Enmao Diao CS4641
Project 3

Unsupervised Learning Report

DATASETS

Wine Quality White Dataset. The goal is to model wine quality based on physicochemical tests. Although there are just 6 and 7 categories in these two datasets. However, the class distribution is extremely skewed. The quality scores in *Wine Quality White Dataset* range from 3 to 9. Low and high Quality scores have very limited amount of data samples. The complete *Wine Quality White Dataset* contains 4,898 instances and 12 attributes.

Mice Dataset. The data set consists of 78 Values of expression levels of 77 proteins that produced detectable signals in the nuclear fraction of cortex and 3 other binary attributes. The dataset contains a total of 1080 measurements per protein. Each measurement can be considered as an independent sample/mouse. The eight classes of mice are described based on features such as genotype, behavior and treatment. Attributes information for these two dataset can be found in the data.info file (see README.txt).

WHY INTERESTING

The reason I choose these two datasets is that they have complementary characteristics which can help on analysis. The wine dataset have an interesting class distribution. As mentioned above, there are only 1~3% instances labeled as quality 3 and 4 in the datasets. In multiclass classification, it is common to encounter a situation that there is very limited amount of data for a specific class. The mice dataset, however, has equally distributed labels. The wine dataset has a small amount of dimensions but most of them can be interpreted with some basic chemistry. On the other hand, the mice dataset has totally 80 dimensions so we can show the effect of feature reduction, but it is hard to interpret protein modification signals. These different properties can contribute to data analysis on clustering and feature selection.

APPROACH

In order to have a clear view of clustering algorithms. We first study the optimal number of clusters for both K-Means (KM) and Expectation Maximaztion (EM). We will then describe what kind of clusters we retrieve from these two algorithms. Then we evaluate the performance of each clustering algorithm by computing the number of instances that are not clustered correctly based on the given labels. We also evaluate four dimensionality reduction algorithms including Principle Component Analysis (PCA), Independent Component Analysis (ICA), Random Projection (RP) and Random Subset (RS). We will dig into the eigenvalues and eigenvectors from PCA and also compute the kurtosis for ICA. We will then fix the number of clusters for RP and RS to the optimal number of clusters from PCA in order to show the tradeoff between computation and accuracy. We will show the variations of RP and RS from several re-rans. Next, we will run clustering algorithms upon these four dimensionality reduction algorithms and compare the result to those not using these dimensionality reduction algorithms. Finally, we will evaluate the performance of each combination of clustering and dimensionality reduction algorithms with Neural Network (NN).

CLUSTERING

K-MEANS

We use the elbow method to determine the proper K for KM. We compute the sum of squared errors within cluster for each K value and the result is shown in Figure 1. For only KM, the optimal K for the wine dataset is about 15 and for the mice dataset is about 10. The optimal K value for the mice dataset is close to the number of classes which is 8. However, for the wine dataset, the number of clusters is more than the number of classes. This is due to the notion of wine quality is a kind of subjective categorization instead of scientific classification like in the mice dataset. The performance (incorrectly clustered instance) of KM on these two dataset is shown in Figure 5. As you can see, the number of incorrectly clustered instance keeps increasing for the wine dataset which means that the clustering does not make sense in this case because it cannot line-up with the subjective labels in the first place. Figure 4 also shows that there is no clear line up between the clustering and original labels for the wine dataset. On the other hand, the result of the mice dataset shows that when K equals 10, clustering yields the best performance. This means that the clustering is approximately line up with the labels at K equals to 10. Figure 3 shows that the clustering has a good lineup with the original labels. This is because the relationship between the observable labels and hidden variables is well described by the attributes of the mice dataset. The observable labels of the wine dataset, however, are not fully described by the attributes that the data provide. Obviously, wine tasters have their own standard of labeling which in fact has little correlation with physicochemical signal like free sulfur dioxide and pH. This results in the different performance of the K-means on the two datasets.

