Representation learning on gene expression data

Xinwei Shen, Yunfei Yang
Department of Mathematics
The Hong Kong University of Science and Technology
{xshenal,yyangdc}@connect.ust.hk

Abstract

Data representation benefits us in identifying and extracting information or underlying explanatory factors. By representing the high-dimensional data in a low-dimensional latent space, we reduce the analytical difficulty of the problem. In this report, we explore three levels of representation learning methods: (i) linear dimension reduction methods: PCA and robust PCA, (ii) non-linear dimensionality techniques: manifold learning, and (iii) deep representation methods: VAE, on a single-cell gene expression dataset. We implement various methods and stress on a common problem in single-cell expression analysis, cell clustering, to demonstrate their representation power. We also give detailed analysis and comparison of different approaches as well as different settings of hyperparameters. In conclusion, manifold learning and VAE outperform linear dimensionality reduction approaches on this dataset.

1 Introduction

Data representation plays an essential role in machine learning. Traditional feature engineering is important but labor-intensive. By contrast, representation learning is to learn representations of the data that make it easier to extract useful information when building classifiers or other predictors [1]. This set of techniques allows a system to automatically discover the representations needed for feature detection or classification from raw data. Since the number of independent facts supplied by the data is usually far greater than the number of facts sought, and in consequence much of the information supplied by any body of actual data is irrelevant [2]. Therefore, we agree that the latent representation space is of much lower dimension than the original data space.

When some form of dimensionality reduction is desirable, we hypothesize that the local directions of variation least represented in the training data should be first to be pruned out [1]. The most commonly used dimensionality reduction technique include the principal component analysis (PCA), its generalizations that are more robust outliers, as well as manifold learning. Another branch of methods builds deep representations, taking advantage of the representation power of deep learning. Specifically, representation learning with deep generative models has achieved great progress in recent years, including Deep Boltzmann Machines (DBMs) [3] and variational autoencoders (VAEs) [4].

Representation learning has been applied successfully in various fields including computer vision, speech recognition and gene expression analysis which is the main focus of this report. By representing the high-dimensional gene expression data in a low-dimensional latent space, we reduce the analytical difficulty of the problem. Moreover, in the low-dimensional latent space, it is hoped that patterns or connections between data points that are hard or impossible to identify in the high-dimensional space will be easy to visualize and utilize [5].

Specifically, we explore three levels of representation learning methods: (i) linear dimension reduction methods: PCA and robust PCA, (ii) non-linear dimensionality techniques: manifold learning, and (iii) deep representation methods: VAE, on a single-cell gene expression dataset. We implement various methods and stress on a common problem in single-cell expression analysis, cell clustering, to demonstrate their representation power. We also give detailed analysis and comparison of different approaches as well as different settings of hyperparameters, specifically, the dimension that we use to represent the original data.

The report is organized as follows. Section 2 describes the methods that we use and the reasons for considering them. Section 3 demonstrates the utility of various methods on real data and analyzes the results comprehensively. Section 4 concludes.

2 Methodology

We consider the following three levels of methods and describe the reasons for applying them based on the properties of gene expression data.

2.1 PCA and robust PCA

PCA is one of the most frequently used dimensionality reduction approach. The idea of PCA is to identify the directions of largest variance (principal components) and uses a spectral transformation of the data into a latent space spanned by these principal components. Let $X \in \mathbb{R}^{n\times p}$ be a data matrix. Classical PCA is trying to find a decomposition $X = L + E$ where L is low rank and the error matrix E has a small Frobenius norm which usually is the case for Gaussian noise. However, if some outliers exists, i.e. there are a small amount of sample points which are largely deviated from the main population of samples, the classical PCA is well-known very sensitive to such outliers.

In single-cell gene expression data, outliers are ubiquitous, caused by contaminants occurring during experiments, cell death, etc. Some of these events may appear highly extreme, leading to unreasonable analysis results with PCA. Therefore, we consider robust PCA (RPCA) which behaves much stable at the existence of outliers. Unlike classical PCA, robust PCA looks for the decomposition $X = L + S$ where L is low rank while S is sparse [6].

Most recently, [7] proposes a robust scatter estimation method under the framework of generative adversarial nets (GANs). These estimators are proved to achieve the minimax rate of scatter estimation under Huber's contamination model. We try to apply robust PCA via GAN. Specifically, we first use the methods in [7] to obtain a robust estimate of the scatter matrix, based on which we then conduct PCA. Notice that here our data are not from Huber's contamination model. However, the covariance estimation will hopefully still work if the true distribution P is not far away from Huber's model.

