CS 4641 Project 3 Report

HU Heng

March 23, 2019

1 Overview

This report is for CS 4641 Machine Learning project 3 Unsupervised Learning. In the following pages, you will see clustering algorithms including Expectation Maximization and K means and dimensionality reduction algorithms including PCA, ICA, Randomized Projection and Feature Selection. I will apply the algorithms on the dataset I used in my project 1 and show the results by some figures and analysis.

2 Datasets

The datasets I choose are wine dataset and adult dataset which are used in my project 1.

2.1 Adult Dataset

Adult dataset is extracted from the 1994 Census database. There are 14 attributes and the result is whether the income is greater than $50k or not. It is a binary classification problem and there are 48842 instances in total.

2.2 Wine Dataset

Wine data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. There are 13 dimensions and three possible classes. There are 178 instances in total.

3 Clustering

3.1 K means

To select a good K, I would like to run K means algorithm multiple times on the two datasets with different K and see the results. K means will start with random cluster centers that have components on each of the given dimensions of the data. Using a distance function, it distributes each instance to the closest cluster center and in the end, takes the mean of the clusters to calculate the new cluster center and restarts the distribution process. It loops until the centers' coordinates on the hyper-dimensions stops changing. I will use sum of squared error to justify the "goodness" of the result. Fig.1 is the result for adult dataset and Fig.2 is the result for wine dataset. It can be predicted that the more cluster we get, the better the result. The trade off is that we are making the data more complicated.

These two figures also contains the curve for doing dimension reduction first and apply K menas algorithm on reduced data. Here the result is very close and I can hardly find any difference for wine dataset. For adult dataset, the difference is also very small.

Figure 1: Adult

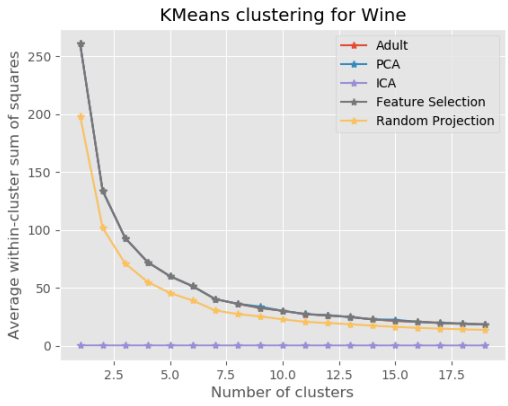

Figure 2: Wine

3.2 Expectation Maximization

Expectation Maximization (EM) is another clustering algorithm that is similar to K means clustering. EM is a more general algorithm and K means is a specific version of it. Instead of directly place the data point into a certain center, EM will assign data points to cluster centers with different probabilities. Thus an instance can be belongs to multiple clusters with certain probability. The result of EM is again several cluster centers that cover different areas of all data points and the list of all the instances with their cluster that probably include them. Log likelihood describes how the instance is related to the cluster center and the result is shown in Fig.3 for adult dataset and Fig.4 for wine dataset.

These two figures also contains the curve for doing dimension reduction first and apply EM algorithm on reduced data. It is clear that log likelihood goes up as number of clusters increases. This is what I expect because large log likelihood means the data points is closer to a certain cluster center. Increase the number of clusters will increase log likelihood trivially. Another observation is that apply ICA before clustering will produce much higher log likelihood result, which is because ICA is going to find the independent vectors. Treat clusters as linearly independent vectors and log likelihood describes the data points in each vectors. Fig.5 and Fig.6 shows the AIC value for two datasets. Fig.7 and Fig.8 shows the BIC value for two datasets.