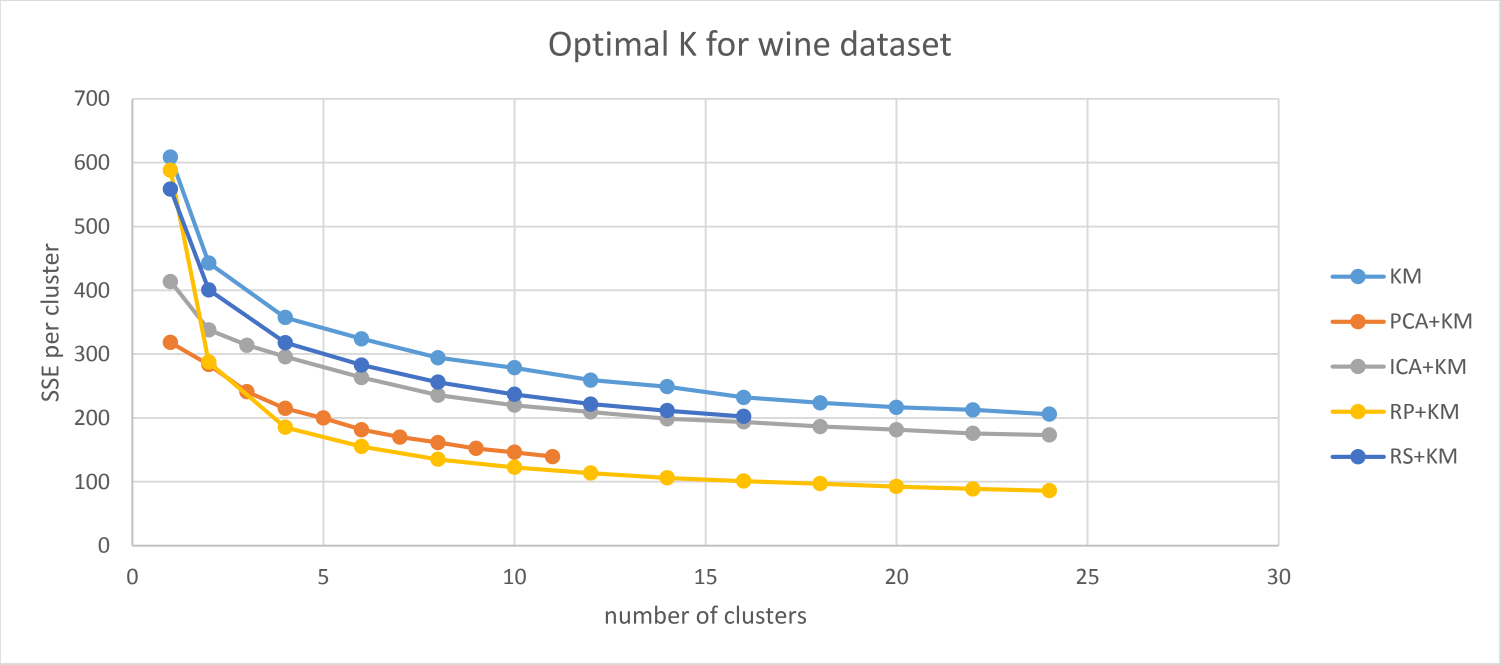

Figure 1. Optimal K for the wine dataset.

