CS 4641 Unsupervised Learning Assignment

Jiahao Luo GT#903103970

1 Introduction

In this report, it mainly talks about two interesting datasets analyzed by implementing two basic clustering algorithm (*k*-means clustering and Expectation Maximum) and four dimensionality reduction algorithms (PCA, ICA, Randomized Projection and Random Subset). Through the entire analyses for both datasets, it will cover on different interesting topics. The whole exploration is based on a popular suite of machine learning software, Weka.

2 Dataset Details

At first, the two datasets that I selected are downloaded from the Auto-Weka website. (http://www.cs.ubc.ca/labs/beta/Projects/autoweka/datasets/) They are the "Car Evaluation" and "Wine Quality" datasets. Here are some details about them.

Table 1. Dataset details of Car Evaluation

Title	Car Evaluation		
Number of instances	1728 (Nominal)	Missing	0
Number of attributes	6	Buying	Maint
Number of classes	4	Doors	Persons
		Lug_boot	Safety

Table 2. Dataset details of Wine Quality

Title	Wine Quality		
Number of instances	6497 (Numeric)	Missing	0
Number of attributes	11	Fixed acidity	Volatile acidity
		Citric acid	Residual sugar
		Chlorides	Free sulfur dioxide
Number of classes	11	Total sulfur dioxide	Density
		pH	Sulphates
		Alcohol	Quality

3 Why Are These Interesting Datasets?

Before I choose these two datasets, I am curious about how to determine if a commodity is valuable or not, especially in the field that may require more experiences to make a decision, such as car and wine quality evaluations. When you are a layman in the car or wine field. How can you estimate the value of them? Which attribute of them should be regard as very important? Can machine learning help us to make a better decision even if we are not familiar with the knowledge about them? If we can figure out this, it not only provide a better decision for consumers (or auto dealers and wine collectors), but also help the producers to better produce the products.

With this question in my mind, I will carry out several experiments to explore the inner significance behind the data.

4 Experiment Methodology

The entire experiments will include the following 5 parts.

1. Run k-Means clustering and Expectation Maximum clustering algorithms on both datasets.
2. Apply PCA, ICA, Randomized Projection and Random Subset, dimensionality reduction algorithms to both datasets.
3. Reproduce the clustering experiments, but on the data after dimensionality reductions have been applied on it.
4. Apply the dimensionality reduction algorithms to one of the datasets from assignment #1 and rerun neural network learn, MultilayerPerceptron on the newly projected data.
5. Apply the clustering algorithms to the same dataset to which has applied the dimensionality reduction algorithms, treating the clusters as new features and rerun neural network learner, MultilayerPerceptron on the newly projected data.

5 Experiment Results

5.1 Clustering Algorithms (k-Means & Expectation Maximum)

According to the part 1, I applied 2 clustering algorithms to both datasets. The dataset will be various depending on the number of clusters. For example, if I choose k equals to 2, then the algorithms will assign instances into two clusters based on Euclidean Distance. And the following figures are the relationship between within group sum of squared errors (SSE) percentage and number of clusters k.

