also assessed the classification accuracy of Random Forest (RF), a state-of-the-art supervised learning algorithm. The latter approach represents the best-case scenario when all cell types are known.

For the E1 data set, all methods except KM recovered the clusters with similarly high accuracies. As expected, the small sample size and a few cell types to consider made the clustering a simpler task (Fig. 2A; Supplemental Table 2). When processing E2, DUSC had the highest accuracy among all methods, and while there was only a small accuracy drop for RF, both KM and EM experienced significant losses in accuracy. The drop in performance on the E2 data set, which had the same number of cell types as E1, could be explained by the fact that both the sample size and feature size approximately doubled, therefore quadrupling the problem size and making it a harder computational challenge. For the main hierarchy level in SN (SN-i), the sample size was 731, making it a larger search space, but with only five major cell types. Here, DUSC performed well ($Acc = 0.9$) and was closely followed by RF, while KM and EM performed poorly (accuracies were 0.88, 0.53, and 0.62 correspondingly). For the second level of SN subtypes (SN-ii), the sample size was still 731, but the number of cell subtypes increased to nine, thus resulting in a smaller sample size for each cell subtype (Fig. 1B) and smaller feature differences between the subtypes. As a result, it was not surprising that all methods experienced a drop in their performance, with RF performing best ($Acc = 0.74$) and DUSC being the first among the unsupervised methods, closely behind RF ($Acc = 0.69$). When considering the lowest level of SN, SN-iii, with the number of subtypes being 12 and cell cluster sizes ranging from 12 to 233, we noticed that RF and DUSC both have similar accuracy ($Acc = 0.71$), while KM and EM still performed poorly

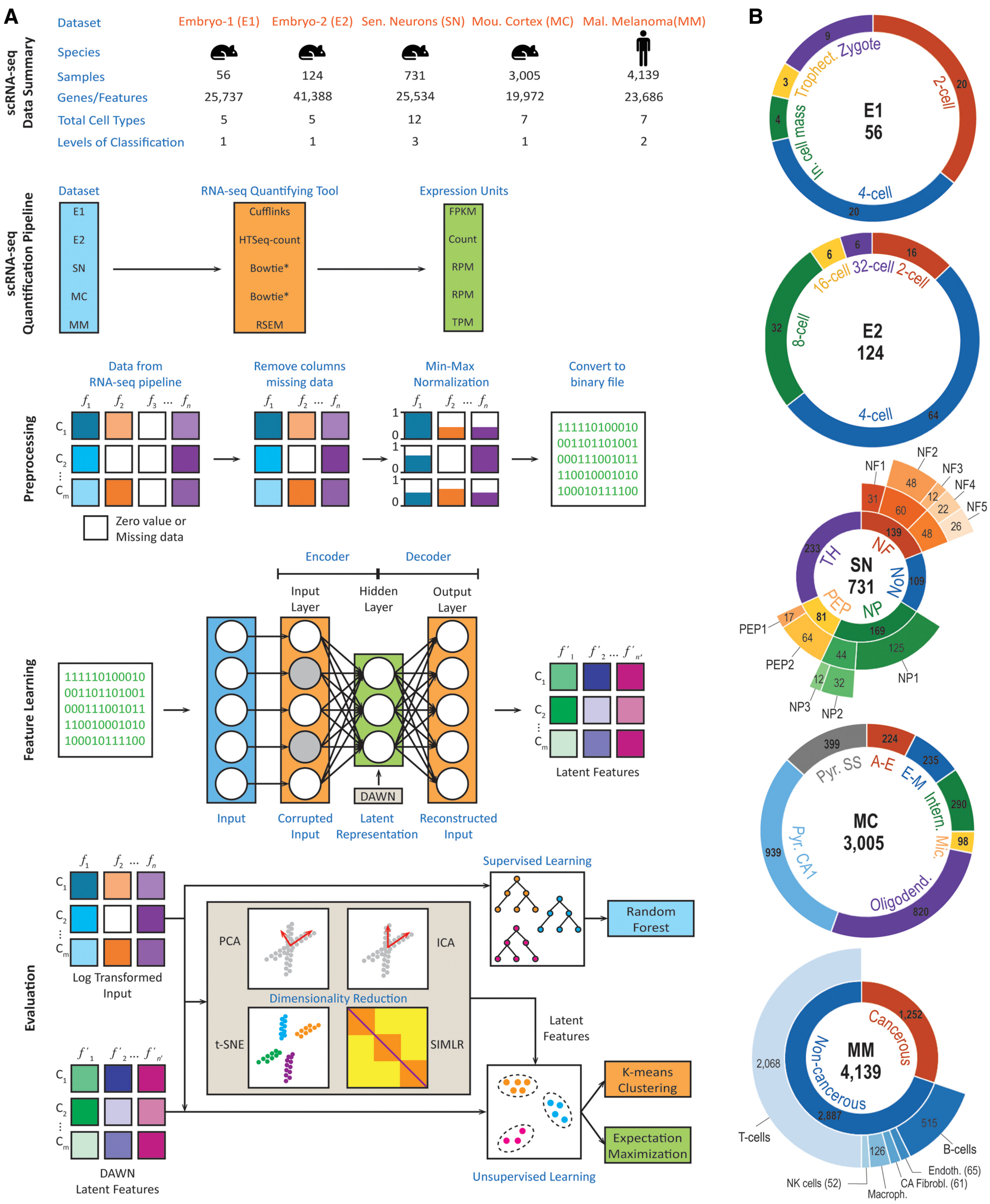

FIGURE 1. Overview of DUSC approach. (*A*) Basic stages of the deep clustering method and overview of the five data sets to which it was applied. Each of the five data sets was processed using a different RNA-seq quantification tool, with the data quantified in different expression units. During the evaluation, our approach was compared against the standard clustering methods as well as their enhanced versions using four feature learning approaches. (*B*) The detailed description of the five data sets: embryonic data set-1 (E1), embryonic data set-2 (E2), sensory neurons (SN), mouse cortex (MC), and malignant melanoma (MM), their multilevel hierarchical organizations, and subpopulation distribution. The total number of cell samples is depicted in the *center* of each sunburst chart.

(*Acc* = 0.46 and 0.40, correspondingly). We note that for all evaluations based on the subtypes of either SN-i (i.e., SN-ii and SN-iii levels) or MM-i (i.e., MM-ii level), we did not filter out the major cell clusters from the higher levels to recursively process subclusters of the lower levels. This is because, the cellular hierarchy was not known a priori when analyzing a novel data set and its structure could only be discovered after the recursive analysis of

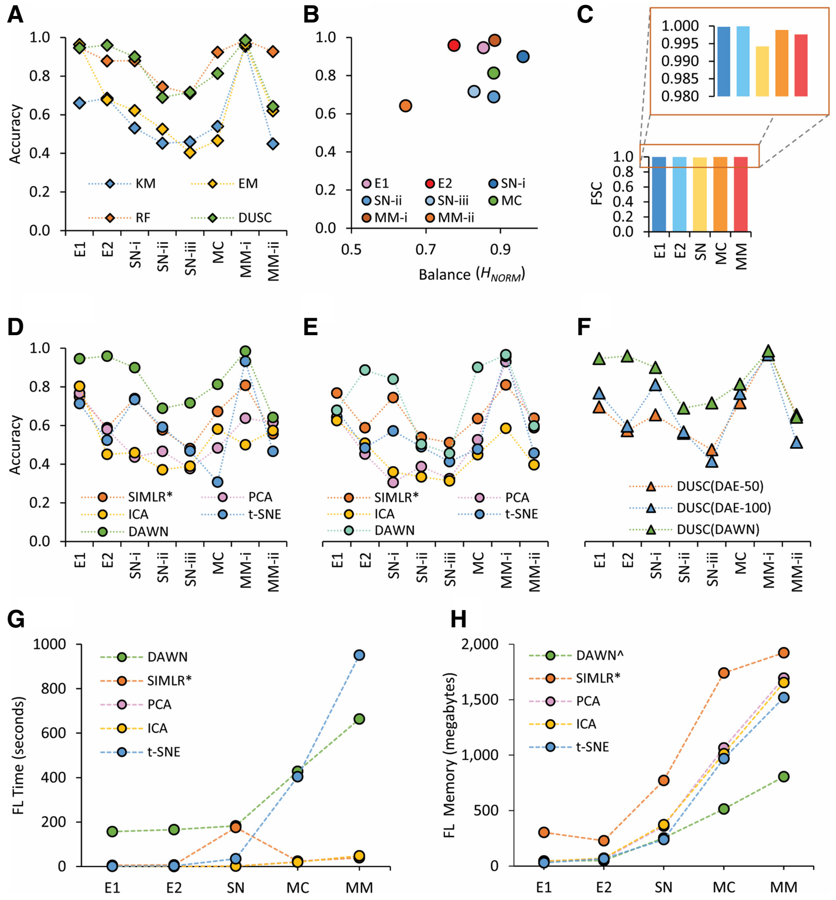

FIGURE 2. Comparative assessment of DUSC. The methods considered in this figure include: K-means (KM), expectation-maximization (EM), random forest (RF), principle component analysis (PCA), independent component analysis (ICA), t-distributed stochastic neighbor embedding (t-SNE), single-cell interpretation via multikernel learning (SIMLR), and deep unsupervised single-cell clustering (DUSC). The data sets used in the figure include embryonic data set-1 (E1), embryonic data set-2 (E2), sensory neurons (SN-i, SN-ii, and SN-iii correspond to the subpopulations at the first, second, and third levels of hierarchy, respectively), mouse cortex (MC), and malignant melanoma (MM-i and MM-ii correspond to the subpopulations at the first and second levels of hierarchy, respectively). (*A*) Overall performance of DUSC in comparison with two clustering approaches, KM and EM, and a state-of-the-art supervised learning approach, RF. DUSC outperforms both clustering methods, and its accuracy is comparable with that of the supervised classifier. (*B*) The performance accuracy by DUSC is affected by the distribution balance of the subpopulations forming the data set: applying DUSC to the more unbalanced data set results in lower accuracy and vice versa. (*C*) Feature space compressed (FSC) calculated for all five data sets. (*D*) The performance of EM clustering combined with DAWN and other feature learning methods ([*] use of the large scale implementation of SIMLR for MC and MM data sets). (*E*) The performance of K-means clustering combined with DAWN and other feature learning methods; DAWN shows a significantly greater improvement in both *B* and *D*. (*F*) The clustering performance of DUSC using DAWN, versus using two manual configurations of the standard DAE (50 and 100 neurons). DAWN performs significantly better than the manual configurations and with fewer hidden neurons. (*G*) The execution time for all the methods during feature learning (FL), considering that DAWN is a deep learning method; it has communication cost and I/O cost from CPU to GPU. (*H*) Memory used during feature learning by all methods, DAWN uses the least amount of memory (^ total of System and GPU memories used by DAWN).

subclusters that, in turn, required multiple iterations. Here, we generated clusters only through a single pass of our processing pipeline.