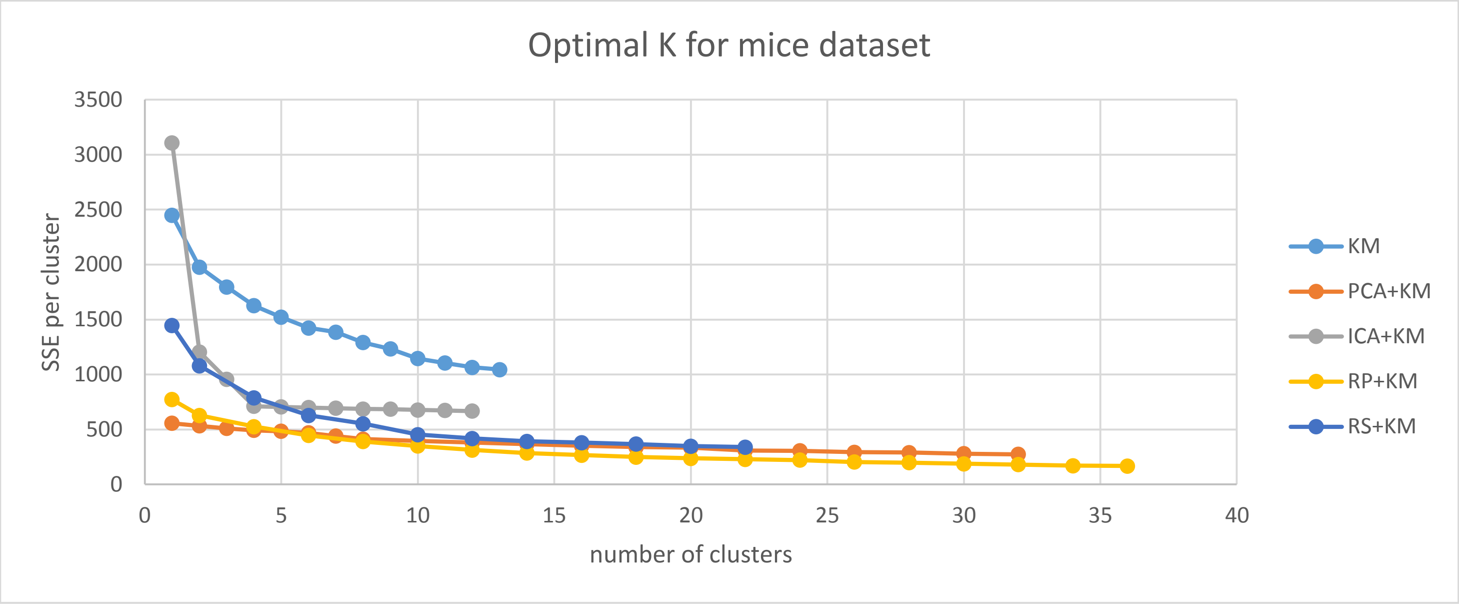

Figure 2. Optimal K for the mice dataset.

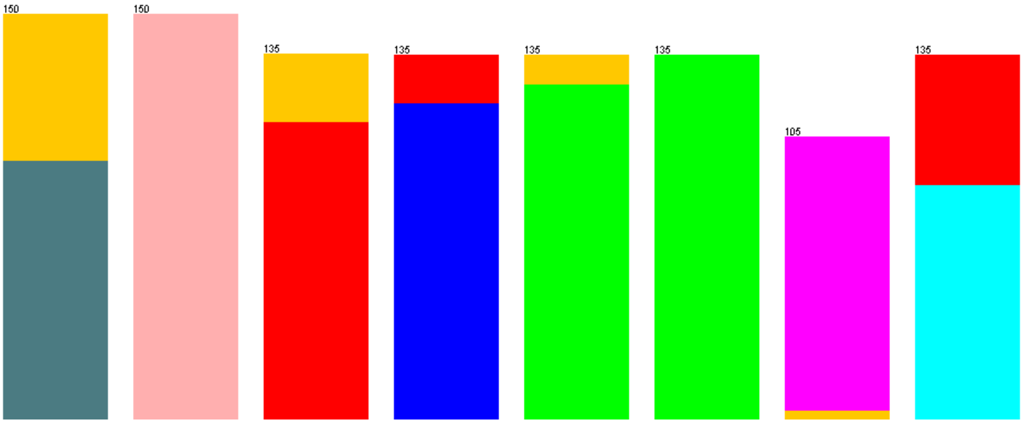

Figure 3. KM clustering distribution of the mice dataset.

