Autoencoder combined with the multilayer perceptron for Alzheimer's disease classification

Xinran Yu

Department of Biochemistry and Molecular Biology, University of California, Davis, CA 95616, U.S.A.

xryu@ucdavis.edu

Abstract. Alzheimer's disease (AD) is a prevalent neurodegenerative disorder that poses significant challenges for accurate diagnosis and treatment. The classification of AD Neurofibrillary Changes (ADNC) levels is crucial for understanding disease progression and developing effective interventions. In this paper, a method was proposed for classifying ADNC levels based on single-cell RNA sequencing (scRNA-seq) data obtained from the SEA-AD dataset. An autoencoder was employed to reduce the dimensionality of the scRNA-seq data, followed by a Multilayer Perceptron (MLP) for classification based on the autoencoder's embedding. The autoencoder effectively reduces the dimension of the scRNA-seq data from 4344 to 30 features. However, the embedding does not exhibit clear boundaries between different ADNC levels. The MLP model achieves a classification accuracy of 39% on the ADNC levels, indicating the complexity of the task and the need for more advanced classification methods. Additionally, the overfitting in both models was observed, and dropout regularization is applied to mitigate this issue. While the results indicate the potential of feature extraction and dimensionality reduction using autoencoders, the accuracy of ADNC level classification remains limited. Combining multiple approaches and aspects in AD diagnosis is necessary, as RNA-seq data alone may not be sufficient for accurate prediction. Future work could explore more sophisticated classification algorithms to improve the accuracy of ADNC level classification and consider integrating other data modalities to enhance disease diagnosis and understanding.

Keywords: Alzheimer's disease, autoencoder, machine learning.

1. Introduction

Alzheimer's disease (AD) is a progressive neurodegenerative disorder that afflicts a substantial portion of the global population. It is the most common form of dementia, accounting for 50 to 56% of cases at autopsy in clinical series [1]. AD can be characterized by cerebral plaques laden with β-amyloid peptide (Aβ) and dystrophic neurites in neocortical terminal fields as well as prominent neurofibrillary tangles in medial temporal-lobe structures [2]. The clinical diagnosis of AD relies on a comprehensive evaluation that incorporates medical history, cognitive testing, and imaging studies. Neuroimaging and cognitive tests which assess memory, language, attention, and other cognitive functions that are affected in AD can aid in diagnosis. Nevertheless, distinguishing AD from other forms of cognitive impairment remains a challenge, as symptoms of AD overlap with those of other types of dementia, such as cognitive impairment caused by vitamin deficiencies or thyroid dysfunction.

Early and accurate diagnosis of AD is crucial for optimal management and treatment of the disease. Thereinto, Alzheimer's Disease Neurofibrillary Changes (ADNC) refers to the characteristic pathological features observed in the brains of individuals affected by AD. The severity and extent of ADNC pathology in the brain are often correlated with the progression and clinical manifestation of cognitive decline observed in individuals with AD. Understanding the underlying mechanisms and effects of ADNC, including the formation, and spread of neurofibrillary tangles, is crucial for developing effective diagnostic tools, therapeutic interventions, and potential disease-modifying treatments for Alzheimer's disease. In this case, the prediction of ADNC levels is of great importance for analyzing the situation of AD.

Previous studies have shown that AD patients have a various gene expression and splicing pattern in the brain region [3]. Therefore, using a transcriptome dataset seems reasonable. In this paper, the scRNA-seq data was employed. Single cell RNA sequencing (scRNA-seq) is a powerful technique for studying gene expression on single-cell level, which allows for the comprehensive analysis of transcriptomes. The principle of scRNA-seq involves isolating individual cells, reverse transcribing their RNA into complementary DNA (cDNA), amplifying the cDNA, and sequencing it using high-throughput next-generation sequencing technologies. This process generates massive amounts of sequencing data, necessitating sophisticated computational methods for data analysis and interpretation.