Figure 1. Within group SSE percentage for Car Evaluation dataset

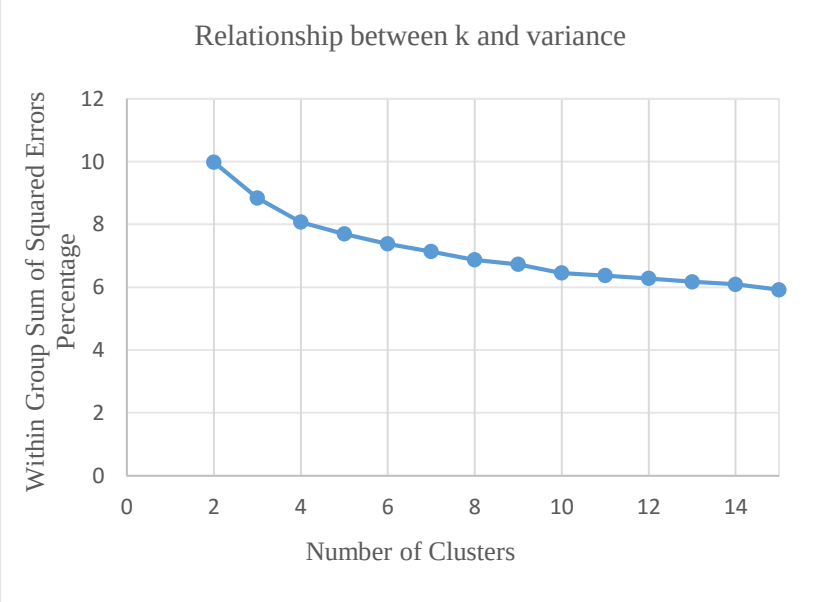


Figure 2. Decline Rate of Variance for Car Evaluation dataset

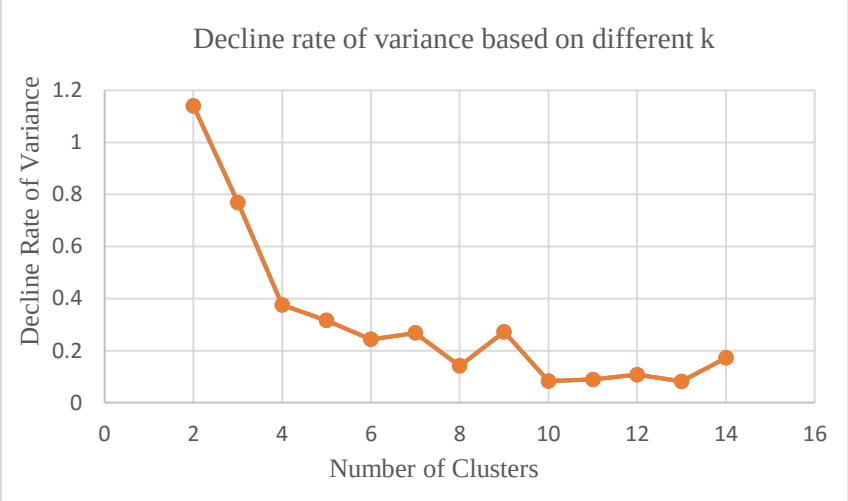


Figure 3. Within group SSE percentage for Wine Quality dataset

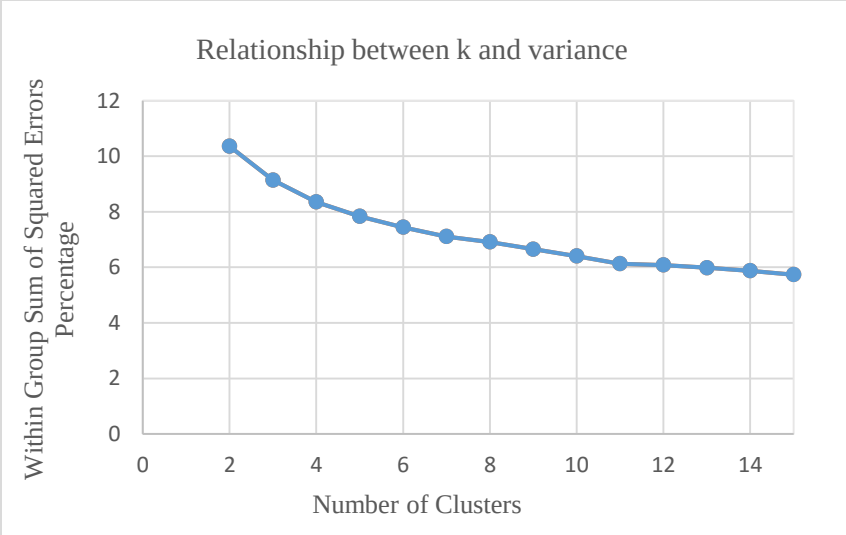
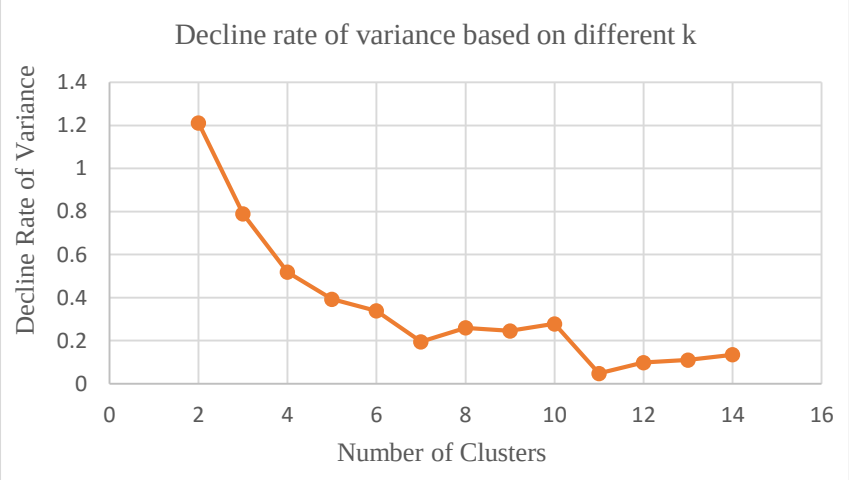


