Classification and Clustering of Single Cell RNA-seq Gene Expression Data Using Deep Learning Features

Chieh Lin
Machine Learning Department, Carnegie Mellon University
chiehl1@andrew.cmu.edu

Introduction

To understand the regulations of cell functions in complex biological systems, it is very important to identify various cell types in the systems. Take nervous systems as example, the cell type diversity is still poorly understood so that the most of the brain functions remain a mystery. Also, even the cells with the same cell type might play different roles in different phases. Therefore, classification of different phases of cells is also important. For now, several of the cell type classification works are based on principal component analysis (PCA) with manual curations. This project aims to apply deep learning methods to extract non-linear features from the single cell RNA-seq data and use the feature in other classification or clustering methods. The ultimate goal of this project is using deep learning to generate deep features to help classify/cluster unknown cell types in order to discover new cell types.

Materials and Methods

From literature survey, papers have been found to use denoising autoencoders to extract deep features. Autoencoder model is selected because its simplicity and it can generate compact representations of hign-dimentional inputs. Keras tool, which uses Theano as backend, is adopted to perform the deep learning part because it it easy and fast to start. 4 mouse RNA-seq datasets with 3007 samples and 15 cell types in total are adopted. Among the datasets, 335 samples have original labels. In the integrated dataset, 20520 genes are the intersection genes and have been tried for deep learning classification directly. A subset of genes is also selected as input dimension since 20520 genes takes too much time to train the deep network. LINCS landmark genes is selected because it has only 978 genes and can capture about 80% information of original set of genes. Two architectures of neural networks have been tried: one hidden layer of size 100 and three hidden of the smallest layer of size 100 with size $\sqrt{input\_dimension/100} * 100$ intermediate layers. Since there are two values of input dimension, four models in total have been trained.

