Assignment 3: CS 7641

Gandharv Kashinath

April 5, 2015

Introduction

In this assignment six algorithms were studied; two clustering algorithms and four dimensionality reduction algorithms. These algorithms were then used to construct the Neural Network (NN) for the two datasets used in Assignment 1. The algorithms were studied for performance and also to identify trends and nuances of these algorithms.

Data Description

Car Evaluation Data Set (car):

The following table summarizes the dataset;

Data Set Characteristics:	Multivariate	Number of Instances	1728
Attribute Characteristics:	Categorical	Number of Attributes:	6

The following data transformations were applied to the dataset to convert nominal attributes to numeric attributes and Figure 1 shows the data distribution after pre-processing;

Attribute	Nominal	Numeric Value
Buying	low	1
	med	2
	high	3
	vhigh	4
Maintenance	low	1
	med	2
	high	3
	vhigh	4
Doors	5more (>4)	5
Persons	more (>4)	5
Luggage Boot	small	1
	med	2
	big	3
Safety	low	1
	med	2
	high	3

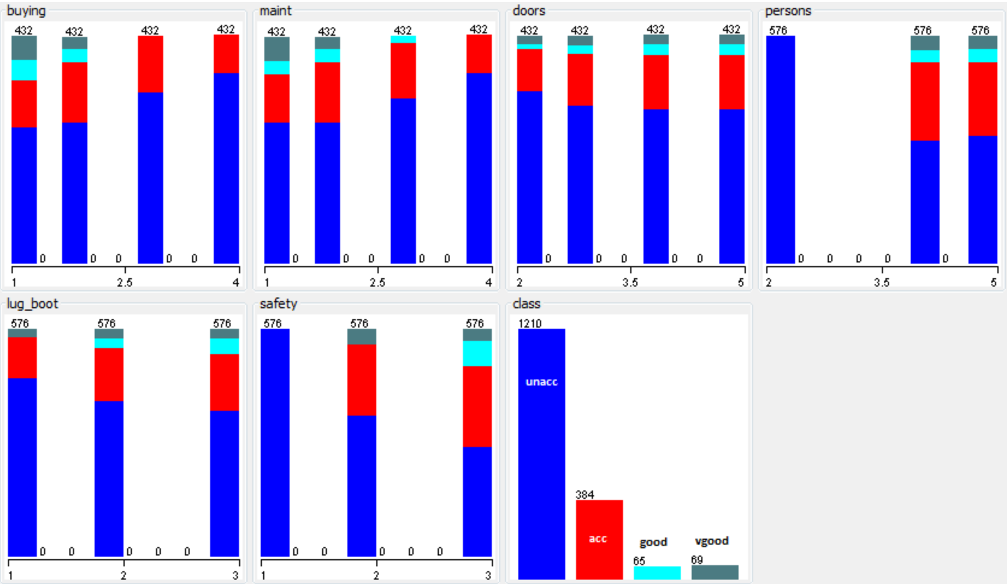


Figure 1: Car Evaluation dataset after pre-processing.

From assignment 1, trends like safety and persons played an important role in classifying the car as acceptable or not were observed from the classification algorithms. In this assignment, the question of clustering these cars in to the 4 categories ranging from 'vgood' to 'unacceptable' will be explored.

Glass Identification Data Set (glass):

The following summarizes the glass identification data set;

Data Set Characteristics:	Multivariate	Number of Instances	214
Attribute Characteristics:	Real	Number of Attributes:	10*

Type of Glass	Numeric Value
Building windows float processed	1
Building windows non float processed	2
Vehicle windows float processed	3
Vehicle windows non float processed	4
Containers	5
Tableware	6
Headlamps	7