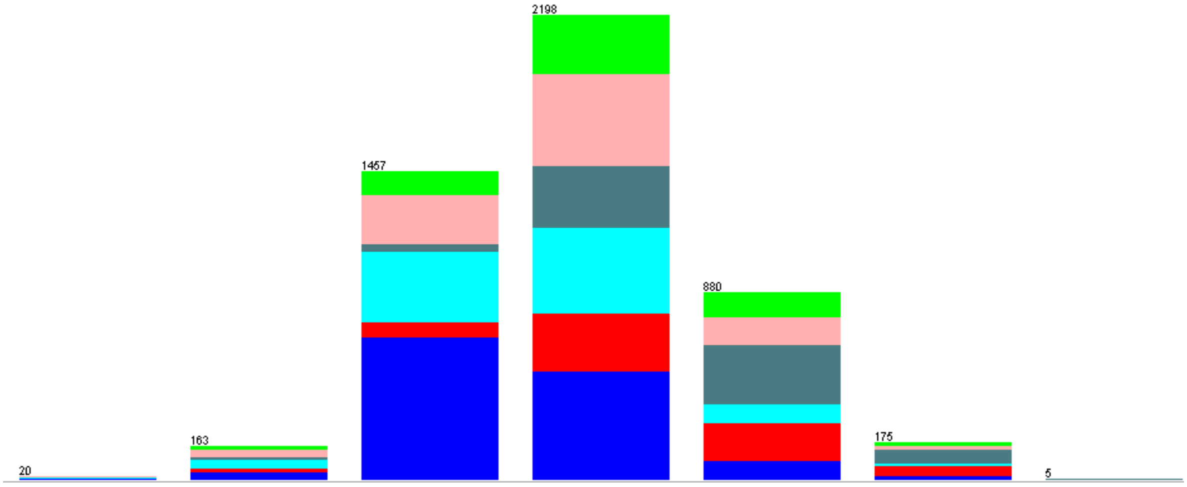

Figure 4. KM clustering distribution of the wine dataset.

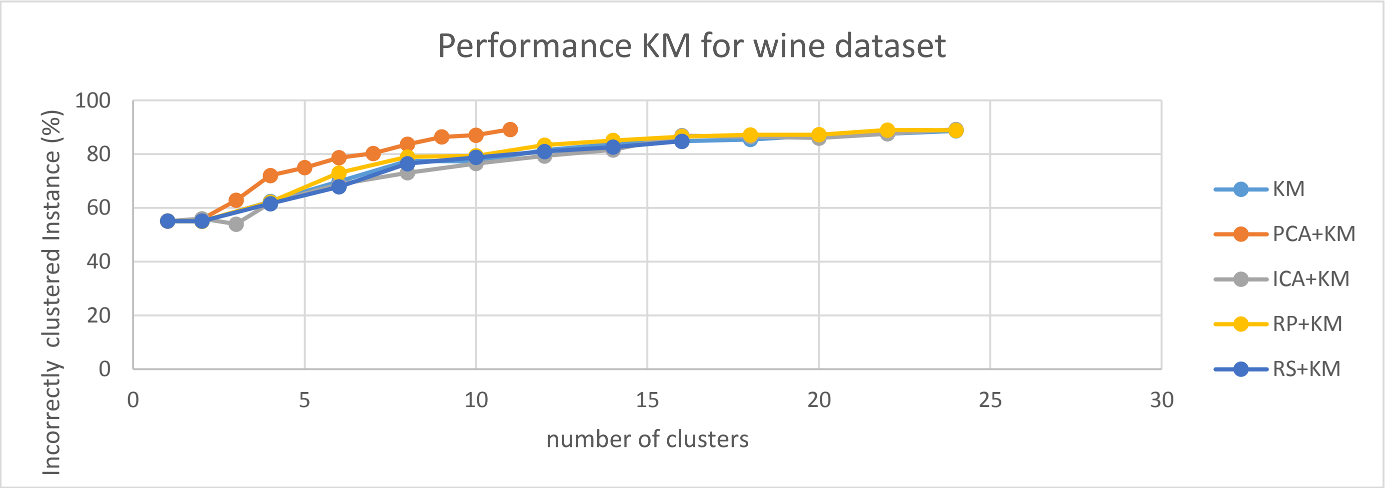

Figure 5. KM performance for the wine and mice dataset.