Figure 4. Decline Rate of Variance for Wine Quality dataset



How to choose k? Since the proper choice of k is often ambiguous. Thus, in order to choose the optimal k, I decided to use the Elbow Method. This method tries multiple values of k and looks at the percentage of SSE. The optimal k will be the smallest one after which we see diminishing returns.

According to the figures above, I can obtain the optimal k for Car Evaluation dataset is 4, and for Wine Quality is 8.

5.2 Dimensionality Reduction Algorithms (PCA, ICA, Randomized Projection and Random Subset)
According to the part 2, I applied 4 dimensionality reduction algorithms to both datasets. And the datasets are different after applying those algorithms.

Description of clusters:
According to the following table, the number of attributes is smaller than the original number for both cases. That means the datasets have been reduced the dimensionality by those 4 algorithms which makes sense.
Besides, the clusters that the algorithms generated are not 100% lined up with the relative labels in the original datasets. In my opinion, there are three main reasons might cause this problem.

(1) The algorithms have their applicable range. For example, there is an assumption for PCA that the datasets are under Gaussian distribution, while ICA assumes samples are independent.
(2) Also the dataset might not contain the primary features such that we might not find the best features for our datasets.
(3) The features we have is a combination of various features to generate a more sufficient feature to represent the data label. For example, the features from PCA are several linear equation of previous features.

Table 3. Number of attributes for both datasets

# of attributes	Car Evaluation	Wine Quality
Origin	21*	11
PCA	15	9
ICA	21	11
RP	10	10
RS	11	6

* The original number of attributes for Car Evaluation dataset is 6, since we are going to use ICA, then we need to transfer the nominal attributes to binary.

5.3 Clustering Experiments After Dimensionality Reductions

According to part 3, I reproduced the clustering experiments on the data after dimensionality reductions have been applied on it. The following figures are result for both datasets and both clustering algorithms.

Car Evaluation:

Figure 5. Within group SSE percentage for Car Evaluation dataset

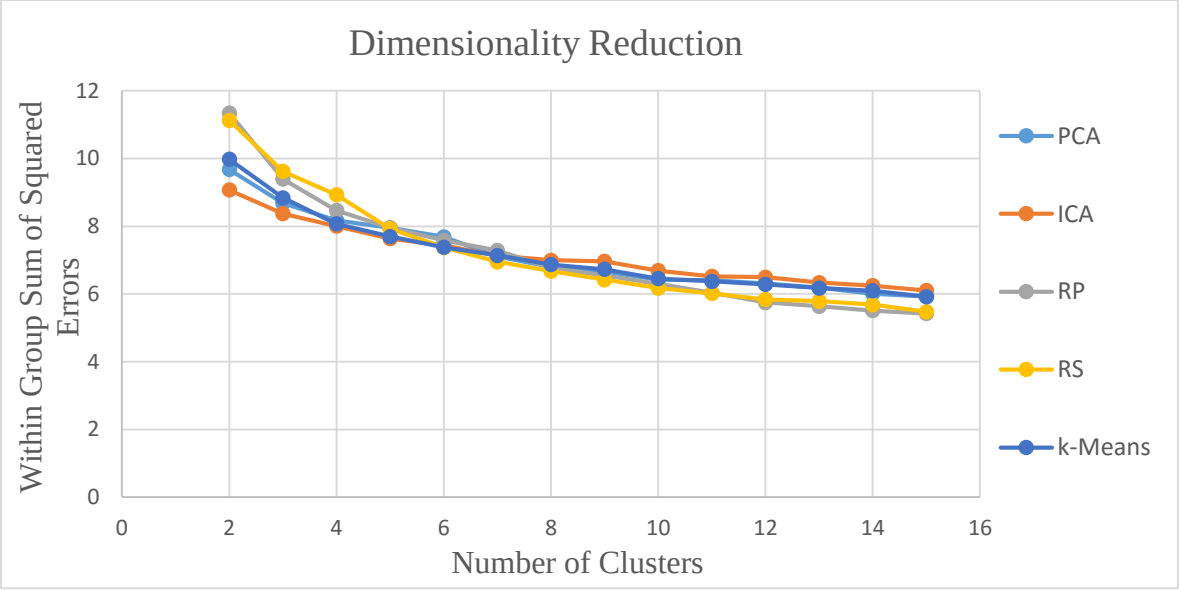