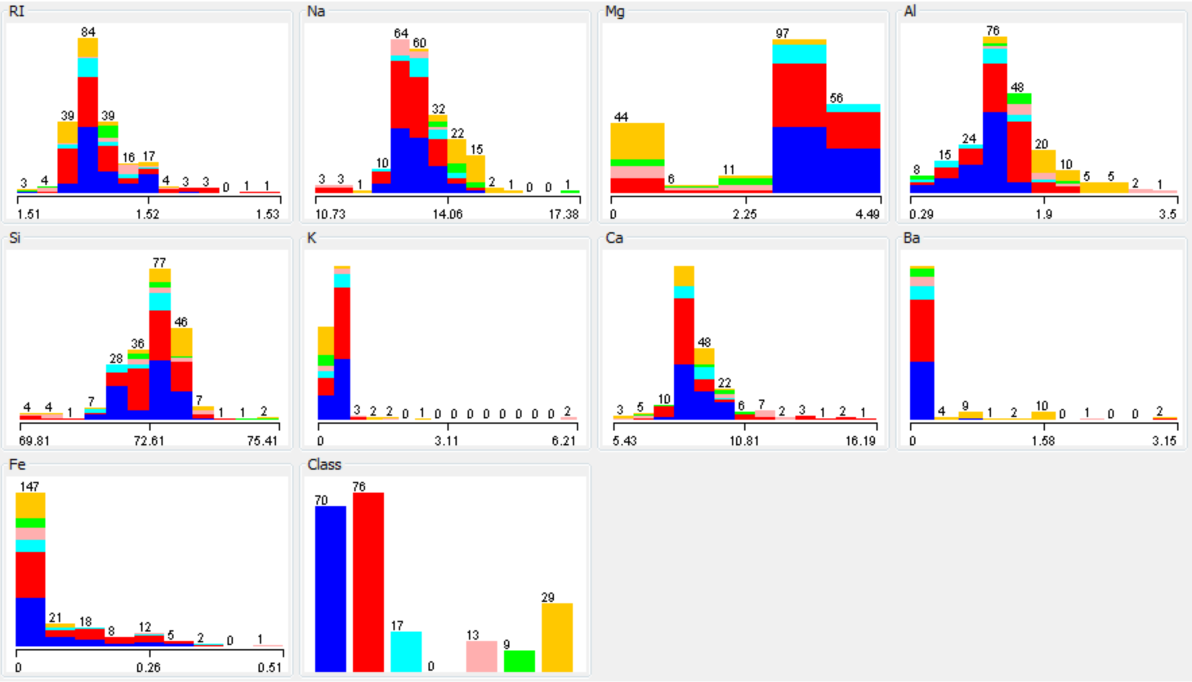


Figure 2: Glass Identification dataset after removing ID attribute.

In assignment 1, several interesting classification trends were observed. Refractive index, magnesium, aluminum and sodium content played an important role in classifying a glass in one of the 7 categories of glass. In this assignment the clustering of various types of glasses will be examined with the help of several algorithms.

Methodology, Algorithms and Tools

In this assignment 6 algorithms were studied. The first two are clustering algorithms:

- *k*-means clustering
- Expectation Maximization

The last four algorithms are dimensionality reduction algorithms:

- PCA
- ICA
- Randomized Projections

All the above mentioned algorithms are readily available on the *Weka* 3.7 data mining software and this tool was exclusively used to perform the experiments and analysis for this assignment. In order to generate plots and pre-process the data MS Excel was employed and all the worksheets are attached for reference. More information regarding the data sets (like why it is interesting and trends) can be found from the assignment 1 write-up.

Experiments, Results and Discussion

Clustering Algorithms

The two data sets used for this assignment were clustered using the *Weka*, *SimpleKMeans* and *EM* algorithm. Since these data sets have multiple attributes with several instances, visualizing the clusters was not trivial. In order to study the performance of these algorithms the timing information and log likelihood error reported by EM were noted and compared.

k-means clustering

k-means is a widely used clustering algorithm which aims to partition "*n*" observations into "*k*" clusters in which each observation belongs to the cluster with the nearest mean. This problem is NP-hard and hence *k*-means which is a type of heuristics algorithm can be employed to solve this problem.

There are several ways to compute the value of *k* for a given data set as shown [1, 2]. For this assignment the simple "elbow" method was employed and the results are presented. The elbow method studies the percentage variance, which can be computed by normalizing the intra-cluster distances between points in a given cluster and summing them across all clusters, as a function of *k*. The variance is measure of the compactness of the clusters. *k* was varied from two to the number of attributes in each data set. The distance/similarity measures were compared between the Euclidean and Manhattan distances and results were analyzed.