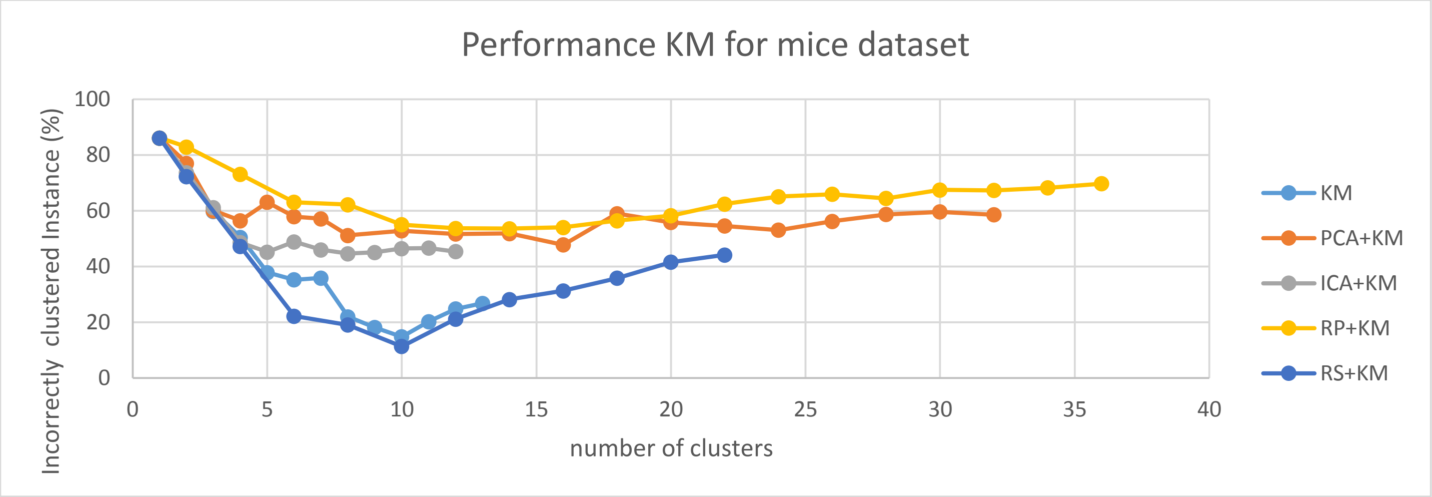

Figure 6. KM performance for the wine and mice dataset.

TABLE I

EM RESULT FOR WINE AND MICE DATASET

	Wine dataset					Mice dataset				
	EM	PCA+EM	ICA+EM	RP+EM	RS+EM	EM	PCA+EM	ICA+EM	RP+EM	RS+EM
Number of Clusters	21	17	22	20	13	9	29	9	35	16
Incorrectly Clustered Instance (%)	85.4226	82.9931	87.3826	89.0568	81.6456	28.3333	65.8333	37.963	59.537	37.3148

EXPECTATION MAXIMAZTION

We use cross-validation to choose the number of clusters for EM. The result is shown in Table I. The cross-validation method returns the optimal number of clusters at the point where the EM returns the maxima log likelihood and maximal likelihood algorithm. The result here is similar to that from KM. The optimal number of clusters of EM on mice dataset is 9 which is similar to that of KM. However, the performance is not exactly the same. EM on mice dataset yields 28.3% but KM on mice dataset can achieve 18%. As for the wine dataset, both clustering algorithms seem to fail. The reason, I believe as stated before, is the notion of human labeled wine quality are not fully described by the signals. Generally, KM performs better than EM in terms of incorrectly clustered instance on the mice dataset. This is because the boundary of the labels in the mice dataset is more suitable for hard clustering like KM. The eight classes of mice are described based on features such as genotype, behavior and treatment which are exactly the features of this dataset. Meanwhile, EM also takes more time to finish than KM does because it is difficult to determine expectations and we use a simple distance function in KM.