Figure 6. Decline Rate of Variance for Car Evaluation dataset

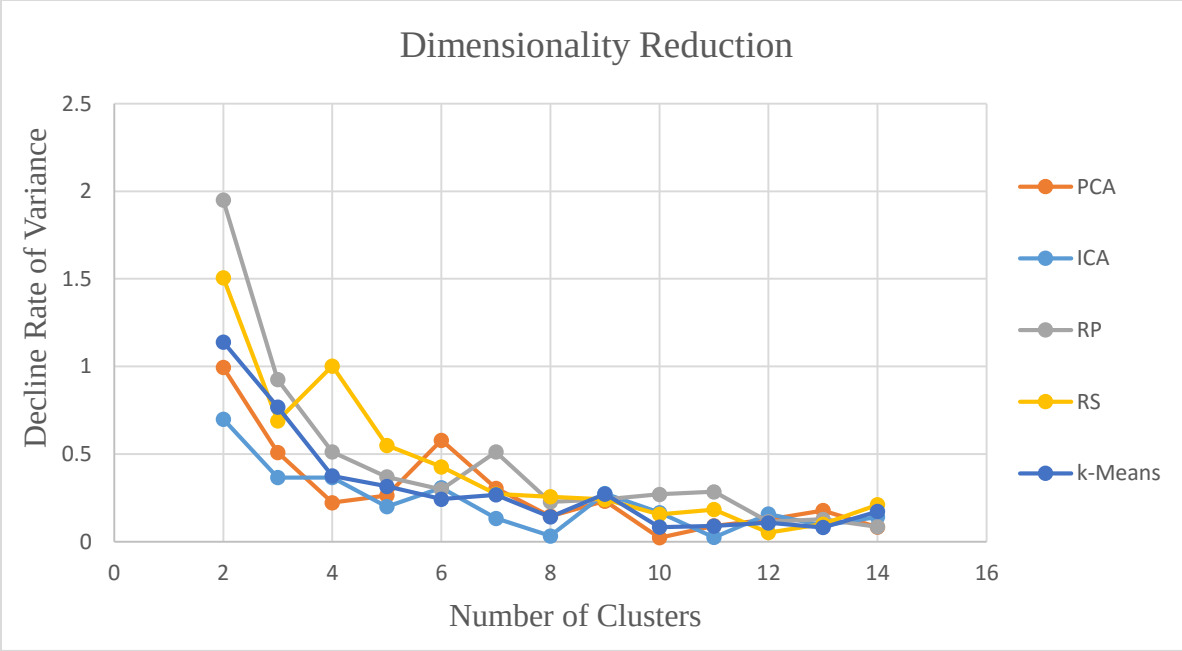
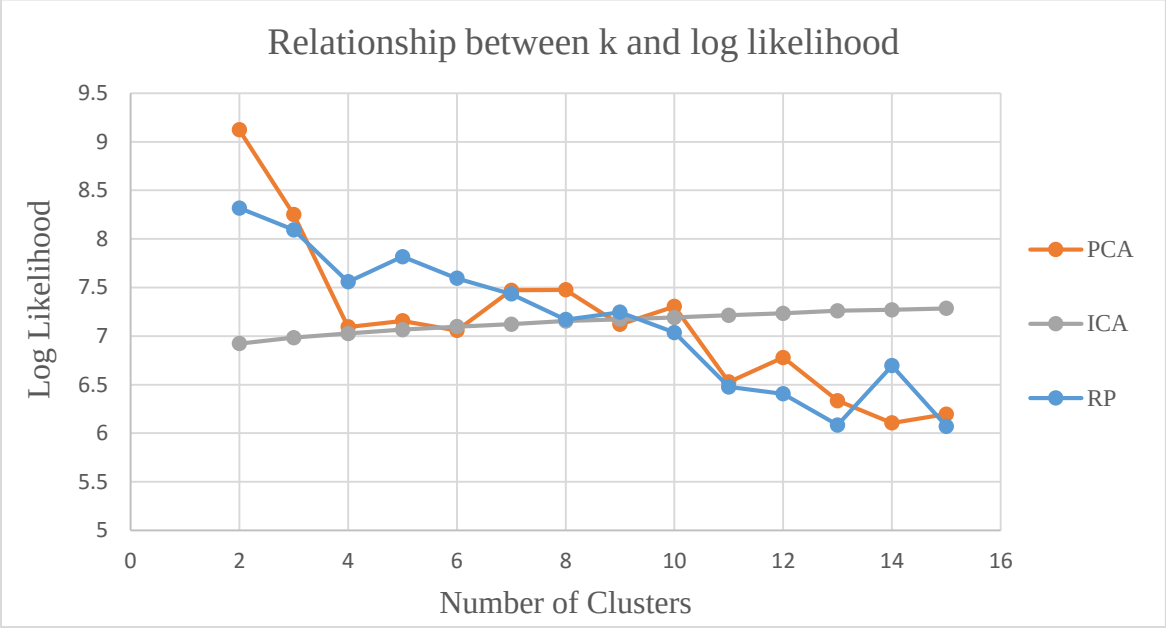