From Figure 3 and 4, the following observations and conclusions can be drawn;

- The elbow method provides **3 as the optimal *k***, for both datasets.
- The results for the Glass dataset is very distinct compared to the Car dataset and the variance is much lower. This shows that the *k*-means algorithm performs better for the Glass dataset as opposed to the Glass dataset.
- For both datasets, the Euclidean distance/similarity function is the better choice. In the means calculation the Euclidean distance provides the shortest distance between 2 points for both these datasets and since we are using the variance, the compactness of each cluster, to evaluate the performance of the algorithm this result was expected.

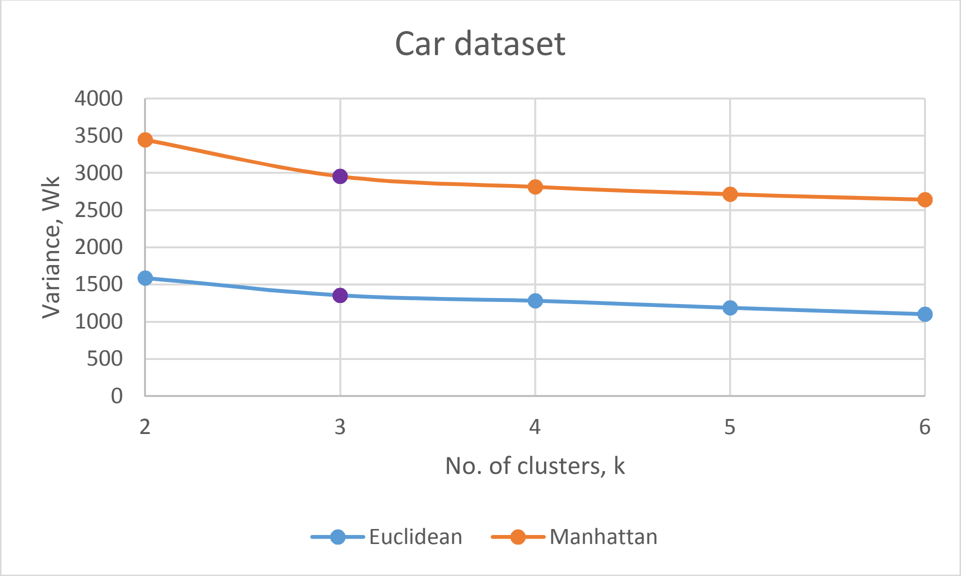


Figure 3: Variance as a function of *k*, for Car dataset (elbow method).

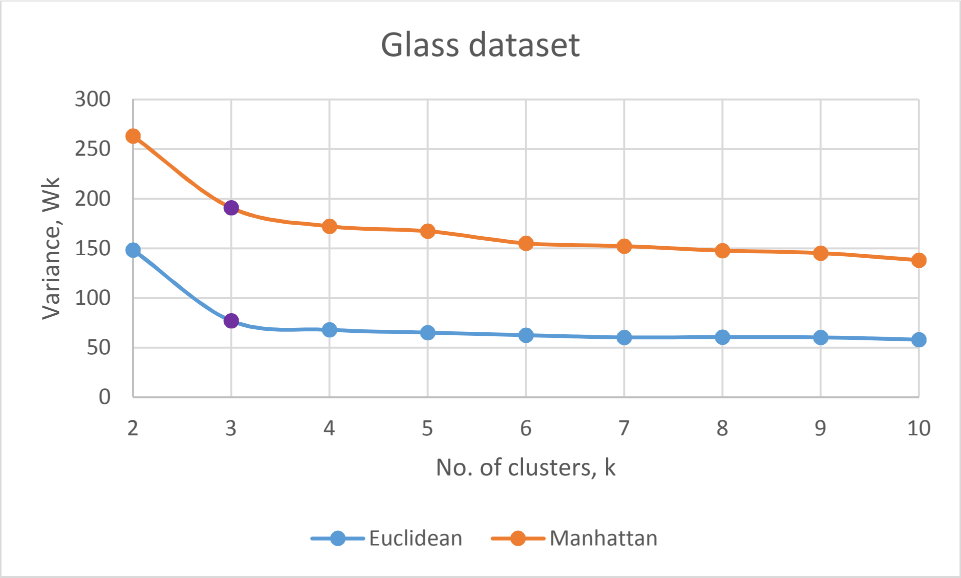
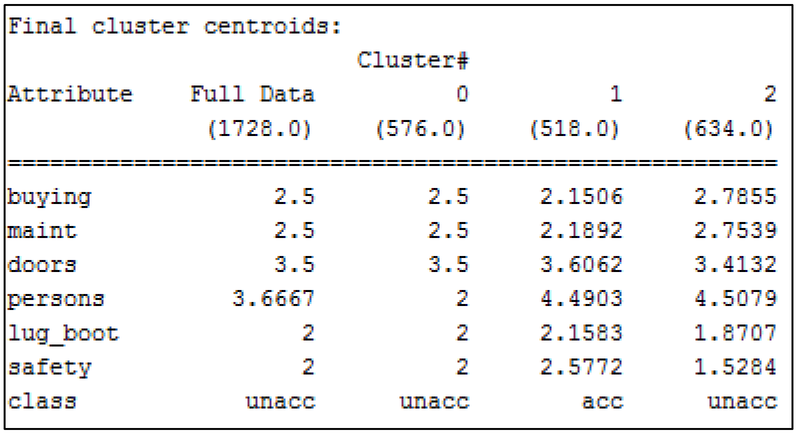