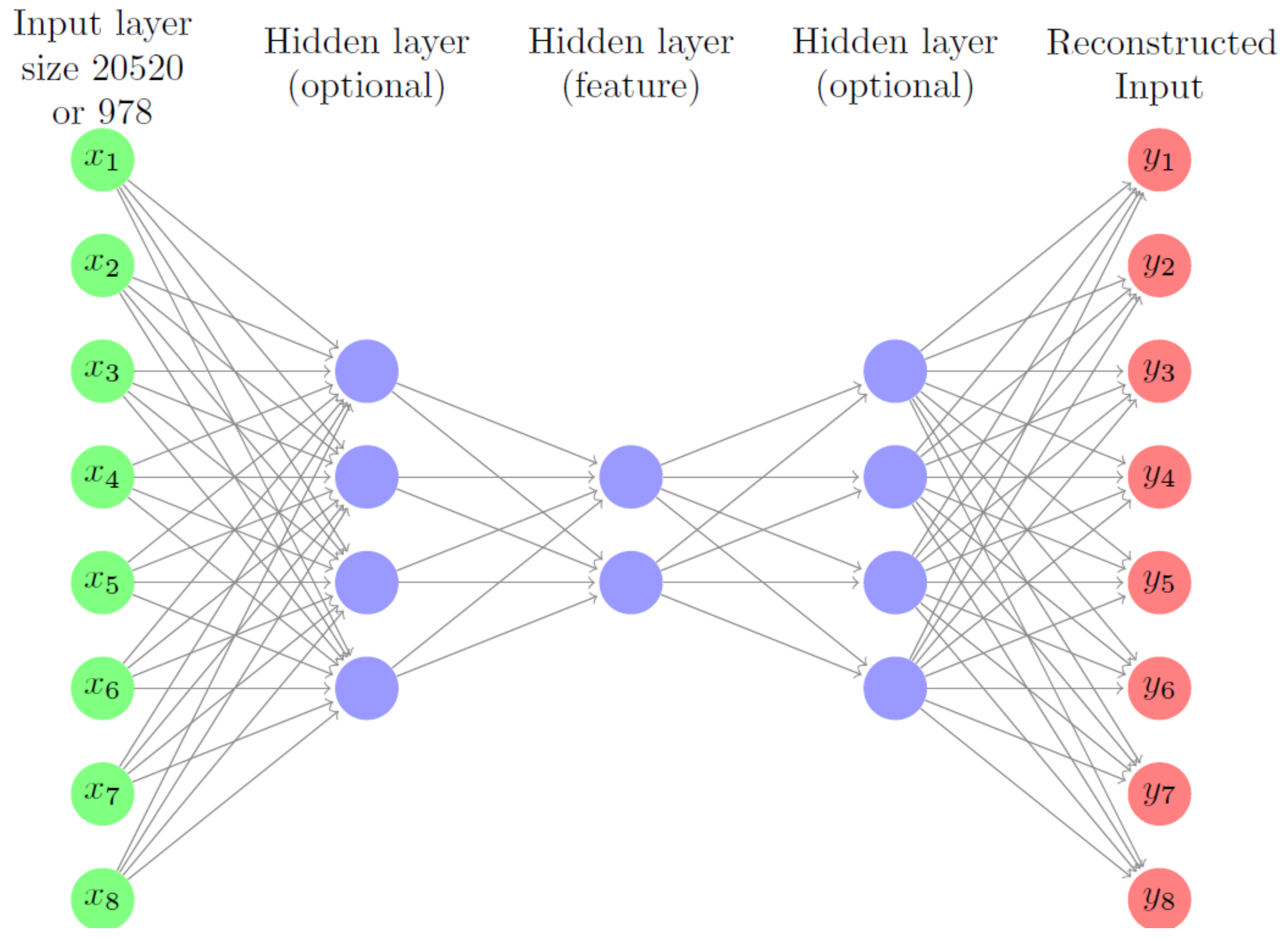

Fig. 1: The neural network structure

To evaluate the performance of deep features, PCA features are selected for comparison and classification and clustering are performed. For classification, two popular methods, Random Forest (RF) and Support Vector Machine (SVM), with four sets of manually chosen parameters are performed on each feature. For clustering, K-means++ is applied to the same set of features. 100 random initializations of K-means++ are performed and the best configuration is selected to compute adjusted random index (ARI) as the performance measurement.

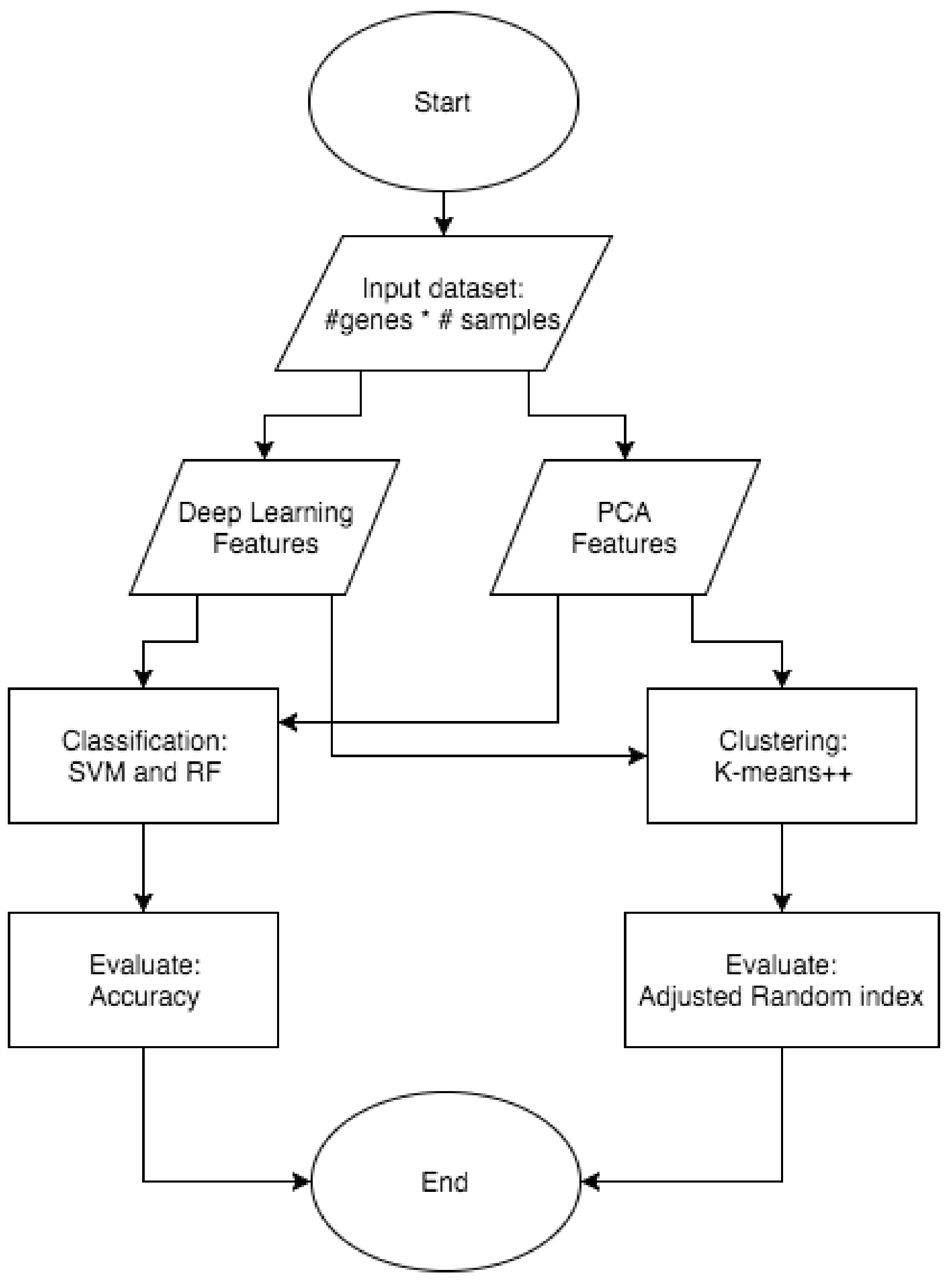
Start
Input dataset: #genes * # samples
Deep Learning Features
PCA Features
Classification: SVM and RF
Clustering: K-means++
Evaluate: Accuracy
Evaluate: Adjusted Random index
End