Specifically, the autoencoder was considered in this study for reducing the complexity in the collected scRNA-seq data. Autoencoders is a powerful class of unsupervised learning algorithms. The basic architecture of an autoencoder consists of an encoder network and a decoder network [4, 5]. The encoder takes in the high-dimensional input data and maps it to a lower-dimensional representation known as the latent space or bottleneck layer. This process effectively compresses the input data, resulting in dimensionality reduction. The decoder network then reconstructs the original data from this lower-dimensional representation, aiming to faithfully capture the salient features and patterns. Applying autoencoder to the SEA-AD dataset, the dimension of the data was reduced from 4344 to 30 in this study. Furthermore, the Multilayer Perceptron (MLP) was constructed to carry out the prediction based on the preprocessed data. It is a widely used artificial neural network architecture [6, 7]. At its core, an MLP is a feedforward neural network consisting of multiple layers of interconnected artificial neurons, known as perceptions. These perceptions are organized in a sequential manner, with each layer receiving inputs from the previous layer and producing outputs that are passed to the subsequent layer. The layers between the input and output layers, known as hidden layers, contain multiple nodes that apply non-linear transformations to the input data, allowing the network to learn complex patterns and relationships. The experimental results demonstrated the effectiveness of the proposed method.

2. Method

2.1. Dataset description and pre-processing

The dataset utilized in this paper called SEA-AD: Seattle Alzheimer's Disease Brain Cell Atlas from Allen Institute [8]. The dataset is an AnnData object, which includes the cell-gene matrix and many labels assigned to each cell, such as ADNC and supertype. There are 40, 000 rows which represent cells and 36, 517 columns which represent genes.

The preprocessing of data is composed of 4 parts. Firstly, genes that only express in fewer than 30 cells are filtered out to avoid introducing bias and to enhance the accuracy of predictions. This step results in a reduction of the number of gene from 36, 517 to 25, 521. Then this study performs the Counts Per Million normalization (CPM) normalization by sc.pp.normalize_total, which normalize each cell by total counts over all genes so that every cell has same total count after normalization [9]. The CPM normalization means the sum of all counts is 1×10^6. By normalizing the gene expression values to a common scale, CPM normalization helps to reduce the impact of differences in sequencing depth between samples, which is important when comparing gene expression across different biological conditions or experimental groups. After the normalization, log transform is applied to the data followed by selecting highly variable genes. The genes that are highly variable among cells is only required since it makes the difference between cells more explicit. This step leads to a huge reduction of number of genes, which is from 25, 521 to 4344. Finally, 80 percentage of the data is randomly assigned to training data while the remaining 20% is assigned to testing data.

2.2. Autoencoder

The first model implemented is autoencoder shown in Figure 1. Both encoder and decoder are composed of 3 linear layers with dropout layers and rectified linear unit (RELU) activation layers between each layer. The first layer of the encoder takes 4344 in feature and 2172 out feature following by a RELU layer and a drop out layer with a 0.4 dropout rate. The second and third layer is the same as the first one except the in feature are 2171, 1086 and out feature are 1086, 30 respectively. The decoder is the inversed mirror image of the encoder with 30 in feature in the first linear layer. The dropout layers are utilized to prevent overfitting by discarding the nodes in a neural network with an assigned probability. In this instance, 40% of the nodes is dropped.

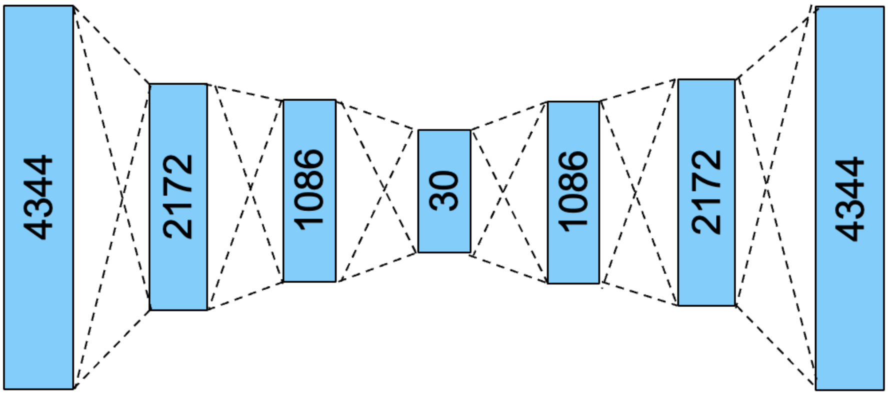

Figure 1. The architecture of the autoencoder used in this study.