Figure 4: Variance as a function of k, for Glass dataset (elbow method).

For the car dataset, the most interesting clustering would be to use the attributes given in the dataset and create a cluster of acceptable (ranges from acc, good and vgood) and unacceptable cars. This was achieved by the k-means clustering algorithm. The *k*=3 cluster has the following final cluster centroids;

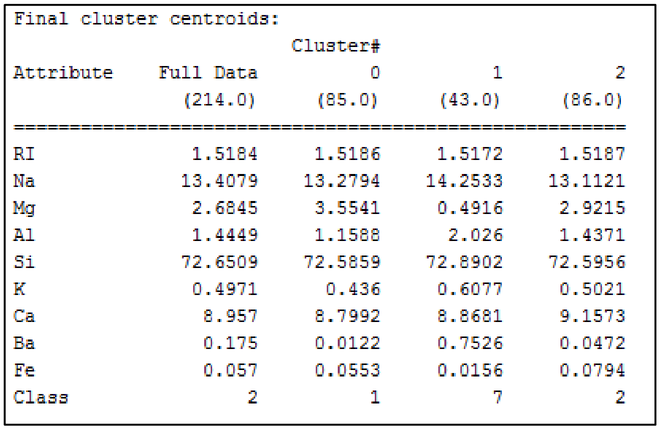


```
Final cluster centroids:
                           Cluster#
Attribute     Full Data           0          1          2
               (1728.0)     (576.0)    (518.0)    (634.0)
=========================================================
buying              2.5         2.5     2.1506     2.7855
maint               2.5         2.5     2.1892     2.7539
doors               3.5         3.5     3.6062     3.4132
persons          3.6667           2     4.4903     4.5079
lug_boot              2           2     2.1583     1.8707
safety                2           2     2.5772     1.5284
class             unacc       unacc        acc      unacc
```

Figure 5: *k*-means (*k*=3) cluster for car dataset.

From the above cluster, it can be concluded that a high safety rating was critical for all “acceptable” cars cluster. This was particularly interesting because intuitively in assignment 1 it was expected that this attribute would play an important role in classifying cars but it was the third most important attribute after buying and persons as predicted by the classification algorithms. The clustering algorithm has a distinctly high centroid value for safety thus reinforcing the initial intuition. Similar trends in lug_boot and persons were seen. Due to the dependencies on all the attributes, the complexity of the clusters are naturally observed and although this cluster can be used to make a clear distinction between acceptable and unacceptable cars, it will be rather challenging to find a cluster to further break down the acceptable group in to vgood, good and acc. A total of 70% of the instances belonged to the unacceptable cluster. The data distribution for the Car dataset is discrete and because various attributes are dependent on one another the clustering *k*-means algorithm struggles to find compact clusters. This can be seen in the magnitude of the variance,