The size of the next data set, MC, was several folds greater than of the previous two, and in this case, RF had the best accuracy (*Acc* = 0.92) and our unsupervised method DUSC had a significantly higher accuracy (*Acc* = 0.81) than KM and EM (*Acc* = 0.54 and 0.57, correspondingly). Lastly, in the final data set, MM, we initially tried to find only two clusters of cancerous and noncancerous cells, and this binary problem with two very different cell types and approximately the same cluster sizes was unsurprisingly an easy challenge. Thus, all methods perform very well with the accuracies above 0.95, but DUSC still lead the unsupervised algorithms with the same accuracy as RF (*Acc* = 0.99). When the subtypes of noncancerous cells had to be considered as separate groups along with cancerous cells (MM-ii), the complexity of the problem increased, and all unsupervised algorithms experienced a significant drop in performance when compared to RF (*Acc* = 0.93), with DUSC still achieving the best result (*Acc* = 0.64).

In summary, the assessment on all four data sets demonstrated that DUSC performed better than the KM and EM clustering algorithms and in many instances by large margins. Even more importantly, DUSC had comparable performance with Random Forest supervised approach in many cases, and in some cases even outperformed it.

Effects of data balance on accuracy

The data balance metric introduced in this work allowed us to find how the data complexity and imbalance affected the performance of DUSC (Fig. 2B). Indeed, for all unsupervised methods, including DUSC, the clustering accuracy was impacted by the data complexity (Supplemental Fig. 3; Supplemental Table 3). This was especially evident in the cases of SN-ii and MM-ii data sets, where the number of cell types increased compared to the original data sets, SN-i and MM-i, respectively. The higher number of clusters, in turn, leads to a greater variation in cluster sizes, and in the same time, a lower number of differentiating features. Here, we observed that both data balance and clustering accuracy decreased when moving down the cell type hierarchy in SN and MM data sets.

Feature compression

To study the information content of the initially sparse feature space, another metric, feature space compressed (FSC), was used for DUSC (Fig. 2C). With the combination of the preprocessing stage and the neuronal approximation, DAWN compressed at least 0.994 (99.4%) of the original feature space reaching 0.998 (99.8%) for four out of five data sets (Supplemental Table 7). The maximum compression occurred for E2, where 41,388 of the original features were cleaned and compressed to just three latent features resulting in FSC of 99.99%. The data compression capacity of DAWN could also be a useful tool for storing cell type critical information in large scRNA-seq studies. For instance, the size of an average data set obtained from a single study could be reduced from 1 gigabyte to only 5 megabytes using DAWN. We note that the highly efficient compression occurred simultaneously when improving the clustering performance.

Assessment of unsupervised feature learning algorithms and their impact on clustering

Next, we compared the performance of DUSC against the four feature learning methods, SIMLR, PCA, ICA, and t-SNE. Since DUSC is a hybrid approach that combined a new feature learning method (DAWN) and a clustering algorithm (EM), for a fair comparison, we also paired the other four feature learning methods with the EM clustering method (Fig. 2D; Supplemental Table 4). The results showed that the previously observed effects of sample size, number of cell clusters, and number of important features on DAWN's performance also affected the other four methods. For the easier data sets w.r.t. the above criteria, such as E1 and MM-i, all the algorithms had an accuracy greater than 0.7, with DUSC reaching significantly higher accuracies of 0.95 and 0.99, respectively. Interestingly, when more complex problems were considered, that is, E2 and MC, we noticed a significant performance drop for all algorithms; however, when compared to SIMLR, the best performing method of the four currently existing ones, DUSC still clustered E2 more accurately ($Acc = 0.96$) and also had a 14% higher accuracy ($Acc = 0.81$) on the MC data set. We also recall that SIMLR was used in a less challenging setup when the true number of clusters was provided as an input. Overall, DUSC had the better accuracy across all data sets, compared to all other unsupervised feature learning algorithms. The results also suggest that any scRNA-seq analysis tools that utilize PCA or ICA, such as Seurat and Monocle, respectively, might not be capturing the single-cell information optimally. Furthermore, the features extracted from PCA or ICA algorithms when used for K-means clustering to obtain the final cell clusters might not be accurate; the same features when used for cell cluster visualization through t-SNE suffer from similar problems.

Assessment of the contributing factors in the hybrid approach

We then hypothesized that between the feature learning (DAWN) and clustering (EM) components of our approach, DAWN was contributing more to the clustering accuracy. To determine the impact of DAWN, we paired it as well as the four other feature learning methods with K-Means clustering. We found that DAWN either exceeded the clustering accuracy of SIMLR, (for E2, SN-i, MC and MM-i) or closely matched it in the other cases (Fig. 2E; Supplemental Table 5). The other methods, PCA, ICA and t-SNE, had significantly lower accuracies for the majority of the tasks. The findings suggested that DAWN provided the key contribution toward improving the clustering accuracy. A consistent trend that was observed across all methods (Fig. 2D,E) was that for SN and MM data sets, the accuracy decreased as the feature learning and clustering methods traversed the cell type hierarchies. The smaller differences in the numbers of uniquely expressed genes together with a larger set of common genes across the cellular subtypes, compared to the main cellular types, made it a more challenging problem for feature learning and clustering.

Assessment of neuronal approximation

To further assess the benefits of our novel neuronal approximation in DAWN, we compared it with the standard DAE. We created two configurations of the standard DAE, by choosing the number for the hidden units to be 50 and 100, respectively. All other aspects of the DUSC approach were kept intact, and the end-to-end analysis was repeated for DAE-50 and DAE-100. The clustering results showed that DAWN outperformed the standard DAE configurations in six cases and had extremely similar performance in the remaining two cases (Fig. 2F; Supplemental Table 8). This analysis showed that the automated technique to set the number of hidden units was superior to the manual value selection for this important parameter. The structural patterns discovered in the DAWN features are compared against PCA (Supplemental Fig. 5); the optimization of the latent features that are dependent on the number of training epochs is also analyzed (Supplemental Fig. 6). The results showed the capacity of DUSC to be a fully automated clustering approach, which can be applied to small data sets (E1, 56 cells) as well as large data sets (MM, 4139 cells).

Computational performance

The execution time for each of the five feature learning methods was profiled across all five data sets. After loading the data from storage, all methods utilized CPU, except for DAWN, which used GPU for the actual feature learning. DAWN worked in a master/slave configuration, where the data load and preprocessing steps were performed on the CPU (master) and the CPU instructed the GPU (slave) to carry out the feature learning and send back the results. Due to this configuration, DAWN incurred additional time costs to move the data between the system memory and GPU memory and for the constant interprocess communication between the CPU and GPU. This trend in the execution time for DAWN (Fig. 2G) started higher than the other methods but remained almost flat for data sets E1 (157 s), E2 (166 s) and SN (183 s), and despite this additional cost, DAWN performed significantly faster than t-SNE for the largest data set. The execution time of PCA and ICA scaled well with the increasing data set size: they were the fastest methods. We then used the default implementation of SIMLR for data sets E1, E2, and SN data sets. For SN it took 176 sec, the highest among all other methods. Furthermore, when using the same implementation for MC, the execution time increased drastically to 4450 sec, which was attributed to reducing the noise and dropout effects in the data. Thus, as suggested by the authors of SIMLR, we used its large-scale implementation mode for MC and MM. The large-scale implementation employed different steps to process the data with a speed similar to PCA, but possibly sacrificing the quality of feature learning.

Additionally, we profiled memory usage during feature learning for all methods. The total used memory (System and GPU) was reported for DAWN, and we observed that DAWN had the lowest memory usage of all methods due to the added optimizations (Fig. 2H; Supplemental Table 9). PCA, ICA, and t-SNE all had similar memory profiles. SIMLR had the highest memory usage of all methods, considering the large-scale implementation for data sets MC and MM. However, when the default implementation of SIMLR was used, the memory footprint for the MC data set increased sharply to 8270 megabytes.

Cluster embedding and visualization

To illustrate the capacity of our approach to preserve the local structure of the data, we generated two-dimensional embeddings for the two largest data sets, MC and MM. Specifically, we applied the t-Distributed stochastic neighbor embedding (t-SNE) method to the latent features generated by DAWN and compared it to the four other feature learning methods. We considered the MC data set first because it was a complex data set with 3005 cells and seven cell types (Fig. 3A). When comparing the embeddings obtained from the original data and after applying the five feature learning methods, the embedding produced from the DAWN-generated latent features showed cell clusters that were the most clearly separated and had smooth elliptical boundaries.