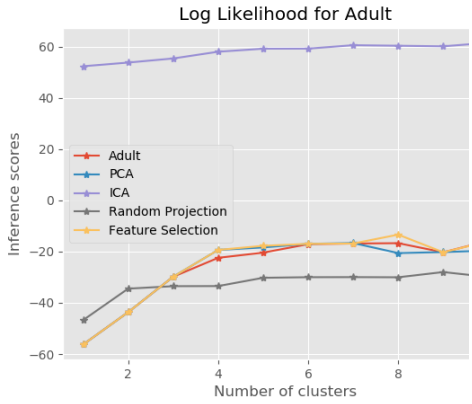

Figure 3: Adult Log Likelihood

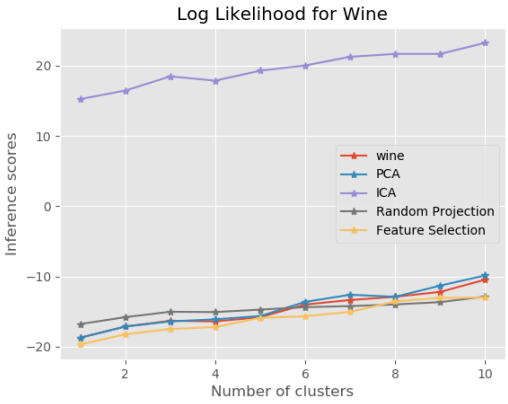

Figure 4: Wine Log Likelihood

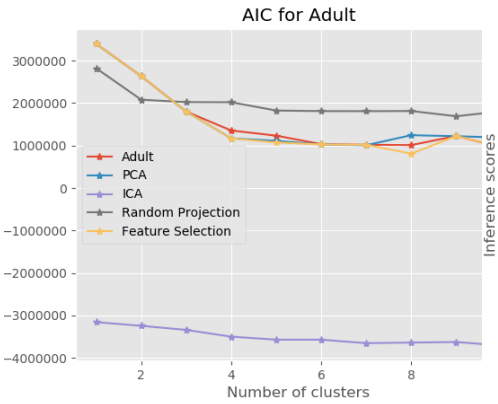

Figure 5: Adult AIC

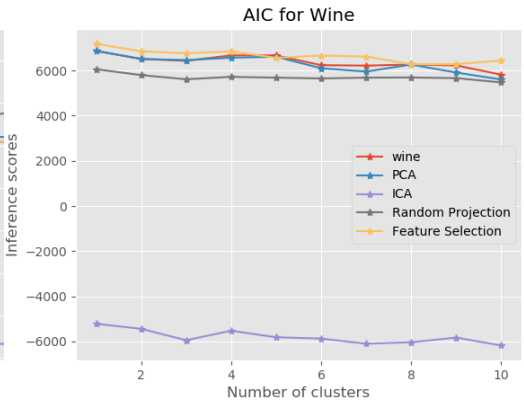

Figure 6: Wine AIC

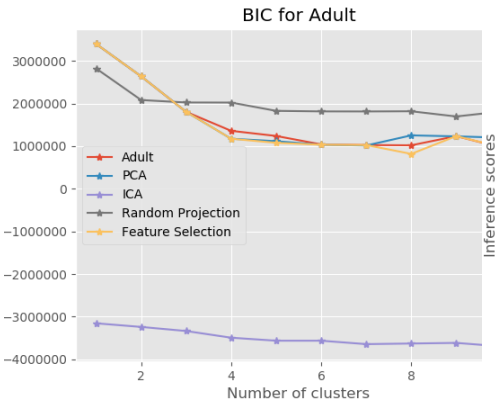

Figure 7: Adult BIC

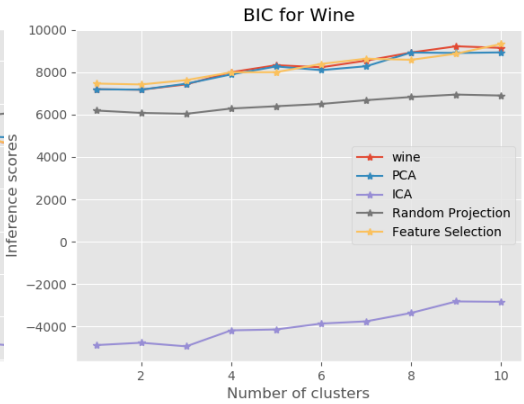

Figure 8: Wine BIC

3.3 Data After Clustering

Another interesting question is what does my data look like after clustering. To have a better idea, I use the wine dataset and plot the result after clustering using K means algorithm. The cluster centers are the big blue points. Fig.9 uses 3 clusters and Fig.10 uses 5 clusters. From the figures, we can see clearly that some parts of the data points are aggregated together with a "center". The data points are almost equally separated. To describe it in another way, the data are "clustered" into different parts to minimize the sum of square error between each point and the center point.