Figure 7. Relationship of Log Likelihood and Variance for Car Evaluation dataset



Wine Quality:

Figure 8. Within group SSE percentage for Wine Quality dataset

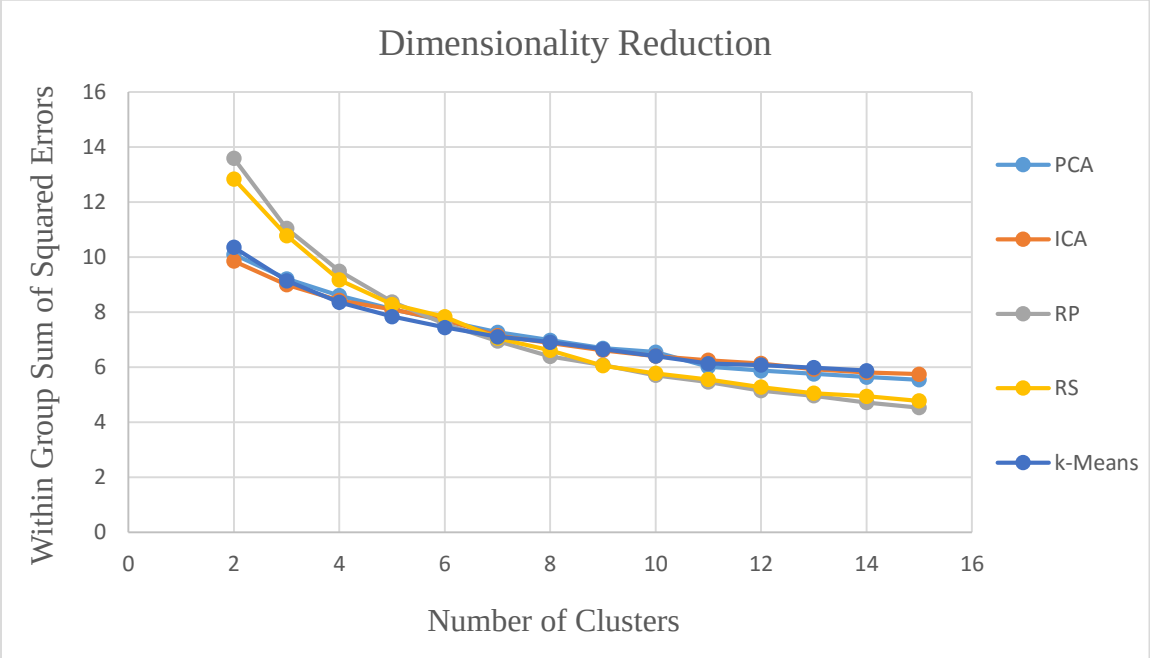


Figure 9. Decline Rate of Variance for Wine Quality dataset

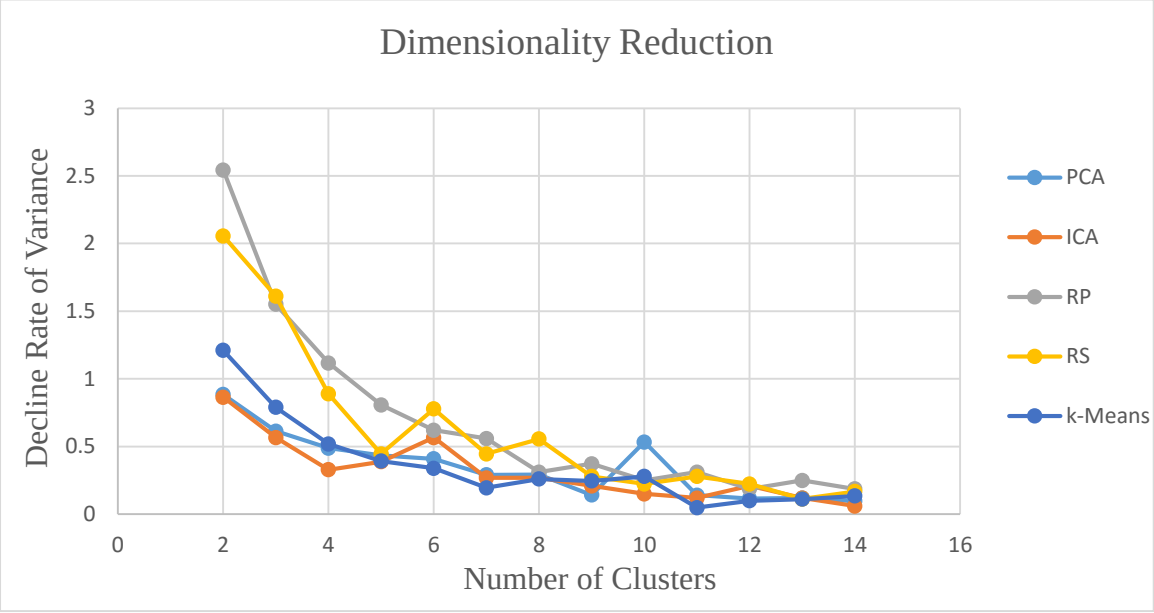
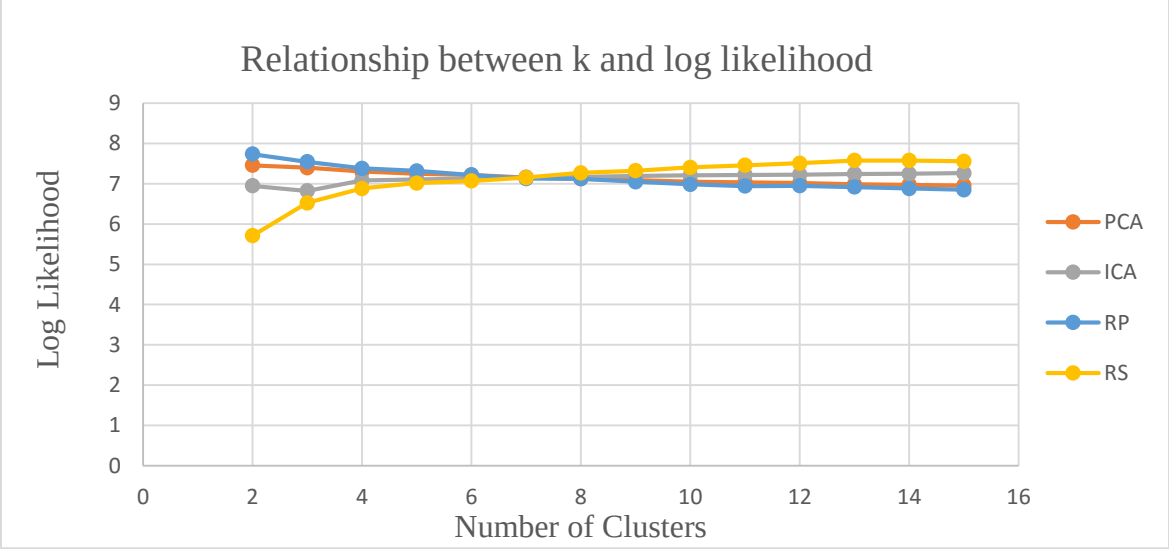