2.2 Manifold learning

As mentioned before, representation learning is often based on the assumption that data concentrate around a low dimensional manifold in high dimensional spaces. Often the manifold may be nonlinear, leading to manifold learning. To visualize and reduce the dimension of the data, we use the following manifold learning methods:

- Multidimensional Scaling (MDS) [8, 9] takes a matrix of pair-wise distances between all data points as input and computes a position for each points. It finds a low-dimensional representation of the data such that the distances between data points are preserved as well as possible.
- Isomap [8, 10] is a generalization of MDS. Isomap uses the distances between neighboring points to estimate the geodesic distances between all data points and then uses MDS to compute the low-dimensional representation. It finds a lower-dimensional embedding which maintains geodesic distances between all points.
- Locally linear embedding (LLE) [8, 11] describes each data point by a linear combination of its neighbors and computes the best weights for each point. It then finds the low-dimensional embedding that preserves the linear combination weights. We also use two variants of LLE (Hessian LLE and LTSA) for visualization.
- t-distributed stochastic neighbor embedding (tSNE) [12] computes a probability distribution over pairs of data points such that similar points have high probability while dissimilar points have small probability, then it finds a low-dimensional representation that produces similar distribution.

2.3 Variational autoencoder

In generative models, the commonly assumed data generating process (DGP) consists of two steps: (1) a value z_i is generated from the prior distribution $p_\theta(z) = \mathcal{N}(\mathbf{0}, \mathbf{I})$; (2) a value x_i is generated from some conditional distribution $p_\theta(x|z)$ [4]. The goal is to infer the parameter θ (learn a probabilistic model of the observed data) or the values of the latent variables z_i (learn the latent representations). In this report we focus on the second goal.

To deal with the intractable maximum likelihood problem, VAE uses an encoder

$$q_\phi(z|x) = \mathcal{N}(\mu_z(x, \phi), \sigma_z^2(y, \boldsymbol{\phi})\mathbf{I}),$$

represented by two neural networks, to approximate it as the variational lower bound. We consider two choices of the conditional distribution $p_\theta(x|z)$ which is known as the decoder: Gaussian decoder and Bernoulli decoder to explicitly and implicitly reconstruct data respectively.

3 Gene expression analysis

We now demonstrate the utility of the above representation learning methods based on a real single-cell expression data set. The Mouse Cortex Cells dataset from [13] contains 3,005 mouse cortex cells and gold-standard labels for seven distinct cell types. We retained the top 558 genes ordered by variance as in [14]. See `https://github.com/xwshen51/csic-pj2` for source code of all the implementations.

We implement various methods mentioned above to demonstrate their representation power. In particular, we stress on a common problem in single-cell expression analysis, cell clustering, to identify the distinct cellular subtypes or states. Typically, this occurs by reducing the high-dimensional gene expression measurements to a low-dimensional representation. The data are then clustered in this low-dimensional space to identify groups of cells exhibiting similar expression behaviors. Similarity is usually defined in terms of the relative positions of the cells in this low-dimensional space: cells that are close together are more likely to be of the same subtype, whilst cells that are far apart are more likely to be of different types.

Furthermore, single-cell RNA-seq data may be challenging for classical dimensionality reduction methods because of the prevalence of dropout events, which lead to zero expression measurements. These dropout events may be the result of technical sampling effects (due to low transcript numbers) or real biology arising from stochastic transcriptional activity [5]. Therefore, we also focus on the impact of dropout events on the output of different dimensionality-reduction algorithms.

3.1 Data reduction and visualization

We implement the representation learning methods to learn the latent representations of gene expression data and reduce the dimensionality. We first apply PCA and robust PCA to see the representation ability of linear dimensionality reduction methods and the robustness against potential outliers. Figure 1 shows the cumulative percentage of eigenvalues of PCA and robust PCA respectively. Hence we know that the top ten PCs and robust PCs can explain more than 70% and 95% variance proportion of the original data.

We visualize the distribution of data on a two-dimensional latent space in Figure 2. Among all the manifold learning methods that we apply, tSNE has the best visualization in the two-dimensional space and it separates different classes of data very well.

3.2 Clustering

To evaluate the performance of the representation learning methods, we use the learned latent representations to cluster the cells. Since we have the ground truth class assignments, we can use the adjusted Rand index (ARI) to measure the clustering performance.