In order to improve the performance of two algorithms on the mice dataset, we can use Minkowski distance function given higher p value. This will help on resolving the large amount of protein modification signals. Meanwhile, we can also use the elbow method on EM so that instead of using the model with the highest log likelihood, we can use the number of clusters at the point where the likelihood starting to converge. This will improve the run time of EM since cross-validation can takes a long time to finish. As for the wine dataset, there is really not much we can do about with the algorithms. In order to have better clustering performance, we need to have more dimensions, for example, the standard that the wine tasters use to determine the quality of wines, so that we can enough data to describe the relationship between the subjective labeling and hidden variables.

DIMENSIONALITY REDUCTION

PRINCIPLE COMPONENT ANALYSIS

The specific distribution of eigenvalues and eigenvectors is shown in the raw output folder of supporting files folder (see README.txt). Here, I raise some examples of eigenvalues and eigenvectors to show how PCA combines different features together. The eigenvector of the first principle component for the wine dataset is [-0.1572 -0.0051 -0.144 -0.4274 -0.212 -0.3003 -0.4067 -0.5115 0.1288 -0.0434 0.4372] and its corresponding eigenvalue is 3.22225 and for reference the eigenvalue of the second component is 1.57524. Here, the size or the absolute value of the eigenvector shows the weight of each feature of this component. In the first component, alcohol is the most positive and free/total sulfur dioxide, density, and residual sugar is the most negative. This means that alcohol is inversely related to the combination of these negative features. As we know that, fermentation takes sugar and other inorganic substances into organic substances like alcohol. This process may also purify then filter out large substances and as a result the density also decreases. There are some other components, for example pH and acidity is also shown as inversely related in the second component which is consistent with our chemistry knowledge. PCA, therefore, combines the features to maximize the variance of data and also minimizing information loss by using mutually orthogonal components. Each eigenvector is approximately orthogonal to each other and indeed they represent realistic linear relationship between different features. We can evaluate how well we can reconstruct the data by observing the orthogonality of these eigenvectors.

When running Clustering algorithms on PCA, we denote them as PCA+KM and PCA+EM as shown in Figure 1. After running PCA, the number of features of wine and mice dataset are respectively 9 and 32. This shows a significant dimensionality reduction for the mice dataset however the performance also degrades significantly. The clusters we have from KM and EM after PCA are different from before. As shown in Figure 2, the elbow method shows that PCA reaches optimal K at the beginning. This is due to the maximized variance of each feature prevent effective clustering since each data point now is far apart in the space. Even though we have more and more clusters, the variances within the clusters are about the same. As shown in Figure 6 and Table I, PCA performs badly for both datasets. Since clustering does not make sense for the wine dataset, we focus on the mice dataset. Although we have significantly decreased the number of dimensions, we also yield the worst performance now. I guess this is simply due to the fact that labels are not indicated by the interrelations of the features. Even though we can understand the eigenvectors with some chemistry knowledge, this does not improve the performance because the labels are in fact not strongly related with these physicochemical signals and their interrelations.

INDEPENDENT COMPONENT ANALYSIS

ICA which is fundamentally different from PCA maximize the independence between features. It attempts to find the hidden variables that are mutually independent of each other. We can use kurtosis (fourth central moment) to rank the features. The specific kurtosis values are shown in Analysis.xlsx (see README.txt). The maximal kurtosis of the wine and mice dataset are respectively 14.00772 and 6.95296. The features from the wine dataset have higher kurtosis values because this dataset has only 11 dimensions while the mice dataset has up to 80 dimensions. Since there are more observables for the mice dataset, it is hard for ICA to separate independent components. In the wine dataset, only 11 independent components are separated and therefore some of them are very dominant in terms of probability distribution skewedness. I do believe ICA captures some meaningful projections for the wine dataset, since the features in the wine dataset do share some linear relationship for example pH and acidity, sugar and alcohol. These features should be able to be separated into statistically independent components and kurtosis also justifies that. However, I believe there is not any linear relationship between protein modification signals. They may be correlated in a very sophisticated way but it cannot be separated into dependent components. Therefore, it is hard for ICA to separate hidden variables from observables in the mice dataset.