To determine if the biological relationship between the clusters of related cell types could be reflected through the spatial relationship in the 2D embedding, we next created a hierarchical network overlay on the DAWN embedding using the cell type dendrogram obtained from the original study (Fig. 3B). The network topology revealed immediate connections between the clusters corresponding to the more similar cellular types. The more dissimilar clusters were not immediately connected; instead they were connected through the hierarchical nodes and edges in the network, as expected. The obtained network overlay indicated that DAWN preserved the relationships between the cell types during the learning process. The 20 latent features learned by DAWN on the MC data set were then analyzed using a heatmap representation (Fig. 3C), where the rows represent individual cells, and columns represent the latent features. The heatmap, where the cells were grouped by their types revealed the "block" structural patterns formed by the groups of features, showing that the latent features learned by our method could recover the intrinsic structure of the original data. The heatmap also shows the orchestrated work of hidden neurons to learn complementary patterns.

Finally, we obtained the two-dimensional embeddings for the MM data set (Fig. 3D), another complex data set with a high variation in the cluster size (52–2068 cells). We found that DAWN was the only feature learning method capable of producing compact and well-separated clusters, where the two major cell types, that is, cancerous and noncancerous cells, were separated with no overlap. The subtypes of noncancerous cells were also well-separated, with the only exception being natural killer (NK) cells (52 cells), which partially overlapped with the largest cell cluster of T cells (2068 cells). This overlap could be explained by the disproportionately small size of the NK cluster and the substantial similarity between NK cells and T cells (Narni-Mancinelli et al. 2011).

Integrating scRNA-seq clustering with CNV data suggests the role of clonal heterogeneity in triple-negative breast cancer

In the triple-negative breast cancer data set, we specifically selected the data of a patient (patient-39) who had a large tumor (9.5 cm), which using the original study manually inferred the presence of subclones, but did not identify the clones by a clustering approach (Karaayvaz et al. 2018).

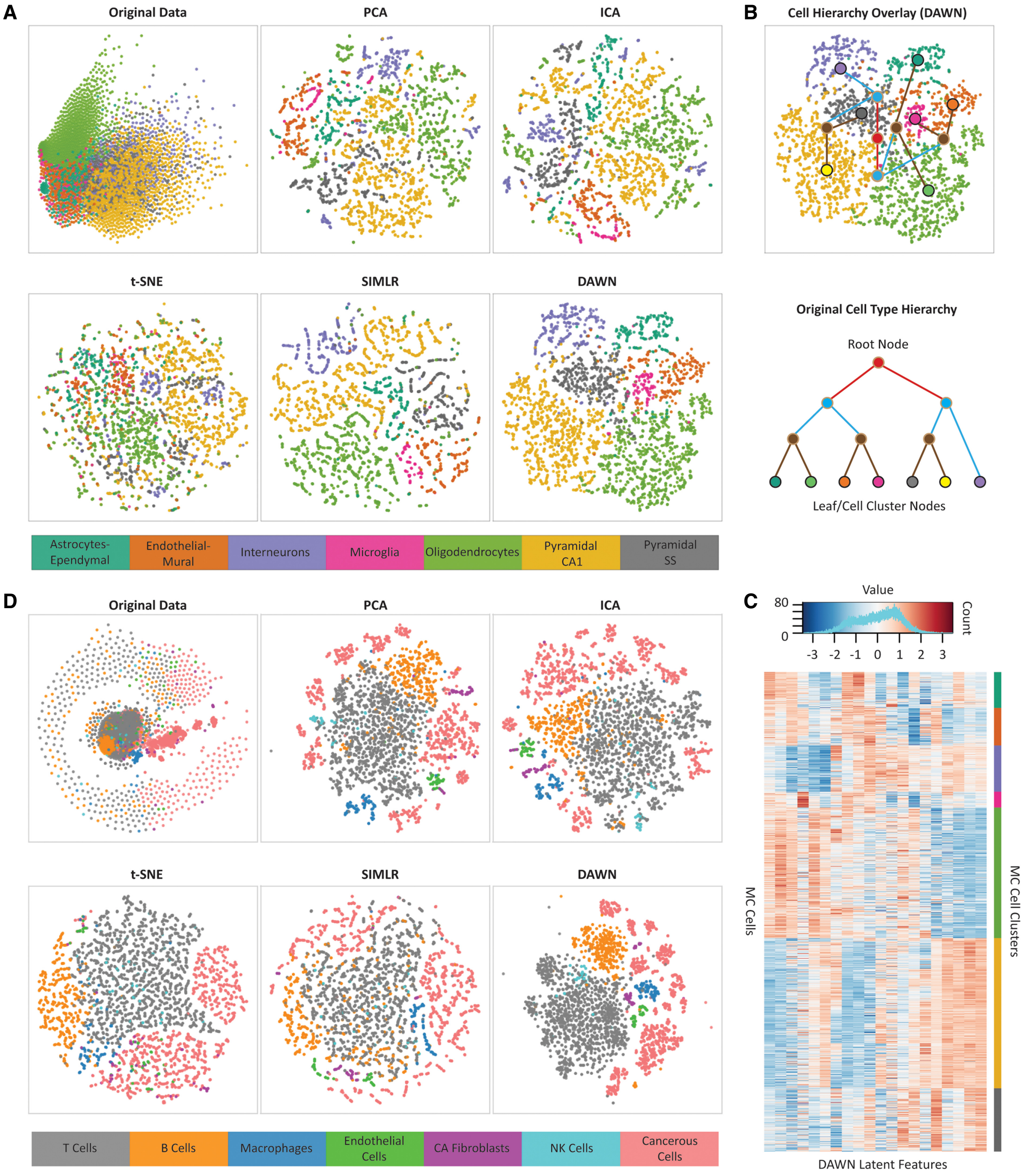

FIGURE 3. Analysis of clustering performances using visualization approaches. (*A*) Two-dimensional embedding of the mouse cortex (MC) data set in the original feature space compared with the embeddings of the same data set in the feature space generated by DAWN and four other feature learning methods, principle component analysis (PCA), independent component analysis (ICA), t-distributed stochastic neighbor embedding (t-SNE), and single-cell interpretation via multikernel learning (SIMLR). (*B*) Hierarchical clustering overlay (*top*) constructed from the two-dimensional embedding of the DAWN feature space. The hierarchy is created based on the proximities of mass centers of the obtained clusters. The obtained hierarchy is compared to that of biological cell types (*bottom*) extracted from the original study (Zeisel et al. 2015). The leaf nodes correspond to the original cell types, while the root and internal nodes correspond to the three other levels obtained through the agglomerative hierarchy. The two-dimensional embedding of the DAWN feature space can recover all but one of the defined relationships between the related cell types extracted from the literature. (*C*) The heatmap of the 20 latent features generated by DAWN on the MC data set, showing the block structure of the expression profiles of the individual cells grouped by the cell types (*bottom*). The values of the latent features corresponding to the weights in the hidden layer are distributed in [−3, 3] range (*top*). (*D*) Two-dimensional embedding of the malignant melanoma (MM) data set in the original feature space compared with the embeddings of the same data set in the feature spaces generated by DAWN and four other feature learning methods: PCA, ICA, t-SNE, and SIMLR.

Thus, to identify clones in an unbiased unsupervised manner, we first applied DUSC and identified three clusters. Generating a 2D embedding showed the structure and similarity of the three clusters (Fig. 4A). Specifically, it was observed that Clusters 1 and 2 were significantly different from each other, while Cluster 3 was more similar to Cluster 1 than to Cluster 2. Based on previous work (Tirosh et al. 2016), we annotated the embedding with G1/S and G2/M cycling status, revealing the cycling cells that were suspected to be malignant and that Cluster 1 contained a substantial proportion of such cells. We then dissected each cluster into the main four molecular subtypes of TNBC using Lehmann's classification (Fig. 4B), finding that all clusters had a high proportion of Basal Like-1/2 cells that were associated with cancer aggressiveness and poor prognosis, as well as Mesenchymal cells that were associated with relative chemoresistance (Park et al. 2018). To validate the DUSC clusters for clonal heterogeneity, the tumor data and cluster labels were used with the inferCNV tool along with the normal cells used as a reference. In Clusters 1 and 3, we found many genomic regions (Fig. 4C; Supplemental Fig. 4) which carried significant amplifications (e.g., chromosomes 1 and 2) and deletions (e.g., chromosomes 5 and 15). The significant CNV aberrations indicated the likely clonal heterogeneity. Next, for a more fine-grained analysis we perform differential gene expression analysis, identifying the top 100 genes for each cluster. When querying the top 100 genes against Disgenet (Piñero et al. 2016), we found 23 unique genes associated with breast cancer that were shared across all three clusters (Fig. 4D). The association of these genes to breast cancer was one of three types: biomarker, casual mutation, or genetic variation. We further looked at the expression pattern of these 23 genes across the clusters to find high gene activity. To do so, we $\log_2$-normalize the quantified expression to observe the log-fold change, finding that some genes are expressed five to 10 times greater than normal in Clusters 1 and 3 (Fig. 4E). Specifically, genes *ARF1*, *ALDOA*, *VIM*, *RPS6*, *PABPC1*, and *LDHB* were overexpressed and clustered together. These genes play roles in many critical biological processes (Svensson et al. 2018).