According to the explained variance proportion of PCA and RPCA, we choose the top ten PCs or robust PCs to do clustering with k-means. Because the results of k-means are stochastic, we repeat for 100 times to obtain a fair evaluation. The mean ARI of PCA is 0.4030 with a standard deviation 0.0568. Surprisingly, the ARI of robust PCA is even a bit worse at 0.3817 (0.0538). We speculate

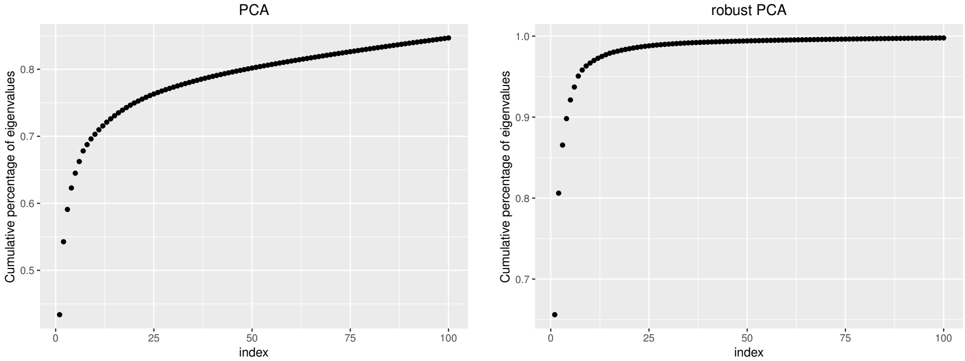
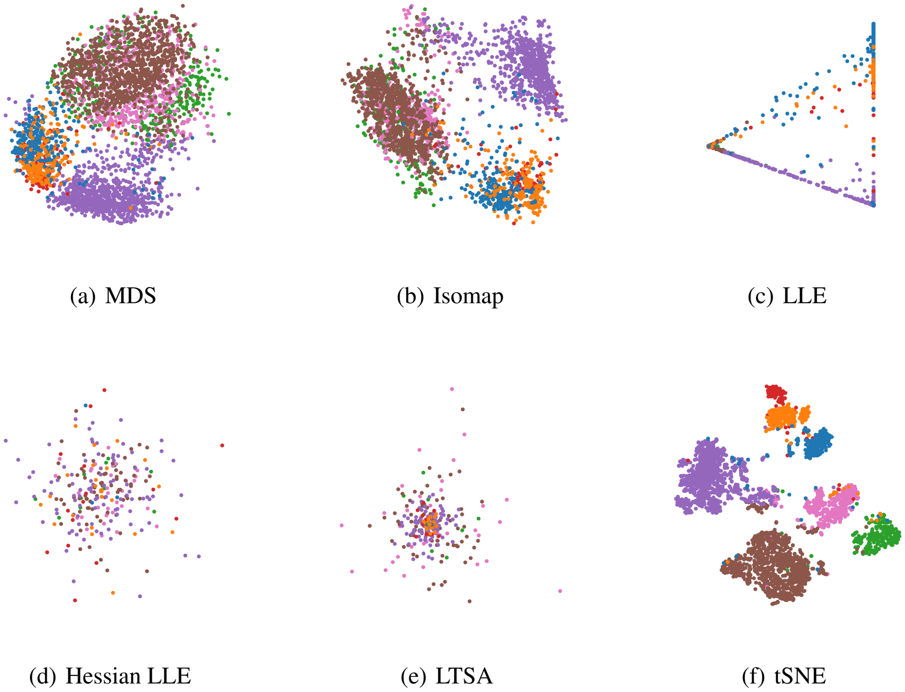


Figure 1: Cumulative percentage of eigenvalues



(a) MDS

(b) Isomap

(c) LLE

(d) Hessian LLE

(e) LTSA

(f) tSNE

Figure 2: Visualization of the dataset

that because this dataset is well preprocessed and is of high quality, the outliers may be removed beforehand. In this normal case, robust PCA, with a special modeling design to deal with outliers, leads to even worse results than classical PCA. For this reason, and also due to the difficulty we meet during GAN training, we consider not to apply PCA with robust scatter estimation on this dataset.

Next, we apply manifold learning (Isomap, LLE and tSNE) and VAE to learn the nonlinear latent representations, and then use k-means and Ward's hierarchical clustering method to cluster the dataset. The results measured by ARI with respect to different latent dimensions are showed in Figure 3 and 4.