2.3. Implementation details for autoencoder.

The batch size for both training dataset and test dataset is 128. Prior to being input into the autoencoder, the data within the training dataset was randomly shuffled. The loss function utilized in training process is nn.MSELoss, which measures the mean squared error (squared L2 norm) between each element of two inputs and returns the mean of all the Mean squared error(MSE). The optimizer is Adam with the learning rate equals to 0.0005. The number of epochs is 100, which leads to a slightly overfitting result.

2.4. Extracting embedding, assigning labels, and balancing data.

After finishing the training of the autoencoder, the embedding based on the training and testing dataset was extracted from the last layer of the encoder The feature dimensions of the embedding are30 and 30, respectively. Then, ADNC levels were assigned to each cell, consisting of five distinct levels: Reference, Not AD, Low, Intermediate, and High. To facilitate feeding into the MLP, numerical values were assigned to each level, whereby 4 represents High, 3 represents Intermediate, 2 represents Low, 1 represents Not AD, and 0 represent reference. After the process is completed for both training dataset and test dataset, the number of each category in the training dataset was examined. It was observed that category 4 had 15012 counts, while category 0 had only 871 counts, indicating a significant data imbalance issue. Therefore, the Synthetic Minority Over-sampling Technique (SMOTE) was applied to the training dataset [10]. SMOTE resolve the over sampling issue by creating artificial examples that interpolate between existing minority class instances. First, the algorithm identifies the instances belonging to the minority class. Then, it randomly selects one of these minority instances. Next, it identifies the k nearest neighbors of the selected instance. From these neighbors, one is randomly chosen.

The algorithm then creates a synthetic instance by interpolating between the selected instance and the chosen neighbor. This is achieved by calculating the difference between the feature values of the two instances, multiplying it by a random ratio, and adding it to the selected instance. By repeating this process until the desired level of oversampling is achieved, SMOTE effectively increases the representation of the minority class, thereby mitigating the class imbalance issue and enhancing the performance of machine learning models. The SMOTE is only performed on the training dataset since the test dataset is not involved in the model training.

2.5. Multilayer perceptron.

The second model in the study is Multilayer Perceptron (MLP) shown in Figure 2. This MLP is composed of 3 linear layers with dropout layers and RELU activation layers between each layer. By experiments, the optimized structure of MLP is that the first linear layer takes 30 in feature and expend it to 60 out features following by a RELU layer and a drop out layer with a 0.2 dropout rate. The second and third layer is the same as the first one except the in feature are 64;16 and out feature are 16;5 respectively.

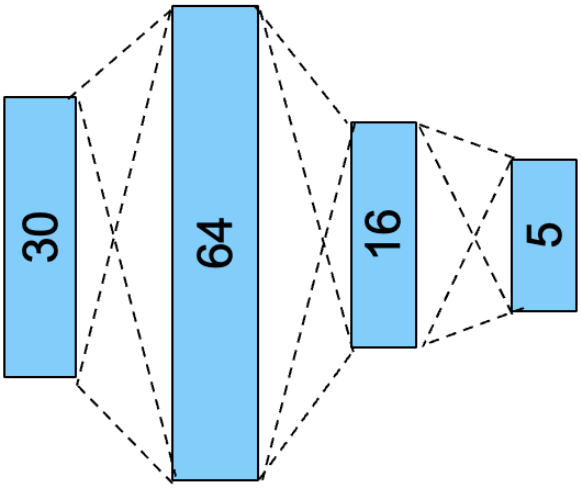

Figure 2. The architecture of the MLP.