High expression of *ARF1* (ADP-ribosylation factor 1) had been demonstrated to be associated with an increased likelihood of metastatic breast cancer and was found to be a characteristic feature of triple-negative breast cancer (Schlienger et al. 2016). *ALDOA* (Aldolase, Fructose-Bisphosphate A) and *LDHB* (Lactate Dehydrogenase B) were genes that partook in the glycolytic process and were known to coexpress (Choobdar et al. 2019). *LDHB* was identified as an essential gene for tumor growth; it was up-regulated in TNBC and was identified as a potential target for personalized treatment (McCleland et al. 2012). In a recent study, *ADLOA* was identified to be overexpressed in melanoma and lung cancer, and such increases in the glycolysis pathway of the tumor were associated with immune therapy resistance (Cascone et al. 2018). *VIM* (Vimentin) was also a known marker for epithelial-mesenchymal transition and breast cancer stem cells suggested to be responsible for the metaplastic process (Jang et al. 2015). In summary, single-cell clustering coupled with CNV analysis and differential gene expression analysis was able to identify the clonal heterogeneity present in a patient, with the mutated and overexpressed genes of high relevance to TNBC.

DISCUSSION

In this work, we have presented DUSC, a new hybrid approach for accurate clustering of single-cell transcriptomics data. Rapid progress in the development of scRNA-seq technologies urges the advancement of accurate methods for analyzing single-cell transcriptomics data (Svensson et al. 2018). One of the first tasks for such analysis is extracting the common patterns shared between cell populations by clustering the cells together based on their expression profiles. The process of clustering, ideally, can help in answering two questions: (i) what is the biological reason for cells to be grouped (e.g., a shared cellular type), and (ii) what are the biological constituents found in the scRNA-seq data that determine the similarity between the cells from the same cluster (e.g., expression values for a set of the overexpressed genes). An important advantage of the clustering methods is their power to extract novel, previously unseen similarity patterns, which leads to the discovery of new cell types (Papalexi and Satija 2018), spatial cellular compartmentalization in disease and healthy tissues (Medaglia et al. 2017), subpopulations of cells from different developmental stages (Gong et al. 2018), and other cellular states. However, the clustering accuracy, despite being continuously tackled by the recent methods, has remained substantially lower when compared to the supervised learning, or classification, methods. Classification methods, in turn, are designed to handle data from the cellular subpopulations whose representatives have been used during the training stage, and therefore cannot identify novel subpopulations. Another question that has not been fully addressed is the robustness of the class definition based on the scRNA-seq data: Does a class defined by a certain supervised classifier depend on other parameters, such as type of experimental protocol, time of the day, developmental stage, or cell location in the tissue?

DUSC improves the clustering accuracy by (i) leveraging a new deep learning architecture, DAWN, which is resilient to the inherent noise in the single-cell data and generates the data representation with automated feature learning, thus efficiently capturing structural patterns of the data, and (ii) pairing this reduced representation with the model-based EM clustering. In particular, DUSC generates

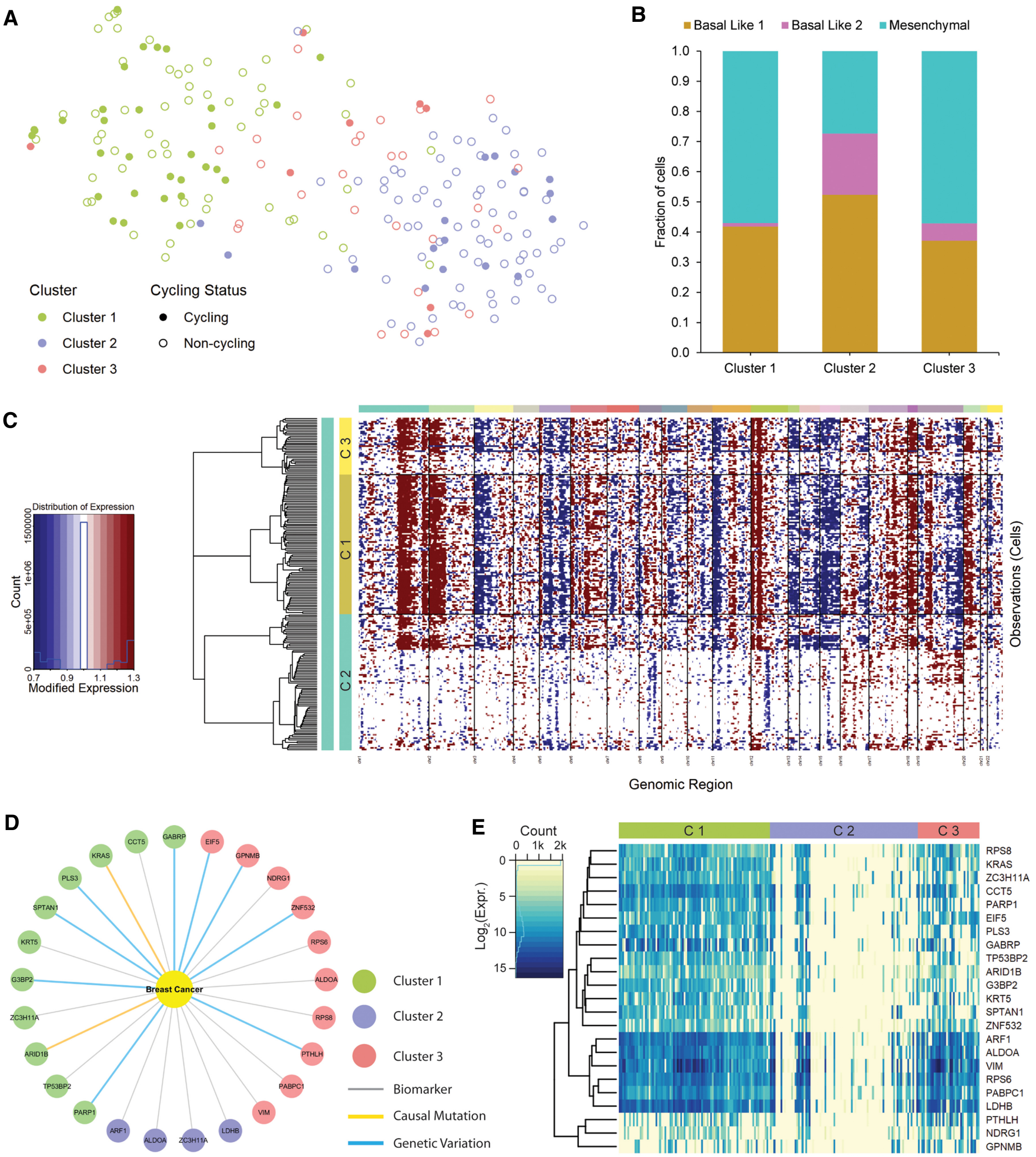

FIGURE 4. DUSC analysis of scRNA-seq triple-negative breast cancer (TNBC) data. (*A*) DUSC embedding of tumor cells from a TNBC patient (Karaayvaz et al. 2018) showing three identified clusters and the cycling status of each cell. Cells in Clusters 1 and 2 are expected to be substantially different based on their transcriptional profiles, while Cluster 3 has similarities to the other two clusters. (*B*) Breakup of cells in each cluster according to their TNBCtype-4 subtypes. (*C*) Inferred copy-number variations in the cells of each cluster; deletions are shown in blue and amplification in red. Cluster 1 (C1) and 3 (C3) cells have similar CNV patterns in many genomic regions, with significant number of both amplifications and deletions. The presence of both types of CNV events are indicative of clonal heterogeneity in C1 and C3. (*D*) Breast cancer associated genes and different evidence of disease associations found in the top 100 differentially expressed genes in each cluster; C1 and C3 have many breast cancer associated (BCA) genes. (*E*) Expression pattern of the 23 BCA genes where the genes are grouped according to expression similarity of their transcriptional profiles. The expression is represented as log-fold change, showing significantly high expression in clusters C1 and C3. Six genes are highly expressed and form a single cluster: *ARF1*, *ALDOA*, *VIM*, *RPS6*, *PABPC1*, and *LDHB*. These genes were found to have implications for TNBC.

more accurate clusters compared to the clustering algorithms alone and is better than four classical and state-of-the-art feature learning methods integrated with the clustering algorithms. Furthermore, our method achieves a comparable performance with a state-of-the-art supervised learning approach. The novel neuronal approximation implemented in the denoising autoencoder simplifies the optimization process for the most important hyper-parameter in the deep architecture, that is, the number of hidden neurons. The simplicity of using DAWN is thus comparable to PCA, and the utility of the newly learned features is illustrated by the better visualization of large scRNA-seq data sets when using a two-dimensional embedding. Our multitiered assessment reveals the dependence of clustering performance on the data set complexity, as defined by an information-theoretic metric, which is due to the size balance of the subpopulations in the data set. Finally, the application of DUSC to a cancer data set shows the ability to reveal clonal heterogeneity in an unsupervised manner and sheds light on the expression patterns of cancer associated genes, and opens the possibility of finding new disease associated genes (Cui et al. 2019).

Considering the current developments in high-performance computing, that is, a drastic increase in the number of CPU cores, the execution time for parallelizable tasks is no longer a major concern. Furthermore, we expect the execution time for DAWN to decrease proportionally if the training epochs are reduced. Contrary to the increase in CPU cores, primary memory density has not seen the same level of improvements and is more valuable than CPU time (Casas and Bronevetsky 2015), moreover, GPU memory is significantly smaller than primary memory. With these considerations for available computing resources, DAWN is a better-suited method, as it is optimized for efficient System memory and GPU memory usage and the execution time scales better for large data sets.