For the glass dataset, the most instances of glass type 1, 2 and 7 (see table in data description for type description) were present. The cluster below with *k*=3, shows that these 3 types of glass were very well clustered using all the attributes. Aluminum and sodium have distinct centroids reinforcing their importance in classifying the instances for the Glass dataset. The variance for the Glass dataset was much lower than the Car dataset thus indicating bettering clustering. High calcium content in type 7 glass was captured well by the centroid by *k*-means.



```
Final cluster centroids:
                         Cluster#
Attribute     Full Data         0          1          2
                (214.0)    (85.0)     (43.0)     (86.0)
=======================================================
RI               1.5184    1.5186     1.5172     1.5187
Na              13.4079   13.2794    14.2533    13.1121
Mg               2.6845    3.5541     0.4916     2.9215
Al               1.4449    1.1588      2.026     1.4371
Si              72.6509   72.5859    72.8902    72.5956
K                0.4971     0.436     0.6077     0.5021
Ca                8.957    8.7992     8.8681     9.1573
Ba                0.175    0.0122     0.7526     0.0472
Fe                0.057    0.0553     0.0156     0.0794
Class                 2         1          7          2
```

Figure 6: k-means (k=3) cluster for car dataset.

Expectation Maximization

The expectation maximization algorithm alternates between performing an expectation (E) step, which creates a function for the expectation of the log-likelihood evaluated using the current estimate for the parameters, and a maximization (M) step, which computes parameters maximizing the expected log-likelihood found on the E step. These parameter-estimates are then used to determine the distribution of the latent variables in the next E step. *Weka's* EM algorithm which was used to study this algorithm, allows the selection of *k* automatically by cross-validation and hence this was used to construct the clusters for both the data sets. The automatically computed *k*, was compared with the results from the *k*-means algorithms (*k*=3).

Four clusters were generated using EM for the Car dataset. The log-likelihood for EM algorithm was -8.59. The EM algorithm did better than the *k*-means algorithm for the Car dataset as it was able to distinguish between good, vgood and acc. The attributes of the Car dataset is not continuous like the Glass data set and hence both the clustering algorithms did poorly for the Car dataset. Although the EM algorithm did better than the *k*-means, it took 22 times longer.

Four clusters were generated by EM for the Glass dataset. The centroids are comparable to the *k*-means algorithm but perform incrementally better with more distinct boundaries. The log-likelihood for EM was 2.94. The centroids of calcium, aluminum and barium were similar to *k*-means and from the class distribution, it can be concluded that type 1, 2, 3 and 7 are the most distinguishable types of glasses.

The EM algorithm should be more adept at handling data with overlap as it allows for "soft" membership of clusters. Figure 8 shows the Glass dataset clustering obtained from *k*-means and EM algorithms. Both these algorithms show very similar clusters and EM's 4 cluster arrangement is incrementally better than the 3 clusters obtained using *k*-means. Also, the EM algorithm took 3 times more time that the k-means algorithm.

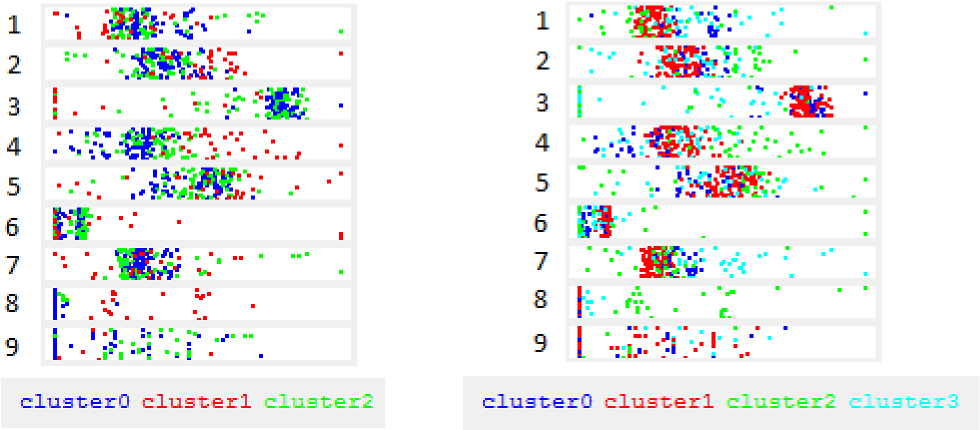


Figure 8: k-means (left) vs. EM (right) clustering for 9 attributes of Glass dataset