Figure 10. Relationship of Log Likelihood and Variance for Wine Quality dataset



Analyses:

(1) Compare and contrast the different algorithms. PCA assumed the samples are under Gaussian distribution and it performs better when the sample size is bigger. While ICA assumed each sample is independent. Since PCA requires all the components aren't related, and ICA requires all

the components to be strictly independent. Therefore, the constraints of ICA is stricter than PCA and the feature selection ability of ICA is better than PCA.

(2) In order to improve the performance of those algorithms, we might change the relative parameters of them. For example, in the previous case, I set the threshold for PCA, ICA etc. as 95%. And what if I set the threshold as 90% and 80%? By plotting a graph about the relationship of different threshold with variance or log likelihood, we can generate a better result about the optimal k.

5.4 Dimensionality Reduction + Neural Network (Wine Quality Dataset Only)

According to the part 5, I applied 4 dimensionality reduction algorithms to both datasets. After that I re-ran the neural network algorithm. Before the classification experiments start, the datasets will be split up to two parts, one is the training set, the other one is the testing set. (Roughly, training set will occupy 70% of the entire data and testing set will be 30 %.) In order to get a more accurate measurements, I will use 10 folds cross-validation to train and test my training data only while remaining the test set unseen by the classifier. Here are the results.

Table 4. Result of different DR algorithms

Neural Network	L = 0.3, W = 0.4	Training instances = 3429	Testing instance = 1469
PCA			
Incorrectly Classified Instances	41.70%	Incorrectly Classified Instances	45.34%
Mean Absolute Error	0.0991	Mean Absolute Error	0.1032
ICA			
Incorrectly Classified Instances	40.86%	Incorrectly Classified Instances	43.24%
Mean Absolute Error	0.0964	Mean Absolute Error	0.1019
RP			
Incorrectly Classified Instances	48.91%	Incorrectly Classified Instances	50.71%
Mean Absolute Error	0.1047	Mean Absolute Error	0.107
RS			
Incorrectly Classified Instances	45.99%	Incorrectly Classified Instances	48.67%
Mean Absolute Error	0.1039	Mean Absolute Error	0.1062

Analyses:

(1) According to the table above, we can easily conclude from the incorrectly classified instances that the ICA performs better than the other algorithms.

(2) Comparing the error rates of PCA and ICA, we can tell the distribution of eigenvalues is not too close to a Gaussian distribution, but it is much better than an independent distribution. Since the error rate of ICA is lower than PCA, and we can conclude that the distribution is close to non-Gaussian, the absolute value of kurtosis is bigger with more closer to non-Gaussian distribution.

(3) Random projection is a simple and computationally efficient way to reduce the dimensionality of data by trading a controlled amount of error for faster processing times and smaller model sizes. ICA constructs the dataset with stricter constraint than PCA.

5.5 Clustering + Dimensionality Reduction + Neural Network (Wine Quality Dataset Only)

According to the part 5, I applied 4 dimensionality reduction algorithms and 2 clustering algorithms to both datasets and add the cluster features in it. After that I re-ran the neural network algorithm.

Table 5. Result of different DR and clustering algorithms (k-Means)

Neural Network	L = 0.3, W = 0.4	Training instances = 3429	Testing instance = 1469
PCA + k-Means			
Incorrectly Classified Instances	36.28%	Incorrectly Classified Instances	45.27%
Mean Absolute Error	0.0859	Mean Absolute Error	0.0966
Time	0.26s		
ICA + k-Means			
Incorrectly Classified Instances	33.92%	Incorrectly Classified Instances	44.52%
Mean Absolute Error	0.079	Mean Absolute Error	0.093
Time	0.22s		
RP + k-Means			
Incorrectly Classified Instances	46.37%	Incorrectly Classified Instances	51.53%
Mean Absolute Error	0.1029	Mean Absolute Error	0.107
Time	0.15s		
RS + k-Means			
Incorrectly Classified Instances	42.26%	Incorrectly Classified Instances	48.47%
Mean Absolute Error	0.097	Mean Absolute Error	0.103
Time	0.12s		