2.6. Implementation details for MLP.

The batch size for both training dataset and test dataset is 256. Prior to being input into the MLP, the data within the training dataset was randomly shuffled. The loss function used in the training process is nn.CrossEntropyLoss(). The loss is calculated by taking the negative logarithm of the predicted probability for the true class, penalizing incorrect predictions more heavily. The optimizer is Adam with learning rate equals to 0.001. In addition, the number of epochs is set to 50.

3. Result and discussion

3.1. The performance of the autoencoder

After undergoing 80 epochs of training, the autoencoder model was at a point that optimized the tradeoff between test loss and overfitting. The loss of the train dataset and test dataset were measured to be 0.548 and 0.552 respectively shown in Figure 3. Subsequent to the training phase, the model was applied to the train dataset and test dataset for extracting embeddings from the last linear layer of the encoder part which has 30 features as expected.

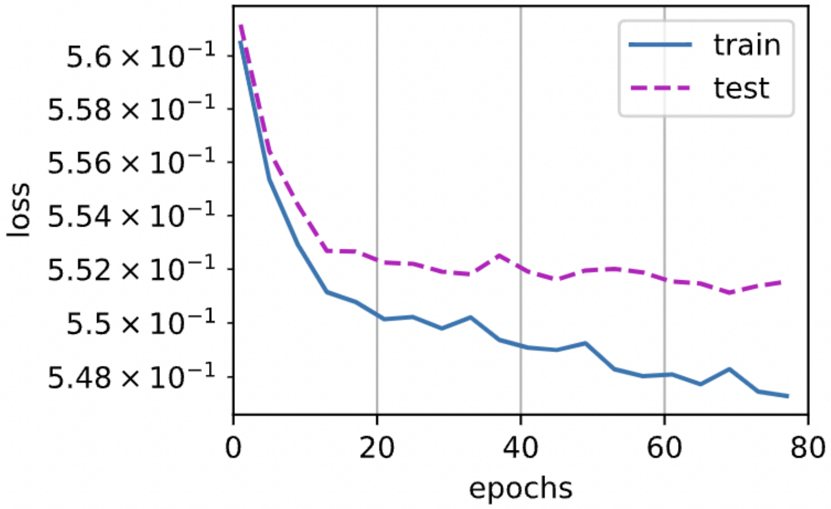

Figure 3. The loss curve of the autoencoder.

Upon generating a 2D Uniform Manifold Approximation and Projection (UMAP) plot of the embedding for the training dataset, the discerned pattern was found to be indistinct shown in Figure 4, lacking a definitive demarcation between the various categories.