Our next step is to improve the execution time of DUSC for very large data sets containing 100,000+ cells, which are highly heterogeneous and may include a certain cell type hierarchy (Tabula Muris Consortium 2018). We also plan to evaluate if a deeper architecture can improve the feature learning on the massive data sets. An even more challenging task is to improve feature learning for the highly imbalanced data, for example, to be able to detect cell subpopulations of disproportionally small sizes, which would either be absorbed by a larger cluster or identified as noise and removed from the analysis by the traditional methods. We have seen this scenario in the MM data set, the noncancerous subtypes vary in sample size from 52 cells to 2068 cells, which affects the performance of all considered methods. Another interesting application of DUSC is to analyze time-sensitive scRNA-seq data of cell differentiation (Hochgerner et al. 2018). In summary, we believe that DUSC will provide life scientists and clinical researchers a more accurate tool for single-cell data analysis, ultimately leading to deeper insights in our understanding of the cellular atlas of living organisms, as well as improved patient diagnostics treatment. DUSC is implemented as an open-source tool available to researchers through GitHub: https://github.com/KorkinLab/DUSC.

MATERIALS AND METHODS

Overview of the approach

The goal of this work is to design a method capable of identifying cellular types from single-cell transcriptomics data of a heterogeneous population without knowing a priori the number of cell types, subpopulation sizes, or gene markers of the population. Our hybrid approach, DUSC, combines deep feature learning and expectation-maximization clustering. The feature learning leverages the denoising autoencoder (DAE) and includes a new technique to estimate the number of required latent features. To assess the accuracy of our approach, we test it on a series of scRNA-seq data sets that are increasingly complex with respect to the biological and technical variability. The performance of our method is then compared with performances of classical and state-of-the-art unsupervised and supervised learning methods.

The DUSC computational pipeline consists of four main stages (Fig. 1). Following a basic data quality check, we first preprocess the data for training DAE. Second, we perform feature learning using DAWN, which includes training DAE and hyper-parameter optimization. We note that the data labels are not required during the training part of the pipeline; instead, the labels are used solely to test the accuracy of DUSC across the data sets and to compare it against the other methods. Third, we use the previously published four feature learning methods, principal component analysis (PCA) (Abdi and Williams 2010), independent component analysis (ICA) (Comon 1994), t-SNE (Maaten and Hinton 2008), and SIMLR (Wang et al. 2017), to generate the compressed dimensions for the same scRNA-seq data set that was used as an input to DAWN. This allows us to assess how well the autoencoder learns the latent features compared to the other methods. Finally, we use the reduced feature representations from each of the above five methods and pass them as an input to the two clustering algorithms, K-means (KM) and expectation maximization (EM), to assess the clustering accuracy.

Denoising autoencoder model design

During the data preprocessing stage, a data set is defined as a matrix where the rows correspond to the cell samples, and the columns correspond to the feature vectors containing the gene expression values. To reduce the computational complexity, we remove the matrix columns where all values are zeros, which is the only type of gene filtering used in this method (number of genes removed in each data set are detailed in Supplemental Table 1). This minimal filtering procedure is significantly different from a typical gene filtering protocol, whose goal is to restrict the set of genes to a few hundred or a few thousand genes (Usoskin et al. 2015; Zeisel et al. 2015; Lopez et al. 2018). Here, we aim to provide as much data as possible for our deep learning algorithm to capture the true data structure. The columns are then normalized by scaling the gene expression values to [0,1] interval:

$$\mathrm{Norm}(x_i) = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}},$$

where $x_{\max}$ and $x_{\min}$ are the maximum and minimum values across all feature values in vector x, respectively, and x_i is a feature value in

x. The normalized matrix is converted from a 64-bit floating-point representation to a 32-bit representation for native GPU computation and then to a binary file format, to reduce the input–output costs and GPU memory usage during the computation.

An autoencoder (Baldi 2012) is a type of artificial neural network that is used in unsupervised learning to automatically learn features from the unlabeled data. A standard neural network is typically designed for a supervised learning task and includes several layers where each layer consists of an array of basic computational units called neurons, and the output of one neuron serves as an input to another. The first, input, layer takes as an input a multidimensional vector $x^{(i)}$ representing an unlabeled example. The intermediate, hidden, layers are designed to propagate the signal from the input layer. The last, output, layer calculates the final vector of values $z^{(i)}$ corresponding to the class labels in a supervised learning setting. In the autoencoder, the output values are set to be equal to the input values, $x^{(i)} = z^{(i)}$, and the algorithm is divided into two parts, the encoder and the decoder. In the encoder part, the algorithm maps the input to the hidden layer's latent representation $y = s(Wx + b)$, where $s(x)$ is a sigmoid function: $s(x) = 1/(1 + e^{-x})$. In the decoder part, the latent representation y is mapped to the output layer: $z = s'(W'y + b')$. As a result, z is seen as a prediction of x, given y. The weight matrix, W', of the reverse mapping is constrained to be the transpose of the forward mapping, which is referred to as tied weights given by $W' = W^T$. The autoencoder is trained to minimize an error metric defined as the cross-entropy of reconstruction, $L_H(x, z)$, of the latent features:

$$L_H(x, z) = -\sum_{k=1}^{d} [x_k \log z_k + (1 - x_k) \log (1 - z_k)],$$

where d is the length of the feature vector.

To prevent the hidden layer from simply learning the identity function and forcing it to discover more robust features, a DAE is introduced. A DAE is trained to recover the original input from its corrupted version (Vincent et al. 2008). The corrupted version is obtained by randomly selecting n_d features of each input vector $x^{(i)}$ and assigning them zero values. This stochastic process is set up by $\tilde{x} \sim q_D(\tilde{x}|x)$, where $\tilde{x}$ is the corrupted input. Even when the corrupted vectors are provided to the neural network, the reconstruction error is still computed on the original, uncorrupted, input. The optimal number of hidden units for the DAE in this approach is explored as a part of model optimization. The DAE is implemented using the Theano Python library (Bergstra et al. 2010), which supports NVidia CUDA. This implementation allows for fast training of the neural network layers with large numbers of nodes using NVidia GPUs.

Model optimization

The overall architecture of the DAE implemented in our approach consists of an input layer, an output layer, and one hidden layer. There are multiple parameters in this DAE architecture that can be optimized. The task of hyperparameter optimization is fairly unambiguous for supervised learning problems (Chapelle et al. 2002), where the data are labeled, and a neural network can be tuned to set its many parameters such that it achieves an optimal classification performance (e.g., measured by accuracy, f-measure, or other measures). However, in the case of unsupervised clustering where no labeled data are provided and the neural network parameters are optimized to minimize the reconstruction error, the impact of this error metric on clustering is not known a priori. To make this optimization a computationally feasible task, we focus on tuning the number of hidden units, which is expected to have the most significant impact on the model performance (Coates et al. 2011), given its single hidden-layer architecture. The tuning is performed by adopting the ideas from principal component analysis (PCA).

PCA works by converting the initial set of features, which potentially correlate with each other, into linearly uncorrelated features (principal components), through an orthogonal transformation of the feature space. It has been shown that PCA is a special case of the autoencoder where a single hidden layer is used, the transformation function in the hidden units is linear, and a squared error loss is used (Vincent et al. 2010). PCA offers an automated technique to select the first n principal components required to capture a specified amount of variance in a data set (Zwick and Velicer 1986), that is, in a linear autoencoder the principal components are simply the nodes in the hidden layer. The similarity between the two approaches leads us to test, if one can use PCA to approximate the number of hidden units required in an autoencoder to capture most of the data complexity in each data set. As a result, we apply PCA immediately preceding DAE, using the original data set as an input to PCA and producing, as an output, the number n of principal components required to capture 95% of the data set variance (PCA for all data sets is shown in Supplemental Fig. 2). The same data are then processed by DAE with the number of hidden units set to n. We then assess if this additional optimization stage to DAE improves the performance of our approach and call this new extension as denoising autoencoder with neuronal approximation (DAWN).

Since we focus only on the impact of the number of hidden units on the learning efficiency, the settings for all other parameters of the DAE are selected based on a recent work that used DAEs to learn important features in a breast cancer gene expression data set (Tan et al. 2014). Specifically, we set: (i) the learning rate to 0.05; (ii) training time to 250 epochs, which has been reported to be sufficient for the reconstruction error to converge; (iii) batch size to 20, to limit the number of batches for the larger data sets; and (iv) corruption level to 0.1, which specifies that 10% of the input vector features are randomly set to zeroes. The number of hidden neurons estimated for each data set are provided in Supplemental Table 7.

To generate cell clusters from the learned features of DAWN, we use the expectation-maximization (EM) clustering algorithm (Do and Batzoglou 2008). We choose this clustering method because it overcomes some of the main limitations of K-means, such as sensitivity to initial clustering, instance order, noise and the presence of outliers (Celebi et al. 2013). Additionally, EM is a statistical-based clustering algorithm that can work with the clusters of arbitrary shapes and is expected to provide clustering results that are different from those of K-means, which is a distance-based algorithm and works best on the compact clusters. Finally, EM clustering can estimate the number of clusters in the data set, while K-means requires the number of clusters to be specified as an input. These attributes make the EM algorithm a good candidate, because we expect the latent features of DAWN to have specific distributions corresponding to different groups of cells, and we can also approximate the number of clusters.

Comparative assessment of DAWN against existing feature learning approaches

The assessment of the overall performance of our DUSC pipeline includes evaluating the performances of both, the DAWN method and EM clustering algorithm. To evaluate the accuracy of feature learning by DAWN, we compare it against the four other feature learning methods: a stand-alone PCA, ICA, t-SNE, and SIMLR.