Table 6. Result of different DR and clustering algorithms (EM)

Neural Network	L = 0.3, W = 0.4	Training instances = 3429	Testing instance = 1469
PCA + EM			
Incorrectly Classified Instances	31.81%	Incorrectly Classified Instances	37.10%
Mean Absolute Error	0.066	Mean Absolute Error	0.075
Time	0.08s		
ICA + EM			
Incorrectly Classified Instances	35.70%	Incorrectly Classified Instances	46.22%
Mean Absolute Error	0.081	Mean Absolute Error	0.0949
Time	0.15s		
RP + EM			
Incorrectly Classified Instances	46.17%	Incorrectly Classified Instances	50.10%
Mean Absolute Error	0.1034	Mean Absolute Error	0.107
Time	0.13s		
RS + EM			
Incorrectly Classified Instances	42.37%	Incorrectly Classified Instances	47.45%
Mean Absolute Error	0.097	Mean Absolute Error	0.1033
Time	0.10s		

Analyses:

(1) When I reproduced clustering experiments on the experiments on the datasets projected onto the new spaces created by ICA, PCA and RP, the clusters are different as before. In my opinion, the main reasons that might cause this problem is the dimensionality reduction algorithms will reduce some redundant features and noises and generate new clusters that based on the existing features or the linear combination of them.

(2) According to the previous assignment, the average time for 10 iterations with 100 hidden layer nodes is about 1.4s. And from the above table, we can tell the time average time is about 0.1875s ((0.26+0.22+0.15+0.12) / 4) which is almost 1/10 of the time that I got from previous assignment. The performance and speed are much better than before. The dataset size is not so big, however, it can tell that enhance of performance and speed will be much more significant if the dataset size is bigger.

(3) Comparing to table 4, we can obviously tell the error rate decreases while both the clustering and dimensionality algorithms are applied. And time elapsed for different algorithms is pretty close. But if the data size is getting bigger, it seems random projection and random subset will consume less time.

(4) Comparing to the previous assignment (See Appendix), the training and testing error rate obtained from table 5 are significantly decreasing. And as to the Wine Quality dataset with 11 attributes, the dimensionality reduction algorithms greatly reduced the redundant attributes and noises.

(5) From table 4 and 5, no matter in time consuming or error rate aspects, it seems EM algorithm works better than k-Means in this dataset.

6 Appendix

A1. Parameter optimization of dataset using MultilayerPerceptron algorithm (Car Evaluation)

Learning rate	Momentum	500	1000	1500	2000	2500	5000	10000
0.1	0.2	98.93	98.76	98.68	98.6	98.43	98.51	98.6
0.3	0	98.43	98.35	98.26	98.18	98.18	98.26	98.18
0.3	0.2	98.76	98.76	98.76	98.76	98.76	98.84	98.76
0.3	0.4	98.76	98.68	98.68	98.68	98.68	98.68	98.76
0.5	0.2	98.51	98.51	98.51	98.51	98.51	98.51	98.51

A2. Parameter optimization of dataset using MultilayerPerceptron algorithm (Wine Quality)

Learning rate	Momentum	500	1000	1500	2000	2500	5000	10000
0.1	0.2	54.97	54.77	54.21	54.27	54.71	55.5	55.96
0.3	0	54.59	54.88	55.41	55.29	55.35	-	-
0.3	0.2	55.35	55.5	55.06	54.53	54.65	-	-
0.3	0.4	55.99	55.76	55.82	56.02	55.88	-	-
0.5	0.2	54.54	54.42	54.24	53.98	54.36	-	-

A3. MultilayerPerceptron performance in 10 iterations

BP	Iterations = 10						Average
Training error (%)	82.35	80.08	84.95	58.99	71.85	55.66	72.31
Testing error (%)	81.22	79.39	86.12	59.25	74.49	55.99	72.74
Time elapsed (s)	1.32	1.44	1.20	1.69	1.47	1.29	**1.40**
Log10(s)	0.12	0.16	0.08	0.23	0.17	0.11	0.14