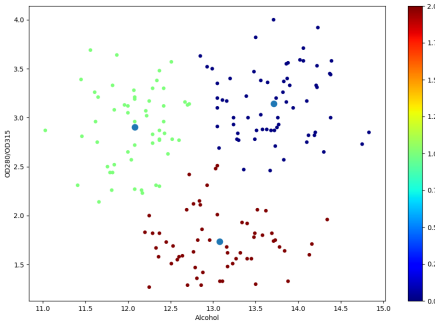

Figure 9: 3 clusters

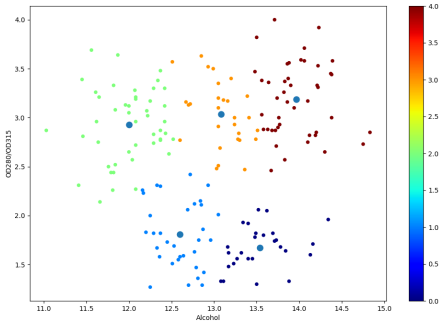

Figure 10: 5 clusters

4 Dimension Reduction

4.1 PCA

PCA will reduce the trends in the data to eigenvectors that represent the major correlations of a given dataset. It tries to find the eigenvector that has the highest variance in order to account for the maximum variability in the data to cover as much instances as possible. Fig.11 shows the wine dataset after applying PCA. There are 13 variables originally and these are shown in the figure by red lines. Points with same colors are actually belongs to the same class. It is clear that data points are well separated. More generally, the process of dimension reduction is to generate a map function and map the raw data from high dimensional space to low dimensional space. Data points from same class will be "closer" after dimensional reduction.

4.2 ICA

ICA is another dimension reduction algorithm that tries to reduce the data into linearly independent vectors. The difference between ICA and PCA is that ICA is going to find the linearly independent vectors while PCA is going to find orthogonal directions in the raw feature space that correspond to directions accounting for maximum variance.

4.3 Random Projection

Random projections is another dimension reduction algorithm that works different with PCA and ICA. It is a simple and computationally efficient way

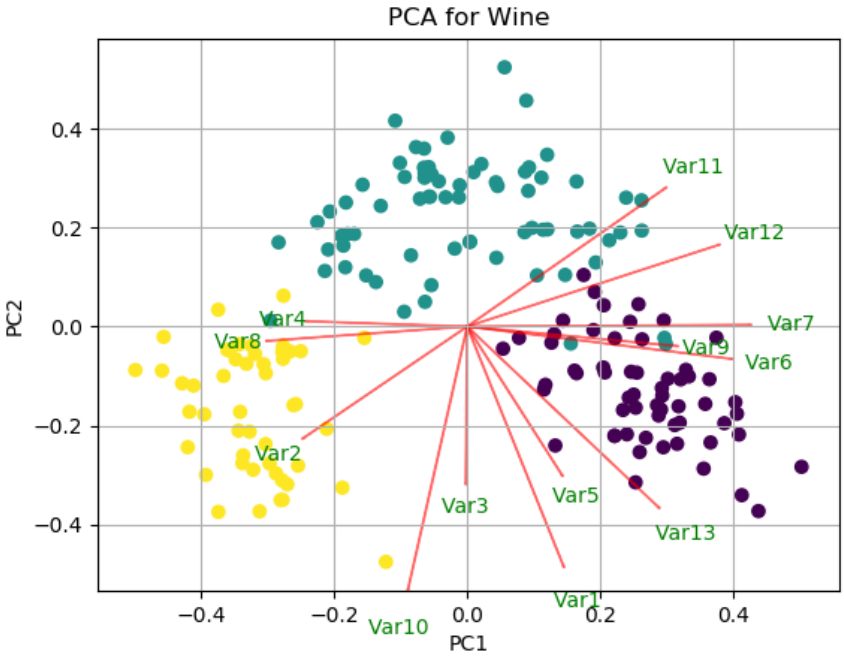

Figure 11: Data after PCA

to reduce the dimensionality of the data by trading a controlled amount of accuracy (as additional variance) for faster processing times and smaller model sizes. In my experiment, I use Gaussian Random Projection.

4.4 Feature Selection

Table 1: Performance of Algorithms

Threshold	**Accuracy**	**Training Time**
0.025	39.88%	0.01097
0.050	28.08%	0.01795
0.075	39.88%	0.01097
0.100	39.88%	0.00897
0.125	49.43%	0.01795
0.150	26.96%	0.01396
0.175	33.14%	0.01097
0.200	26.96%	0.01097
0.225	39.88%	0.00997
0.250	39.88%	0.01296
0.275	39.88%	0.01197
0.300	33.14%	0.01096

I use variance threshold algorithm as the last dimension reduction algorithm. It is very simple and easy to understand. It will look over the features and if the variance of a feature is lower than threshold, the feature will be removed. I use this feature selection algorithm to reduce dimension and feed the data to neural network. The different selection of threshold will influence the accuracy, as shown in Table.1. I may conclude that very low threshold cannot give good accuracy because there still are some useless information or noise. Very large threshold also cannot give good accuracy since it is "over-reduce"(like overfit) and remove too much useful information. From the experiments, 0.125 is a good threshold for wine dataset.