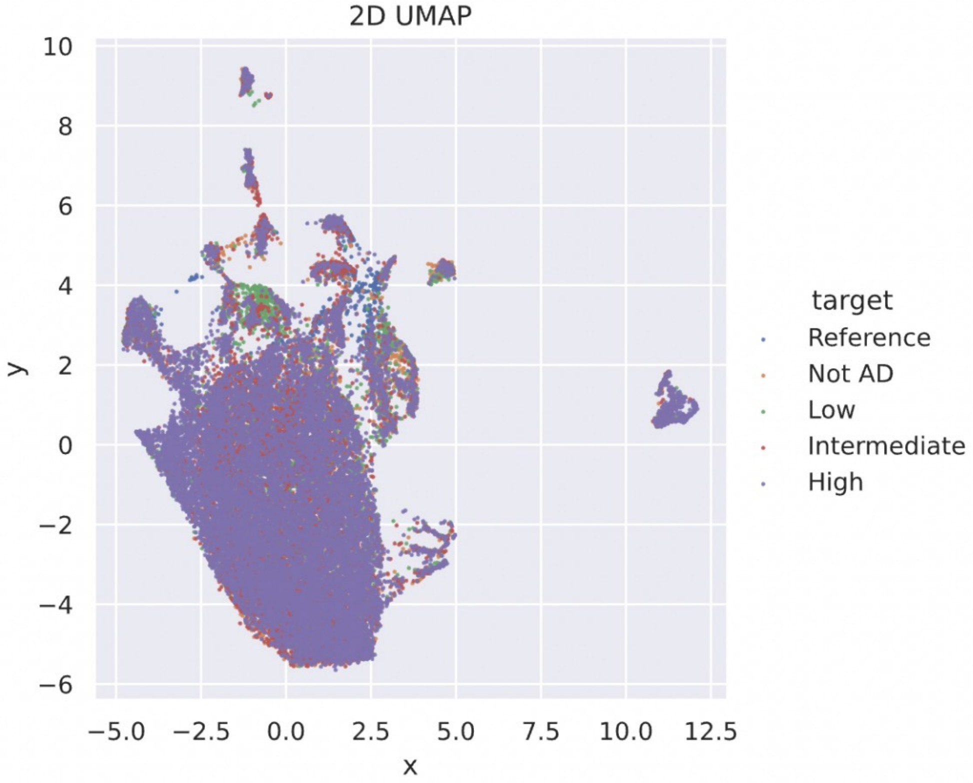

Figure 4. UMAP of embedding generated from training dataset.

The primary goal of autoencoder utilized in this study is dimension reduction. Although it successfully reduced the dimension from 4344 to 30 and displayed some patterns, a few issues persisted throughout the training process and in the resultant outcomes. Specifically, the overfitting of the model could not be avoided, as the loss of the test dataset decreased continuously with the increase in epochs, while the gap between the loss of the training and test datasets widened simultaneously. To mitigate the effects of this issue, dropout was applied to the model. As shown in Figure 5, the model with dropout shows a gap with value 0.006 between training model and test model and that without dropout shows a gap with value 0.04. Therefore, the result with dropout is slightly better. Another issue is the unclear pattern of the embedding, the result of the UMAP failed to present clear boundary between each category. However, the UMAP performed on Principal component analysis (PCA) also generates the similar result. As a consequence, an MLP was implemented in this project to further classify the ADNC levels.

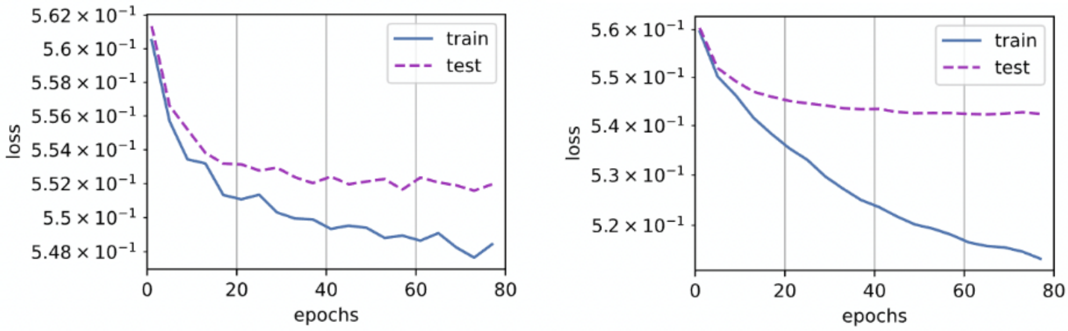

Figure 5. Comparison between with dropout and without dropout.

3.2. *The performance of multilayer perceptron*

This MLP reached the point which has the highest accuracy after 50 epochs shown in Figure 6. At this juncture, the loss of train dataset and test dataset were recorded as 1.037 and 1.3679 respectively. The accuracy of the classification attained was 37%.