Fig. 2: The flowchart of methods of this project

Result

Feature	SVM ACC	RF ACC	K-means++ ARI
orignal data+landmark gene	0.96	0.88	0.45
orignal data+all gene	0.97	0.92	0.46
PCA fit all data (335 PC)	0.97	0.83	0.47
PCA fit labeled data (335 PC)	0.97	0.94	0.49
PCA fit all data (100 PC)	0.95	0.94	0.48
PCA fit labeled data (100 PC)	0.95	0.94	0.47
NN 1layer +all gene	0.94	0.91	0.57
NN 1layer +landmark gene	0.94	0.84	0.46
NN 3layer +all gene	0.94	0.92	0.57
NN 3layer +landmark gene	0.92	0.85	0.43

Fig. 3: The classification and clustering results

The best performance PCA on SVM is 0.97, which is greater than the best NN performance 0.94. Besides, the best performance PCA on RF is 0.94, which is also greater than the best NN performance 0.92. Therefore, it could be claimed that PCA performance are slightly better than NN. Besides, the best performance of PCA is 0.49 where the best performance of NN is 0.57. Clearly, NN is much better than PCA in k-means++ clustering.

Conclusion

From the results, it can be observed that PCA is slightly better than NN in classification but NN is much better than PCA in clustering. Since our ultimate goal is discovering new cell types, the result of clustering is more important than classification. We assume that NN can somehow learn non-linear biological information from the input and help clustering performance. Also, the expected result of landmark genes is no better than the set of all genes, and the result corresponds to our assumption. This result shows that if we use only landmark genes to speed up the model training time, we could lose important biological information that can help distinguish new cell types.

Future Work

One possible future plan is to integrate more dataset to increase training set size and include more cell types. About the network architecture, it is a good idea to try denoising autoencoder with corrupted input as the deep feature extraction model. Besides, biological information such as protein interaction network and transcription factors can be used to constructed network structure to reduce the number of parameters to train to reduce training time and prevent overfiting.

Reference

Gupta, Aman, Haohan Wang, and Madhavi Ganapathiraju. "Learning structure in gene expression data using deep architectures, with an application to gene clustering." Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on. IEEE, 2015.

Tan, Jie, et al. "ADAGE-Based Integration of Publicly Available Pseudomonas aeruginosa Gene Expression Data with Denoising Autoencoders Illuminates Microbe-Host Interactions." mSystems 1.1 (2016): e00025-15.

F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. Goodfellow, A. Bergeron, N. Bouchard, D. Warde-Farley and Y. Bengio. âĂIJTheano: new features and speed improvementsâĂİ. NIPS 2012 deep learning workshop. (BibTex)

J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley and Y. Bengio. âĂIJTheano: A CPU and GPU Math Expression CompilerâĂİ. Proceedings of the Python for Scientific Computing Conference (SciPy) 2010. June 30 - July 3, Austin, TX (BibTeX)

keras tool website:
http://keras.io/

LINCS landmark genes website:
http://support.lincscloud.org/hc/en-us/articles/202092616-The-Landmark-Genes

Deng, Qiaolin, et al. "Single-cell RNA-seq reveals dynamic, random monoallelic gene expression in mammalian cells." Science 343.6167 (2014): 193-196.

Tasic, Bosiljka, et al. "Adult mouse cortical cell taxonomy revealed by single cell transcriptomics." Nature neuroscience (2016).

Usoskin, Dmitry, et al. "Unbiased classification of sensory neuron types by large-scale single-cell RNA sequencing." Nature neuroscience 18.1 (2015): 145-153.

Shalek, Alex K., et al. "Single-cell transcriptomics reveals bimodality in expression and splicing in immune cells." Nature 498.7453 (2013): 236-240.