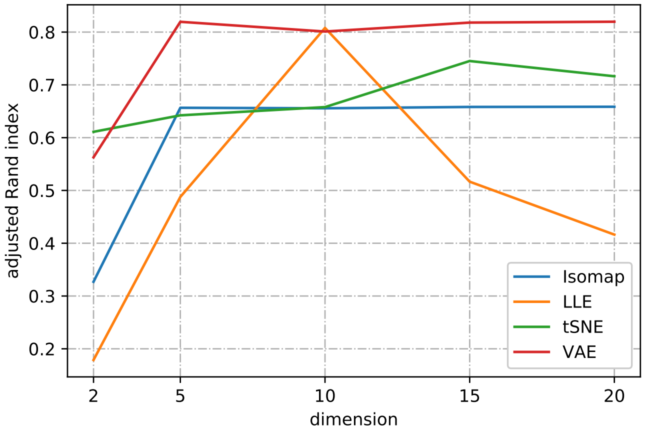


Figure 3: k-means

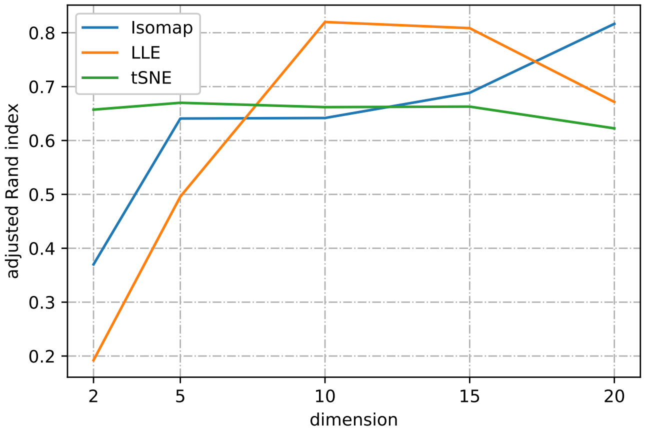
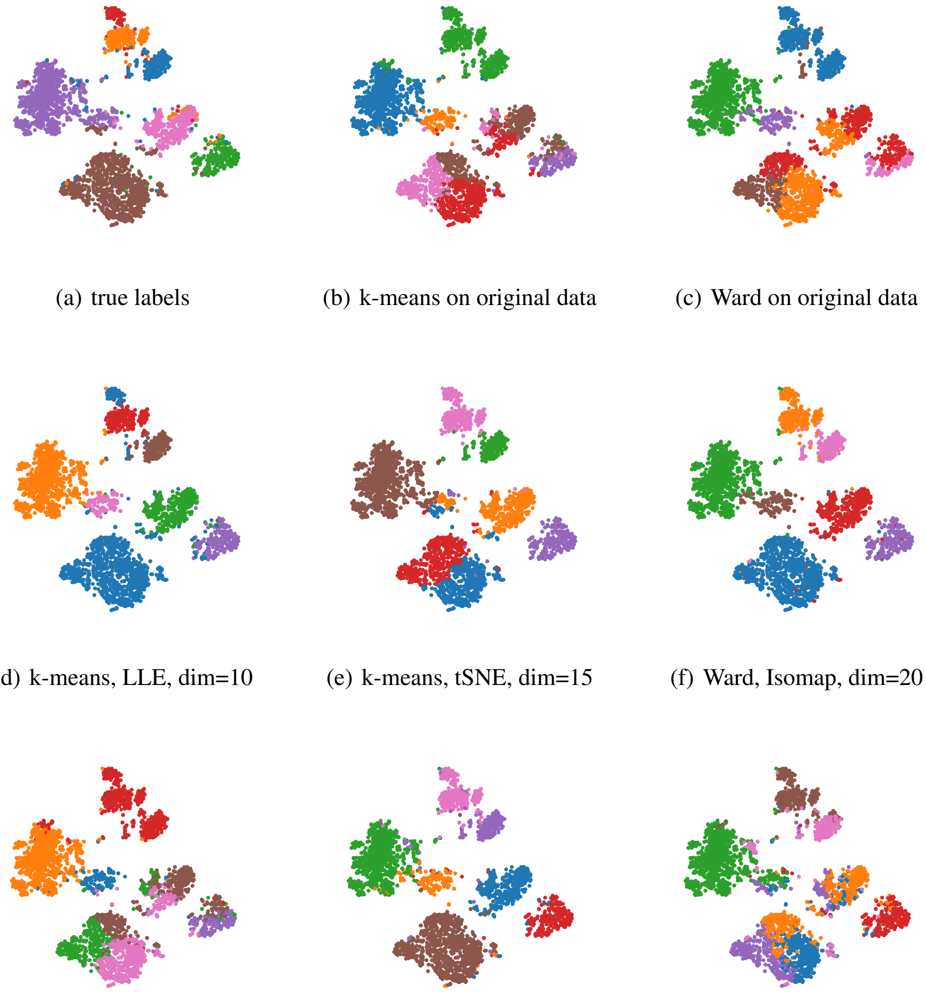


Figure 4: Ward's hierarchical clustering

For comparison, the ARI of k-means and Ward's hierarchical clustering on the original dataset are 0.5 and 0.48 respectively. We notice that all these nonlinear dimensionality reduction techniques significantly outperform PCA, which indicates that the underlying manifold is highly nonlinear. For two dimensions, tSNE has much higher ARI than Isomap and LLE, which is compatible with the visualization in Figure 2 where tSNE separates different classes of data well. An interesting phenomenon is that the ARI of k-means on LLE decreases dramatically as the dimension increases to 15. We think this bad performance is due to the bad convergence of k-means algorithm since the ARI of Ward's hierarchical clustering on the same representation is above 0.8.