5 Rerun Neural Network on Wine Dataset

5.1 Apply Dimension Reduction Algorithm

Fig.12 shows the accuracy of neural networks trained by data that is preprocessed by different dimension reduction algorithms. Fig.13 shows the

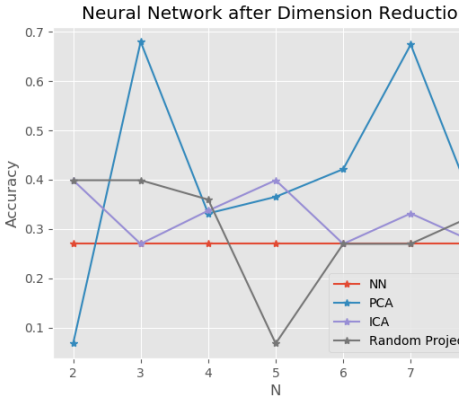

Figure 12: Accuracy

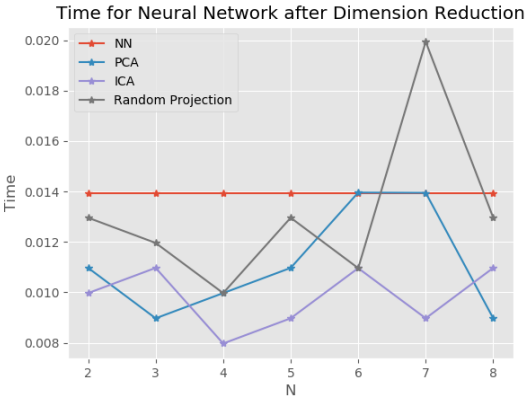

Figure 13: Time

corresponding training time. The x-axis is different selection of N. From the figure, the training time is reduced slightly and the performance is better than the neural network trained by original data. By reduction the dimensions, the training time is slightly reduced since there is less computation. Table.1 shows the performance of using different threshold for feature selection algorithm. Similarly, the accuracy is higher and the training time is reduced. The accuracy increases because the data in high dimensional space may contain noise or useless information. After dimension reduction, the data contain less noise or useless information, thus increasing the accuracy.

5.2 Apply Clustering Algotithm

Fig.14 shows the accuracy of neural networks trained by data that is preprocessed by different clustering algorithms. Fig.15 shows the corresponding training time. The x-axis is different selection of N. From the figure, it is clear that the training time after using clustering algorithms is much shorter. For the accuracy, proper selection of number of clusters will produce a higher accuracy. I believe this is because clustering gives additional information for neural network. A good clustering means the data is being assigned some "correct" feature, so it takes less time to train and can get a better accuracy. On the other hand, a bad clustering will mislead the neural network by containing many data points which are actually belongs to multiple classes. Then the accuracy will drop, as the experiments show.

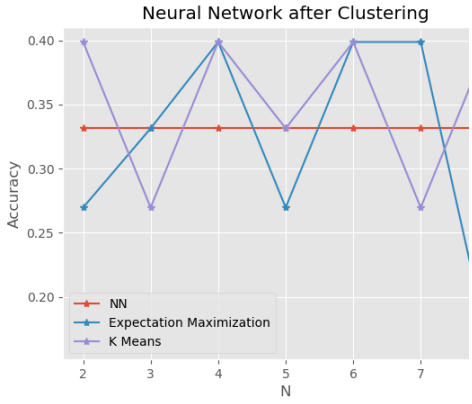

Figure 14: Accuracy

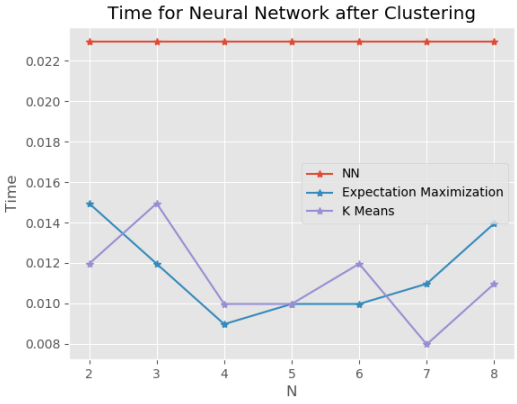

Figure 15: Time