PCA is widely used across many computational areas, including scRNA-seq analysis, to reduce the data complexity and to make the downstream analysis computationally more feasible. The method is used for dimensionality reduction in popular scRNA-seq analysis tools, such as Seurat (Villani et al. 2017) and pcaReduce (Yau 2016) along with many others. However, PCA is not an optimal method for dimensionality reduction in our case because of the inherent noise and complexity in the scRNA-seq data. As a result, we expect that PCA cannot optimally capture the true signals in the data and will lead to loss in information. During the assessment stage, we set the PCA algorithm to select the minimal number of principal components required to learn 95% of variance in the data.

Independent component analysis (ICA) is another statistical method designed to separate a multivariate signal into additive subcomponents, which has been applied to a wide range of image analysis and signal processing tasks (Bartlett et al. 2002; Delorme and Makeig 2004). Assuming that the scRNA-seq data can be represented as a mixture of non-Gaussian distributions, ICA can potentially determine the individual independent components that best capture the cell type information in its transcriptomics profile. ICA was used to develop another popular scRNA-seq analysis tool, Monocle, for determining changes in the transcriptome with temporal resolution during cell differentiation (Trapnell et al. 2014). However, similar to PCA, we expect that ICA will also not be optimal to capture the information from complex scRNA-seq data. Additionally, both PCA and ICA make certain assumptions on the data structure: in addition to being linear methods, they are not designed to handle the considerable amount of noise present in the scRNA-seq data. Unlike PCA, the ICA algorithm cannot automatically choose the number of components required to learn a given amount of data variance. Hence, we manually set the number of components to the same number derived by the PCA method when it is required to learn 95% of the data variance.

t-Distributed stochastic neighbor embedding (t-SNE) is a nonlinear feature learning technique specifically designed to map and visualize high-dimensional data into two-dimensional (2D) or three-dimensional (3D) spaces (Maaten and Hinton 2008). t-SNE is often used in scRNA-seq studies to visualize cell subpopulations in a heterogeneous population (Klein et al. 2015). The technique is very efficient in capturing critical parts of the local structure of the high-dimensional data, while facing difficulties in preserving the global hierarchical structure of clusters (Wattenberg et al. 2016). Another potential drawback of t-SNE is the time and space complexities that are both quadratic in the number of samples. Thus, this method is typically applied to a smaller subset of highly variable gene features. When evaluating it against DAWN, we use t-SNE only for feature learning. t-SNE is dependent on an important parameter, perplexity, which estimates the effective number of neighbors for each data point. Here, instead of setting it arbitrarily in the range of [5,50], we calculate it precisely for each data set based on Shannon entropy (discussed below).

Single-cell interpretation via multikernel learning (SIMLR) is a recent state-of-the-art computational approach that performs the feature learning, clustering, and visualization of scRNA-seq data by learning a distance metric that best estimates the structure of the data (Wang et al. 2017). The general form of the distance between cells i and j is expressed as a weighted combination of multiple kernels:

$$D(i, j) = 2 - 2\sum_{l} w_l K_l(i, j),$$

where, w_l is the linear weight value, which represents the importance of each kernel $K_l(i, j)$, and each kernel is a function of the expression values for cells i and j. The similarity matrix S_{ij} is therefore a $N \times N$ matrix where N is the number of samples, capturing the pairwise expression-based similarities of cells:

$$S_{ij} = \sum_{l} w_l K_l(i, j).$$

In SIMLR, to reduce the effects of noise and dropouts in the data, a diffusion-based technique (Yang and Leskovec 2010) is employed. However, this technique is computationally expensive and therefore can be only applied to small or medium-size data sets (e.g., in the published work, any data set with a sample size greater than 3000 did not use this technique [Wang et al. 2017]). Hence, the noise and dropouts effects remain present in the large data sets. Furthermore, the SIMLR framework uses K-means as its clustering algorithm and is affected by the previously discussed limitations. While SIMLR has the capability to estimate the number of clusters, to compare DAWN with the best possible performance of SIMLR, we set the true number of clusters for each data set as an input to SIMLR. Note that this information about the number of clusters is not provided to any other method. The PCA, ICA, and t-SNE algorithms were evaluated using the implementations in the Python *scikit-learn* library (Pedregosa et al. 2011), while SIMLR was evaluated using its implementation as an R package.

Evaluation protocol

All five feature learning methods are evaluated by integrating each of them with one of the two clustering algorithms used in this work, K-means or EM. To do so, we use the latent features uncovered by each of the five methods as inputs to the two clustering algorithms. This setup also allows us to comparatively assess the individual contributions toward the prediction accuracy by each of DUSC's two components, DAWN and EM clustering. Indeed, one can first assess how much the addition of DAWN to K-means or EM can affect the clustering accuracy by comparing the performance with K-means and EM when using the default features. Second, one can determine if the EM-based hybrid clustering approach is more accurate than K-means based approach for each of the five feature learning methods (including DAWN). In total, we evaluate all 5 × 2 = 10 combinations of hybrid clustering approaches.

Alternatively, to determine if the neuronal approximation implemented in DAWN improves a standard DAE, we compared the performance of the DUSC pipeline with DAWN and with two DAE configurations. Although the number of hidden units of a DAE can be set to any arbitrary value, we manually set it to 50 in the first configuration and 100 in the second one, making these configurations computationally feasible (Tan et al. 2014) (which we name as DAE-50 and DAE-100 for convenience).

Finding the optimal number of clusters in a data set is often considered an independent computational problem. Therefore, for the assessment of clustering accuracy, we set the expected number of clusters to be the number of cell types originally discovered in each study. To establish the baseline, we applied KM and EM clustering on the original data sets with zero-value features filtered out, and the data being $\log_{10}$ transformed. The KM and EM methods are implemented using the WEKA package (Hall et al. 2009).

After evaluating the performance of DUSC against other unsupervised methods, we next compare it against a state-of-the-art supervised learning approach. While a supervised learning method is unable to discover new cell types, it is expected to be more accurate in identifying the previously learned types that the algorithm has been trained on. We use the log-transformed data as an input and apply the multiclass random forest (RF) algorithm (Breiman 2001) implemented in WEKA, with a 10-fold cross-validation protocol (Kohavi 1995) that selects the best model with the highest accuracy.

For each of the above evaluations, it is desirable to have a common evaluation metric that can handle multiclass data sets. Here, we use a simple accuracy measure (*ACC*), which can be calculated by comparing the predicted cell clusters with the known cell labels:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN},$$

where *TP* is the number of true positives, *TN* is the number of true negatives, *FP* is the number of false positives, and *FN* is the number of false negatives.

In addition to the standard evaluation of the performance accuracy, the following three characteristics of the method's performance are explored. First, we study the performance of the methods as a function of data complexity. Each of the five data sets considered in this work varies with respect to the sample size distributions across different cell types, numbers of cell types, and cell type hierarchy. These three properties are expected to affect the complexity of cluster separation, prompting one to study the correlation between these properties and the clustering accuracy. To measure the distribution balance of samples across all cell types for each data set we use normalized Shannon Entropy (Shannon 2001):

$$H_{NORM} = -\frac{1}{\log_2 k}\sum_{i=1}^{k}\frac{c_i}{n}\log_2\frac{c_i}{n},$$

where n is the total number of samples, k is the number of cell types, and c_i is the number of samples in cell type i. Thus, H_{NORM} approaches 0 if the data set is unbalanced and 1 if it is balanced.

Second, since the learned latent features are designed to capture the complexity of each data set and create its reduced representation, one can assess the data compression performance of DAWN. The data compression ratio (*CR*) is defined as a ratio between the sizes of the original uncompressed and compressed data sets. A normalized value that allows interpreting the compression performance more intuitively in terms of feature space compressed (*FSC*), is defined as

$$FSC = 1 - \frac{1}{CR}.$$

FSC value approaching 1 implies that the original data set has been compressed to a very small feature set size.

We also profile the execution time and memory usage of all methods during the feature learning stage to determine the computational requirements. Since KM and EM clustering are applied at the end of each feature learning method, they are treated as a constant with negligible computational impact. We want to determine how DAWN scales with increasing data size and complexity. Since DAWN is a deep learning method, it relies on CPU execution to initialize and then deep learning is carried out using GPU, whereas the other methods rely solely on CPU execution. Thus, for DAWN we profile both the system memory use and GPU memory use.

Finally, to determine if DUSC can improve the cell type cluster visualization, we generate two-dimensional embeddings by applying t-SNE to the features of the four previously considered feature learning methods as well as features generated by DAWN. In our qualitative assessment of the visualizations, we expect to see the clusters that are well-separated and compact (i.e., the intracluster distances are much smaller than intercluster distances), and the instances of incorrect clustering are rare.

Integration of scRNA-seq clustering and copy-number variation analysis to study clonal evolution

Next, we apply DUSC to provide insights into clonal evolution in a recently published study on clonal heterogeneity in triple-negative breast cancer (Karaayvaz et al. 2018). Triple-negative breast cancer (TNBC) is characterized by lacking progesterone and estrogen receptors and human epidermal growth factor. It is known for high levels of inter- and intratumor heterogeneity, which is suggested to cause treatment resistance and metastasis development (Koren and Bentires-Alj 2015). Cancer clonal evolution within a primary tumor is reported to be one of the possible reasons for metastasis occurrence (Hoadley et al. 2016). In this case, by applying DUSC we expect to uncover clusters that have cancer-relevant characteristics and are possibly associated with clonal heterogeneity. We evaluate the discovered clusters in several ways. First, we annotate the cycling status of the cells to find suspected malignant cells and subtypes of TNBC (TNBCtype-4 signatures). Second, we infer the copy-number variation (CNV) for each cell across the clusters to find somatic CNVs that are indicative of clonal heterogeneity. Finally, we perform differential gene expression analysis and query the top 100 differentially expressed genes of each cluster for their association with breast cancer. For CNV analysis, we use the inferCNV tool (https://github.com/broadinstitute/inferCNV), which was developed to infer copy number alterations from the tumor single-cell RNA-seq data. The typical inferCNV analysis centers the expressions of tumor cell genes by subtracting the mean expression of corresponding genes from a reference data set of normal cells. For this purpose, we use 240 normal cells published in a recent breast cancer study (Gao et al. 2017) as a reference input for inferCNV.