In contrast, VAE performs quite well in most cases, which suggests that the deep representations better capture the essential information to distinguish each cluster at least for this dataset. Also its performance is quite robust as the latent dimension that we use to do clustering varies.

To take a closer look at the clustering results, we visualize some of them in Figure 5 on the tSNE representations, since tSNE has the best visualization in two dimensions. The labels (colors) are computed by k-means or Ward's method on different representations. It can be seen that most approaches fail to distinguish the upper part well. Specifically, linear methods like PCA classify all of them into one class. Nonlinear methods are able to detect different data patterns to some extent. LLE does detect three different classes among the upper area but miss up the top class with the one at the bottom (blue). Moreover, most methods mistakenly separate the central part from the left part (see the purple class in the panel with true labels). We see that only Zero Inflated Factor Analysis (ZIFA) [5] which is a statistical model learned with EM algorithm works well in this area. We omit the details of it since has little to do with our focused methods. With the professional knowledge of the cells and genes, we suggest to pay special attention on those hard-to-classify cells which may be some biological signals.



(a) true labels

(b) k-means on original data

(c) Ward on original data

(d) k-means, LLE, dim=10

(e) k-means, tSNE, dim=15

(f) Ward, Isomap, dim=20

(g) k-means, PCA, dim=10

(h) k-means, VAE

(i) k-means, ZIFA model

Figure 5: Visualization of some clustering results

4 Conclusion

In this report, we explore three levels of representation learning methods: (i) linear dimension reduction methods: PCA and robust PCA, (ii) non-linear dimensionality techniques: manifold learning, and (iii) deep representation methods: VAE, on a single-cell gene expression dataset. We implement various methods and stress on a common problem in single-cell expression analysis, cell clustering, to demonstrate their representation power. We also give detailed analysis and comparison of different approaches as well as different settings of hyperparameters. In conclusion, manifold learning and VAE outperform linear dimensionality reduction approaches on this dataset.

References

[1] Y. Bengio, A. C. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 1798–1828, 2013.

[2] R. A. Fisher, "On the Mathematical Foundations of Theoretical Statistics," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 1922.

[3] R. R. Salakhutdinov and G. E. Hinton, "Deep boltzmann machines," in *AISTATS*, 2009.

[4] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *CoRR*, vol. abs/1312.6114, 2014.

[5] E. Pierson and C. Yau, "Zifa: Dimensionality reduction for zero-inflated single-cell gene expression analysis," in *Genome Biology*, 2015.

[6] E. J. Candès, X. Li, Y. Ma, and J. N. Wright, "Robust principal component analysis?," *J. ACM*, vol. 58, pp. 11:1–11:37, 2011.

[7] C. Gao, Y. Yao, and W. Zhu, "Generative adversarial nets for robust scatter estimation: A proper scoring rule perspective," *CoRR*, vol. abs/1903.01944, 2019.

[8] Y. Yao, *A mathematical introduction to data science*. 2019.

[9] I. Borg and P. Groenen, "Modern multidimensional scaling: Theory and applications," *Journal of Educational Measurement*, vol. 40, no. 3, pp. 277–280, 2003.

[10] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[11] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[12] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[13] A. Zeisel, A. B. Muñoz-Manchado, S. Codeluppi, P. Lönnerberg, G. L. Manno, A. Juréus, S. Marques, H. Munguba, L. He, C. Betsholtz, C. Rolny, G. Castelo-Branco, J. Hjerling-Leffler, and S. Linnarsson, "Cell types in the mouse cortex and hippocampus revealed by single-cell rna-seq," *Science*, vol. 347, pp. 1138–1142, 2015.

[14] S. Prabhakaran, E. Azizi, A. Carr, and D. Pe'er, "Dirichlet process mixture model for correcting technical variation in single-cell gene expression data," *JMLR workshop and conference proceedings*, vol. 48, pp. 1070–1079, 2016.

[15] C. Doersch, "Tutorial on variational autoencoders," *CoRR*, vol. abs/1606.05908, 2016.