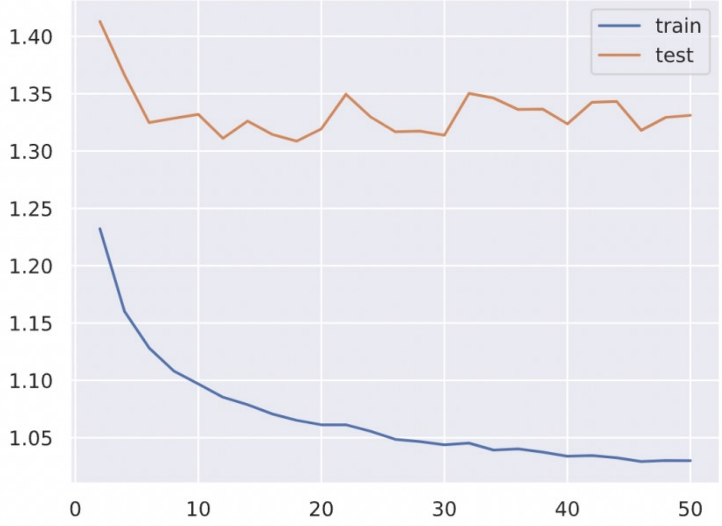

Figure 6. Loss of MLP.

The accuracy of this five categories classification was not commensurate with the anticipated outcome. This can be attributed to several factors. First, as shown in the UMAP, the pattern appeared to be indistinct, suggesting that the feature may not exhibit a high degree of variability and distinctiveness across different ADNC levels. Another possible explanation is that MLP is too straightforward for solving this complicated task. The relatively high loss and the loss tendency support this idea: a more complex algorithm might be more appropriate. Besides, the diagnosis of ADNC levels of AD need to combine different methods and aspects. The RNA-seq data generated from transcriptome is not enough for the accurate prediction.

In order to assess the effectiveness of the autoencoder feature extraction method relative to other techniques, principal component analysis (PCA) was conducted on the training dataset using the same data preprocessing methodology, loss, optimizer, and model. Notably, the accuracy achieved through the use of 30 principal components (PCs) was recorded as 28%. Consequently, the results suggest that the autoencoder represents a superior option for this particular application.

4. Conclusion

This study proposed a method to classify ADNC levels of Alzheimer's disease based on a dataset generated by RNA-seq called SEA-AD. This project utilized the autoencoder to implement the operation of dimension reduction on the original data and MLP to perform prediction based on the embedding of the autoencoder. The accuracy of the classification performance based on autoencoder is 37%, which is obviously higher than the performance of PCA-based classification. These results suggest that the feature extraction and dimension reduction techniques employed in this study are appropriate for the given task. To further advance this research and enhance the model performance, a more complex classification method may be a viable consideration.

References

[1] De-Paula V J Radanovic M Diniz B S Forlenza OV 2012 Alzheimer’s disease Sub-Cell Biochem 65:329–352

[2] Querfurth H W LaFerla F M 2010 Alzheimer’s Disease New England Journal of Medicine Jan 28 362(4):329–44

[3] Twine NA Janitz K Wilkins MR Janitz M 2011 Whole Transcriptome Sequencing Reveals Gene Expression and Splicing Differences in Brain Regions Affected by Alzheimer’s Disease Preiss T editor PLoS ONE 6(1):e16266

[4] Zhang C Liu Y Fu H 2019 Ae2-nets: Autoencoder in autoencoder networks Proceedings of the IEEE/CVF conference on computer vision and pattern recognition 2577-2585

[5] Yu Q Yang Y Lin Z et al 2020 Improved denoising autoencoder for maritime image denoising and semantic segmentation of USV China Communications 17(3): 46-57

[6] Noriega L 2005 Multilayer perceptron tutorial School of Computing Staffordshire University 4: 5

[7] Taud H Mas J F 2018 Multilayer perceptron (MLP) Geomatic approaches for modeling land change scenarios 451-455

[8] CZ CELL×GENE 2023 https://cellxgene.cziscience.com/e/c76098ba-eed3-45b1-98f2-96fcac55ed18.cxg/

[9] Wolf F A Angerer P Theis F J 2018 SCANPY: large-scale single-cell gene expression data analysis Genome biology 19: 1-5

[10] Sudharsan M Thailambal G 2021 Alzheimer’s disease prediction using machine learning techniques and principal component analysis (PCA). Materials Today: Proceedings ISSN 2214-7853 https://doi.org/10.1016/j.matpr.2021.03.061.