DATA DEPOSITION

DUSC is implemented as an open-source tool freely available to researchers through GitHub: https://github.com/KorkinLab/DUSC. The data sets analyzed during the current study are available from the corresponding authors on reasonable request.

SUPPLEMENTAL MATERIAL

Supplemental material is available for this article.

ACKNOWLEDGMENTS

We would like to thank Professor Eibe Frank for helpful information regarding the functionality of the Weka data mining tool. This work was supported by the National Institutes of Health, National Library of Medicine (LM012772 to D.K.); and the National Science Foundation (1458267 to D.K.).

Author contributions: D.K., S.S., and N.T.J. conceived the idea. S.S., A.L., and N.T.J. collected the data sets. S.S. implemented the method DUSC. S.S., A.L., N.T.J., and D.K. analyzed the results. S.S., A.L., and D.K. wrote the manuscript, with all authors contributing to the manuscript discussion and editing

Received January 3, 2020; accepted May 22, 2020.

REFERENCES

Abdi H, Williams LJ. 2010. Principal component analysis. *Wiley Interdiscip Rev Comput Stat* **2:** 433–459. doi:10.1002/wics.101

Anders S, Pyl PT, Huber W. 2015. HTSeq–a Python framework to work with high-throughput sequencing data. *Bioinformatics* **31:** 166–169. doi:10.1093/bioinformatics/btu638

Baldi P. 2012. Autoencoders, unsupervised learning, and deep architectures. In *Proceedings of ICML workshop on unsupervised and transfer learning*, pp. 37–49.

Bartlett MS, Movellan JR, Sejnowski TJ. 2002. Face recognition by independent component analysis. *IEEE Trans Neural Netw* **13:** 1450. doi:10.1109/TNN.2002.804287

Bergstra J, Breuleux O, Bastien F, Lamblin P, Pascanu R, Desjardins G, Turian J, Warde-Farley D, Bengio Y. 2010. Theano: a CPU and GPU math compiler in Python. *Proc 9th Python Sci Conf* **1:** 3–10. doi:10.25080/Majora-92bf1922-003

Biase FH, Cao X, Zhong S. 2014. Cell fate inclination within 2-cell and 4-cell mouse embryos revealed by single-cell RNA sequencing. *Genome Res* **24:** 1787–1796. doi:10.1101/gr.177725.114

Breiman L. 2001. Random forests. *Mach Learn* **45:** 5–32. doi:10.1023/A:1010933404324

Brennecke P, Anders S, Kim JK, Kołodziejczyk AA, Zhang X, Proserpio V, Baying B, Benes V, Teichmann SA, Marioni JC, et al. 2013. Accounting for technical noise in single-cell RNA-seq experiments. *Nat Methods* **10:** 1093. doi:10.1038/nmeth.2645

Carpenter AE, Jones TR, Lamprecht MR, Clarke C, Kang IH, Friman O, Guertin DA, Chang JH, Lindquist RA, Moffat J, et al. 2006. CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biol* **7:** R100. doi:10.1186/gb-2006-7-10-r100

Casas M, Bronevetsky G. 2015. Evaluation of HPC applications' memory resource consumption via active measurement. *IEEE Trans Parallel Distrib Syst* **27:** 2560–2573. doi:10.1109/TPDS.2015.2506563

Cascone T, McKenzie JA, Mbofung RM, Punt S, Wang Z, Xu C, Williams LJ, Wang Z, Bristow CA, Carugo A, et al. 2018. Increased tumor glycolysis characterizes immune resistance to adoptive T cell therapy. *Cell Metab* **27:** 977–987. doi:10.1016/j.cmet.2018.02.024

Celebi ME, Kingravi HA, Vela PA. 2013. A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert Syst Appl* **40:** 200–210. doi:10.1016/j.eswa.2012.07.021

Chapelle O, Vapnik V, Bousquet O, Mukherjee S. 2002. Choosing multiple parameters for support vector machines. *Mach Learn* **46:** 131–159. doi:10.1023/A:1012450327387

Chen CL, Mahjoubfar A, Tai LC, Blaby IK, Huang A, Niazi KR, Jalali B. 2016. Deep learning in label-free cell classification. *Sci Rep* **6:** 21471. doi:10.1038/srep21471

Choobdar S, Ahsen ME, Crawford J, Tomasoni M, Fang T, Lamparter D, Lin J, Hescott B, Hu X, Mercer J, et al. 2019. Assessment of network module identification across complex diseases. *Nat Methods* **16:** 843–852. doi:10.1038/s41592-019-0509-5

Coates A, Ng A, Lee H. 2011. An analysis of single-layer networks in unsupervised feature learning. *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 215–223.

Comon P. 1994. Independent component analysis, a new concept? *Signal Process* **36:** 287–314. doi:10.1016/0165-1684(94)90029-9

Cui H, Srinivasan S, Korkin D. 2019. Enriching human interactome with functional mutations to detect high-impact network modules underlying complex diseases. *Genes (Basel)* **10:** 933. doi:10.3390/genes10110933

Cybenko G. 1989. Approximation by superpositions of a sigmoidal function. *Math Control Signals Syst* **2:** 303–314. doi:10.1007/BF02551274

Darmanis S, Sloan SA, Zhang Y, Enge M, Caneda C, Shuer LM, Gephart MGH, Barres BA, Quake SR. 2015. A survey of human brain transcriptome diversity at the single cell level. *Proc Natl Acad Sci* **112:** 7285–7290. doi:10.1073/pnas.1507125112

Delorme A, Makeig S. 2004. EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *J Neurosci Methods* **134:** 9–21. doi:10.1016/j.jneumeth.2003.10.009

Demšar J, Curk T, Erjavec A, Gorup Č, Hočevar T, Milutinovič M, Možina M, Polajnar M, Toplak M, Starič A, et al. 2013. Orange: data mining toolbox in Python. *J Mach Learn Res* **14:** 2349–2353.

Do CB, Batzoglou S. 2008. What is the expectation maximization algorithm? *Nat Biotechnol* **26:** 897. doi:10.1038/nbt1406

Gao R, Kim C, Sei E, Foukakis T, Crosetto N, Chan LK, Srinivasan M, Zhang H, Meric-Bernstam F, Navin N. 2017. Nanogrid single-nucleus RNA sequencing reveals phenotypic diversity in breast cancer. *Nat Commun* **8:** 228. doi:10.1038/s41467-017-00244-w

Gong W, Kwak IY, Koyano-Nakagawa N, Pan W, Garry DJ. 2018. TCM visualizes trajectories and cell populations from single cell data. *Nat Commun* **9:** 2749. doi:10.1038/s41467-018-05112-9

Goolam M, Scialdone A, Graham SJ, Macaulay IC, Jedrusik A, Hupalowska A, Voet T, Marioni JC, Zernicka-Goetz M. 2016. Heterogeneity in Oct4 and Sox2 targets biases cell fate in 4-cell mouse embryos. *Cell* **165:** 61–74. doi:10.1016/j.cell.2016.01.047

Grün D, Lyubimova A, Kester L, Wiebrands K, Basak O, Sasaki N, Clevers H, van Oudenaarden A. 2015. Single-cell messenger RNA sequencing reveals rare intestinal cell types. *Nature* **525:** 251. doi:10.1038/nature14966

Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH. 2009. The WEKA data mining software: an update. *ACM SIGKDD Explor Newslett* **11:** 10–18. doi:10.1145/1656274.1656278

Hoadley KA, Siegel MB, Kanchi KL, Miller CA, Ding L, Zhao W, He X, Parker JS, Wendl MC, Fulton RS, et al. 2016. Tumor evolution in two patients with basal-like breast cancer: a retrospective genomics study of multiple metastases. *PLoS Med* **13:** e1002174. doi:10.1371/journal.pmed.1002174

Hochgerner H, Zeisel A, Lönnerberg P, Linnarsson S. 2018. Conserved properties of dentate gyrus neurogenesis across postnatal development revealed by single-cell RNA sequencing. *Nat Neurosci* **21:** 290–299. doi:10.1038/s41593-017-0056-2

Hornik K, Stinchcombe M, White H. 1989. Multilayer feedforward networks are universal approximators. *Neural Netw* **2:** 359–366. doi:10.1016/0893-6080(89)90020-8

Huang DW, Sherman BT, Lempicki RA. 2008. Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucleic Acids Res* **37:** 1–13. doi:10.1093/nar/gkn923

Jain AK. 2010. Data clustering: 50 years beyond K-means. *Pattern Recogn Lett* **31:** 651–666. doi:10.1016/j.patrec.2009.09.011

Jang MH, Kim HJ, Kim EJ, Chung YR, Park SY. 2015. Expression of epithelial-mesenchymal transition–related markers in triple-negative breast cancer: ZEB1 as a potential biomarker for poor clinical outcome. *Hum Pathol* **46:** 1267–1274. doi:10.1016/j.humpath.2015.05.010

Jones TR, Kang IH, Wheeler DB, Lindquist RA, Papallo A, Sabatini DM, Golland P, Carpenter AE. 2008. CellProfiler Analyst: data exploration and analysis software for complex image-based screens. *BMC Bioinformatics* **9:** 482. doi:10.1186/1471-2105-9-482

Karaayvaz M, Cristea S, Gillespie SM, Patel AP, Mylvaganam R, Luo CC, Specht MC, Bernstein BE, Michor F, Ellisen LW. 2018. Unravelling subclonal heterogeneity and aggressive disease states in TNBC through single-cell RNA-seq. *Nat Commun* **9:** 3588. doi:10.1038/s41467-018-06052-0

Klein AM, Mazutis L, Akartuna I, Tallapragada N, Veres A, Li V, Peshkin L, Weitz DA, Kirschner MW. 2015. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell* **161:** 1187–1201. doi:10.1016/j.cell.2015.04.044

Kohavi R. 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. *IJCAI (U S)* **14:** 1137–1145.