Due to the discrete nature of the attributes in the Car dataset the log-likelihood was higher compared to the Glass dataset. Better clustering can be achieved using continuous datasets.

Dimensionality Reduction Algorithms

In order to study the performance of the dimensionality reduction algorithms on the clustering algorithms the variance, W_k, for *k*-means and the log-likelihood for EM was recorded. Table 1 and 2 present these results for the Car and Glass datasets respectively.

Algorithms	**W_k**	**log-likelihood**
Original	1353.6431	-8.58617
PCA	1353.6431	-8.43427
ICA	397.48483	14.02377
Randomized Projections	928.62358	12.64171
Naïve dimension	1434.6431	-6.58617

Table 1: Dimensionality reduction and feature selection performance for *k*-means and EM clustering for Car dataset.

Algorithms	**W_k**	**log-likelihood**
Original	77.129723	2.94273
PCA	57.358773	-5.8365
ICA	66.424003	13.79598
Randomized Projections	72.1255	-12.63019
Naïve dimension	78.429753	4.94273

Table 2: Dimensionality reduction and feature selection performance for *k*-means and EM clustering for Glass dataset.

The Neural Network (Multi-layer Perceptron) algorithm in *Weka* was run on the pre-processed data and the following results (Table 3 & 4) for the two datasets was noted.

Algorithm	% Correctly classified	RMS error
Original	93.0556	0.1522
PCA	93.6343	0.1499
ICA	94.0394	0.145
Randomized Projections*	84.0278	0.2238
Naïve dimension	92.088	0.1622

Table 3: Neural Network with dimensionality reduction and feature selection performance for Car dataset.

Algorithm	% Correctly classified	RMS error
Original	69.158	0.2471
PCA	70.5607	0.2533
ICA	67.2897	0.2695
Randomized Projections*	69.1589	0.2618
Naive dimension	67.234	0.2989

Table 4: Neural Network with dimensionality reduction and feature selection performance for Glass dataset.

Independent Components Analysis (ICA)

After applying the ICA dimensionality reduction algorithm to the two datasets, the Gaussian like distribution of the attributes was noted. While all but 2 attributes are leptokurtotic for the Glass dataset, for the car dataset all the attributes were marginally leptokurtotic. This further indicates the dicrete nature of the Car dataset and when the ICA filter is applied the means become smooth. For the Car dataset ICA gave the best results with *k*-means clustering. This could be attributed to the fact that the Car dataset is discrete and ICA performs well with statistically independent data. For the same reason ICA performed poorly for the Glass dataset where the attributes are continuous and highly dependent on one another.

The impact of the performance of the dimensionality reduction algorithm was evident in the Neural Network (NN) training. For the Car dataset, ICA had the best performance compared to the other algorithms. It was able to classify 94% of the instances correctly using the 10-fold cross validation setting. Due to ICA's poor performance on the Glass dataset, the NN training was poor as well. ICA's accuracy increases with higher dimensions, but becomes saturated, i.e., more dimensions fail to improve accuracy. This was seen with the two datasets studied for this assignment.

Principal Component Analysis (PCA)

After applying the PCA dimensionality reduction algorithm to the two datasets, the best performance was seen on the Glass dataset. This was as expected due to the continuous nature of the attributes of this dataset. The orthogonal transformations was successful in linearly uncorrelating the attributes and hence provided the best performance. Due to this, the NN algorithm gave the best performance of 70% correctly classified instances for this filter.