When running KM and EM on ICA, the optimal number of clusters is smaller than before as shown in Figure 2, but the performance degrades from 20% to 40% as shown in Figure 6. This is because when we separate hidden variables from observables, since they are linearly independent, we also need fewer clusters to clustering theses variables. When we use ICA, we also might lose some information from the observables. In this case, there are three binary features that have a strong impact on the labels. After ICA, these features are transformed into hidden variables which in fact may have less impact on the labels. Therefore, the performance decreases and we need less number of clusters when using ICA on the mice dataset.

RANDOM PROJECTION

In order to show the variansions of RP, we use different seeds to generate RP and evaluate their performance with EM. Then we select the one that has the best performance as our model for RP. In fact, the performance evaluation for KM after RP is conducted on the model that is selected from EM after RP. Variations of RP are shown in Table II. Since RP is an simple alternatives of PCA, its main advantage is computation efficiency. However, since the dataset is not big enough, it is hard to compare its run time to that of PCA. Since RP works extremely well for classification problems, we can evaluate its performance not only based on incorrectly clustered instance but also classification accuracy from certain learners. As a rule of thumb, we use Naïve Bayes (NB) to evaluate the performance of each RP variantion as shown in Table II. Generally, these two metrics are consistent with each other. Lower incorrectly clustered instance also yields higher NB accuracy.

The result of running KM on the RP model we choose from RP+EM is shown in Figure 1 and 2. As expected, RP performs very similar to PCA. They both converge at the elbow method at the beginning and yield bad performance on the mice dataset. After all, RP just randomly selects projections by picking up correlations instead of maximizing variances and minimizing information loss. In this case, we can hardly evaluate its biggest advantage which is its run time. However, we do show the run time statistics in later sections.

TABLE II

RP VARIANTIONS FOR WINE AND MICE DATASET

	Wine dataset					Mice dataset				
Seed	5	10	20	**40**	42	5	10	20	**40**	42
Incorrectly Clustered Instance (%)	93.2626	94.2221	92.3234	89.0568	93.6096	63.2407	65.1852	69.4444	63.7037	59.537
NB Accuracy (%)	34.8305	40.3838	43.3238	43.6301	34.218	93.5185	88.1481	77.963	91.1111	90.8333

RANDOM SUBSET

Here we introduce a rule of thumb for dimensionality reduction. RS simply selects a subset of the set of features without merging or separating and features. The new set of features is purely from the orignial dataset. Basically, this approach just deletes some columns in the data. We use a similar mechanism in RP to select the model that has the best performance from EM. Then we conduct KM on the model we select and the result is shown in Figure 3 and 4. The variations are shown in Table III. The main advantage of this method is that it aggressively decreases the problem space by randomly reducing features. This approach may be helpful when given a huge dataset that has many unneccssary or interfered attributes. Through experimental test, we can shrink the problem space and avoid the curse of dimensionality instantly. The result of running KM on our selected model is shown in Figure 3 and 4. Interesting, although RS solves the same problem as the original data, it acutally performs better than the original data. Obviously, there is definitely some useless data in the mice dataset. As you can see from Figure 1, the sum of squared error within cluster is uniformly smaller than using pure KM. Moreoever, the performance evaluation in Figure 4 shows that it also performs slightly better than pure KM in terms of incorrectly clustered instance.