Kolodziejczyk AA, Kim JK, Svensson V, Marioni JC, Teichmann SA. 2015. The technology and biology of single-cell RNA sequencing. *Mol Cell* **58:** 610–620. doi:10.1016/j.molcel.2015.04.005

Koren S, Bentires-Alj M. 2015. Breast tumor heterogeneity: source of fitness, hurdle for therapy. *Mol Cell* **60:** 537–546. doi:10.1016/j.molcel.2015.10.031

Langmead B, Trapnell C, Pop M, Salzberg SL. 2009. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol* **10:** R25. doi:10.1186/gb-2009-10-3-r25

Li B, Dewey CN. 2011. RSEM: accurate transcript quantification from RNA-seq data with or without a reference genome. *BMC Bioinformatics* **12:** 323. doi:10.1186/1471-2105-12-323

Lin C, Jain S, Kim H, Bar-Joseph Z. 2017. Using neural networks for reducing the dimensions of single-cell RNA-Seq data. *Nucleic Acids Res* **45:** e156. doi:10.1093/nar/gkx681

Lopez R, Regier J, Cole MB, Jordan MI, Yosef N. 2018. Deep generative modeling for single-cell transcriptomics. *Nat Methods* **15:** 1053–1058. doi:10.1038/s41592-018-0229-2

Lu X, Tsao Y, Matsuda S, Hori C. 2013. Speech enhancement based on deep denoising autoencoder. *Interspeech* 436–440.

Maaten LVD, Hinton G. 2008. Visualizing data using t-SNE. *J Mach Learn Res* **9:** 2579–2605.

McCleland ML, Adler AS, Shang Y, Hunsaker T, Truong T, Peterson D, Torres E, Li L, Haley B, Stephan JP, et al. 2012. An integrated genomic screen identifies LDHB as an essential gene for triple-negative breast cancer. *Cancer Res* **72:** 5812–5823. doi:10.1158/0008-5472.CAN-12-1098

Medaglia C, Giladi A, Stoler-Barak L, De Giovanni M, Salame TM, Biram A, David E, Li H, Iannacone M, Shulman Z, et al. 2017. Spatial reconstruction of immune niches by combining photoactivatable reporters and scRNA-seq. *Science* **358:** 1622–1626. doi:10.1126/science.aao4277

Menon V. 2018. Clustering single cells: a review of approaches on high-and low-depth single-cell RNA-seq data. *Brief Funct Genomics* **17:** 240–245. doi:10.1093/bfgp/elx044

Narni-Mancinelli E, Vivier E, Kerdiles YM. 2011. The 'T-cell-ness' of NK cells: unexpected similarities between NK cells and T cells. *Int Immunol* **23:** 427–431. doi:10.1093/intimm/dxr035

Papalexi E, Satija R. 2018. Single-cell RNA sequencing to explore immune cell heterogeneity. *Nat Rev Immunol* **18:** 35. doi:10.1038/nri.2017.76

Park JH, Ahn JH, Kim SB. 2018. How shall we treat early triple-negative breast cancer (TNBC): from the current standard to upcoming immuno-molecular strategies. *ESMO Open* **3:** e000357. doi:10.1136/esmoopen-2018-000357

Patel AP, Tirosh I, Trombetta JJ, Shalek AK, Gillespie SM, Wakimoto H, Cahill DP, Nahed BV, Curry WT, Martuza RL, et al. 2014. Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science* **344:** 1396–1401. doi:10.1126/science.1254257

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, et al. 2011. Scikit-learn: machine learning in Python. *J Mach Learn Res* **12:** 2825–2830.

Piñero J, Bravo À, Queralt-Rosinach N, Gutiérrez-Sacristán A, Deu-Pons J, Centeno E, García-García J, Sanz F, Furlong LI. 2016. DisGeNET: a comprehensive platform integrating information on human disease-associated genes and variants. *Nucleic Acids Res* **45:** gkw943.

Regev A, Teichmann SA, Lander ES, Amit I, Benoist C, Birney E, Bodenmiller B, Campbell P, Carninci P, Clatworthy M, et al. 2017. Science forum: the human cell atlas. *Elife* **6:** e27041. doi:10.7554/eLife.27041

Schlienger S, Campbell S, Pasquin S, Gaboury L, Claing A. 2016. ADP-ribosylation factor 1 expression regulates epithelial-mesenchymal transition and predicts poor clinical outcome in triple-negative breast cancer. *Oncotarget* **7:** 15811. doi:10.18632/oncotarget.7515

Shannon CE. 2001. A mathematical theory of communication. *ACM SIGMOBILE Mobile Comput Commun Rev* **5:** 3–55. doi:10.1145/584091.584093

Shapiro E, Biezuner T, Linnarsson S. 2013. Single-cell sequencing-based technologies will revolutionize whole-organism science. *Nat Rev Genet* **14:** 618. doi:10.1038/nrg3542

Stegle O, Teichmann SA, Marioni JC. 2015. Computational and analytical challenges in single-cell transcriptomics. *Nat Rev Genet* **16:** 133–145. doi:10.1038/nrg3833

Svensson V, Vento-Tormo R, Teichmann SA. 2018. Exponential scaling of single-cell RNA-seq in the past decade. *Nat Protoc* **13:** 599. doi:10.1038/nprot.2017.149

Tabula Muris Consortium. 2018. Single-cell transcriptomics of 20 mouse organs creates a *Tabula Muris*. *Nature* **562:** 367. doi:10.1038/s41586-018-0590-4

Tan J, Ung M, Cheng C, Greene CS. 2014. Unsupervised feature construction and knowledge extraction from genome-wide assays of breast cancer with denoising autoencoders. *Pac Symp Biocomput* 132–143. doi:10.1142/9789814644730_0014

Tanay A, Regev A. 2017. Scaling single-cell genomics from phenomenology to mechanism. *Nature* **541:** 331. doi:10.1038/nature21350

Tirosh I, Izar B, Prakadan SM, Wadsworth MH, Treacy D, Trombetta JJ, Rotem A, Rodman C, Lian C, Murphy G, et al. 2016. Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq. *Science* **352:** 189–196. doi:10.1126/science.aad0501

Trapnell C, Roberts A, Goff L, Pertea G, Kim D, Kelley DR, Pimentel H, Salzberg SL, Rinn JL, Pachter L. 2012. Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nat Protoc* **7:** 562. doi:10.1038/nprot.2012.016

Trapnell C, Cacchiarelli D, Grimsby J, Pokharel P, Li S, Morse M, Lennon NJ, Livak KJ, Mikkelsen TS, Rinn JL. 2014. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat Biotechnol* **32:** 381. doi:10.1038/nbt.2859

Treutlein B, Brownfield DG, Wu AR, Neff NF, Mantalas GL, Espinoza FH, Desai TJ, Krasnow MA, Quake SR. 2014. Reconstructing lineage hierarchies of the distal lung epithelium using single-cell RNA-seq. *Nature* **509:** 371. doi:10.1038/nature13173

Usoskin D, Furlan A, Islam S, Abdo H, Lönnerberg P, Lou D, Hjerling-Leffler J, Haeggström J, Kharchenko O, Kharchenko PV, et al. 2015. Unbiased classification of sensory neuron types by large-scale single-cell RNA sequencing. *Nat Neurosci* **18:** 145. doi:10.1038/nn.3881

Villani AC, Satija R, Reynolds G, Sarkizova S, Shekhar K, Fletcher J, Griesbeck M, Butler A, Zheng S, Lazo S, et al. 2017. Single-cell RNA-seq reveals new types of human blood dendritic cells, monocytes, and progenitors. *Science* **356:** eaah4573. doi:10.1126/science.aah4573

Vincent P, Larochelle H, Bengio Y, Manzagol PA. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on machine learning*, pp. 1096–1103.

Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA. 2010. Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *J Mach Learn Res* **11:** 3371–3408.

Wang B, Zhu J, Pierson E, Ramazzotti D, Batzoglou S. 2017. Visualization and analysis of single-cell RNA-seq data by kernel-based similarity learning. *Nat Methods* **14:** 414–416. doi:10.1038/nmeth.4207

Wattenberg M, Viégas F, Johnson I. 2016. How to use t-sne effectively. *Distill* **1:** e2. doi:10.23915/distill.00002

Yang J, Leskovec J. 2010. Modeling information diffusion in implicit networks. *2010 IEEE International Conference on Data Mining*, pp. 599–608.

Yau C. 2016. pcaReduce: hierarchical clustering of single cell transcriptional profiles. *BMC Bioinformatics* **17:** 140. doi:10.1186/s12859-016-0984-y

Zeisel A, Muñoz-Manchado AB, Codeluppi S, Lönnerberg P, La Manno G, Juréus A, Marques S, Munguba H, He L, Betsholtz C, et al. 2015. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347:** 1138–1142. doi:10.1126/science.aaa1934

Zwick WR, Velicer WF. 1986. Comparison of five rules for determining the number of components to retain. *Psychol Bull* **99:** 432. doi:10.1037/0033-2909.99.3.432