The PCA algorithm performed poorly for the Car dataset. This was expected because all of the attributes of the Car dataset were discrete and did not have a continuous distribution to make the PCA algorithm effective in making them linearly uncorrelated. The relative scaling in the Car dataset was more even than the Glass dataset but still the PCA algorithm performed poorly. Applying a normalization to the attributes can make this better and can be explored further. The PCA algorithm's accuracy increases with higher dimensions, obtains a maximum and decreases rapidly with higher dimensions. This is evident from the fact that this algorithm performed better with the Glass dataset which had 10 attributes compared to the 6 attributes in Car dataset.

Randomized Projection (RP)

In randomized projections, the original d-dimensional data is projected to a *k*-dimensional (*k* << d) subspace through the origin, using a random *k* x d matrix *R* whose columns have unit lengths. The RP algorithm was run several times due to the randomization in the algorithm and the mean was used to evaluate the performance. Several variations in the runs were seen for both the datasets but the most was observed with the Glass dataset. This can be explained by the continuous data where variations can be seen more distinctly compared to the Car data set.

This algorithm performed poorly for both the datasets. The pairwise distance between two samples of the dataset was controlled by this algorithm and due to the randomization in constructing the projections the Car dataset performed the worst among all the other algorithms. The NN algorithm with this filer was only able to classify 84% of the instances correctly. The Glass dataset performance although worse than the others was better than the Car dataset.

Naïve Dimension

The naïve dimension reduction algorithm scales each attribute to form a distribution along the mean. This algorithm is best suited for datasets that are continuous. This algorithm is not as robust as other feature selection algorithms. This algorithm is downloaded from the *Weka* repository and is the faster among all the other dimensionality reduction algorithms. This can be used as a first order filter to study complex datasets with multiple attributes.

This algorithm performed in between all the other feature selection algorithms. It was able to successfully classify 92% and 67% of the instances correctly for the Car and Glass datasets respectively. This can be used to conclude that feature scaling is important for these datasets and just normalization is not enough for these datasets.

Clustering and Neural Networks

Finally, the clustering obtained from *k*-means and EM was used as additional attributes to construct the Neural Network in *Weka* and the following results (Table 5 & 6) were obtained for the two datasets. This was done by saving the clustering data obtained by *k*-means and EM algorithm and using that as the input for the NN classifier. The *Weka* experimenter module was used to perform these experiments. Training time was monitored to evaluate the performance of these modifications to the algorithm and compared with the original simulation.

Algorithm	% Correctly classified	RMS error
Original	93.0556	0.1522
k-means	91.0873	0.1802
EM	90.0233	0.1989

Table 5: Neural Network with clustering as dimensionality reduction performance for Car dataset.

Algorithm	% Correctly classified	RMS error
Original	69.158	0.2471
k-means	71.2358	0.2291
EM	72.9897	0.2102

Table 6: Neural Network with clustering as dimensionality reduction performance for Car dataset.

The clustering algorithm when used as a feature selection algorithm for the Neural Network classification worked well for the Glass dataset. It was able to improve the performance over the original classification to 73% with EM as the filter. This was expected because the clustering algorithm overall performed better with the Glass dataset than the Car dataset. The performance of this methodology was better than just using feature selection directly for the Glass dataset. The soft assignment of clusters through the EM algorithm works well for classification. The hard assignment of *k*-means worked better for the Car dataset and it outperformed although, marginally, the EM algorithm.

Dimensionality reduction helped in reducing the time to classify instances. An improvement of 6x in performance was achieved on the Car dataset and 4x on the Glass dataset. This was as expected since the number of attributes were reduced from the original dataset. An interesting aspect of this classification methodology is that an unsupervised clustering algorithm used for dimensionality reduction may introduce bias in to the dataset which may or may not be picked up by the classification algorithm. This aspect needs more investigation and should be explored in the future.

Conclusion

In this assignment six algorithms were studied. The two unsupervised learning algorithms were studied for performance and four dimensionality reduction algorithms were incorporated to evaluate the effects of feature selection in clustering and classification problems. Several trends were identified and key observations were outlined.

References

[1] https://datasciencelab.wordpress.com/2013/12/27/finding-the-k-in-k-means-clustering/

[2] http://en.wikipedia.org/wiki/Determining_the_number_of_clusters_in_a_data_set