TABLE III

RS VARIANTIONS FOR WINE AND MICE DATASET

	Wine dataset					Mice dataset				
Seed	5	10	20	40	**42**	5	10	20	**40**	42
Incorrectly Clustered Instance (%)	82.1151	83.2585	82.9318	81.6456	84.1364	55.463	67.5	59.8148	37.3148	76.3889
NB Accuracy (%)	43.0584	41.3434	44.4875	47.4275	42.1805	86.9444	85.1852	88.6111	97.6852	73.0556

CLASSIFICATION

TABLE IV

CLASSIFICATION ON COMBINATION OF CLUSTERING AND FEATURE TRANSFORMATION

	Wine dataset				Mice dataset			
Algorithm	WINE	**PCA**	**ICA**	RP	MICE	PCA	ICA	RP
NB (%)					90.9259	96.3889	95.0926	90.8333
NN (%)	52.8251	54.3227	**55.8203**	51.1232	100	100	100	100
Runtime(s)	9.54	**8.06**	12.28	8.43	33.37	8.48	31.7	8.58
Algorithm	RS	KM	EM		**RS**	KM	EM	
NB (%)					97.6852	93.5185	90	
NN (%)	53.5739	55.548	53.2335		100	100	100	
Runtime(s)	8.6	12.29	12.67		**7.61**	35.6	34.24	
Algorithm	PCA+KM	ICA+KM	RP+KM	RS+KM	PCA+KM	ICA+KM	RP+KM	**RS+KM**
NB (%)					96.8519	95.1852	89.7222	**99.537**
NN (%)	53.4377	54.7992	52.0082	53.5739	100	99.6914	100	100
Runtime(s)	12.18	12.29	11.14	11.51	9.39	35.3	9.27	9.88
Algorithm	PCA+EM	ICA+EM	RP+EM	RS+EM	PCA+EM	ICA+EM	RP+EM	RS+EM
NB (%)					95.463	96.8519	89.2593	97.4074
NN (%)	53.0973	54.4588	51.9401	52.0762	100	100	100	100
Runtime(s)	11.62	12.14	11.32	11.41	9.51	33.63	9.38	9.45

In this section, we conduct a thorough comparative performance analysis for every combination of clustering and dimensionality reduction algorithms. As shown in Table IV there do exist some combinations of two algorithms that outperform the classification upon the original dataset. Due to the fact that almost all combinations can achieve 100% accuracy with Neural Network. We analyze the performance with respect of accuracy on the mice dataset by using Na ïve Bayes learner. We also analyze the run time statistics with Neural Network learner.

In the wine dataset, as you can see, the accuracy does not vary much. We achieve the highest accuracy with pure ICA. This is probably because the wine dataset only has 11 dimensions. As a result, ICA transforms the feature into highly independent components and some of them have very high kurtosis value.

The most efficient approach for the wine dataset is pure PCA. It only takes 8.06s to one of the highest performance within all possible combinations. Although RP and RS, especially RS, seems to have extreme advantage in computation. However, due to the small scale of problem space, their computation superiority cannot be revealed. PCA in this case, reduces the number of features and therefore improves the runtime. Meanwhile, run time and accuracy also prove our previous claims that clustering does not make sense for this dataset. When using any kinds of clustering algorithms for this dataset, the accuracy will drop and it will take more time to finish as the complexity of the problem grows. If the previous features are replaced by the cluster feature, the performance will be largely degraded so we do not list in the table here.

In the mice dataset, since the three binary features greatly contribute to the classification problem, NN can easily return 100% accuracy. Therefore, we use the rule of thumb NB to help us determining the algorithm that yields the best performance. As you can see from the table, RS outperforms other algorithms. This is basically because many of the protein modification signals are not useful to the classification problem. They may not even be relevant for the labels. Therefore, RS aggressively reduces the number of dimensions and wipes out a large number of useless features. The learner not only performs better when having less useless features but it also runs faster than other algorithms. If give a huge dataset with a large number of irrelevant data, it would be better to use RS first to aggressively and experimentally eliminate such interferences.

CONCLUSION

In summary, we have analyzed the performance and reasoning of clustering and feature transformation algorithms on the wine and mice dataset. The clustering algorithms are not suitable for the wine dataset. With PCA we can perceive the interelations of its features. With ICA, we can effectively separate the features into independent components. This can help on classification problems later. There are dominant features inside the mice dataset that highly correlated with the labels. There are also some useless or even irrelevant features and data inside this dataset. Therefore, RS outperforms other algorithms in terms of both accuracy and run time.