# CS 7641 Machine Learning<br>Robert Bobkoskie \

# Assignment 3: Unsupervised Learning and Dimensionality Reduction

## 1. INTRODUCTION

The goal of this project is to explore the Unsupervised Learning (UL), Clustering: K-Means, Expectation Maximization, and Dimensionality Reduction (DR): Principal Component Analysis (PCA), Independent Component Analysis (ICA), Randomized Projections (RP) and Non-Negative Matrix Factorization (NMF). Unsupervised learning is used to infer some representation of unlabeled data. Clustering groups observations together, showing homogeneity in the data. Both clustering, when applied as a 'feature learning' step, and DR can used for reducing the amount of data. DR reduces the number of dimensions, and is particularly useful in addressing the 'curse of dimensionality', where higher dimensionalities require more training data.

This project will explore clustering in both contexts: inference and feature learning, and DR for feature reduction. It should be noted that unsupervised learning does not have labeled data, thus accuracy, as used in supervised learning is not a relevant metric. Since this project utilized problems that had labels, they were used only to predict and compare accuracy for the unsupervised learning models employed in this project. The clustering, DR and classification algorithms were implemented using scikit-learn[1] (sklearn). Graphing and dataset preprocessing was accomplished using python: pandas, numpy and matplotlib.pyplot.

## 2. Problems

For this assignment, two problems from the first assignment were used. The first problem was the built-in IRIS data set that came with sklearn. The IRIS data set is well known, thus, I will not deeply discuss this data set. Briefly, the data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. There are four attributes (features): sepal length (cm), sepal width (cm), petal length (cm) and petal width (cm). The objective (problem) is to classify (predict) the species of iris based on the four attributes.

The second problem[2] (SPAM) was also obtained from the `UCI Machine Learning Repository`. The data consists of a collection of attributes from both spam and non-spam (ham) email: 48 features related to word counts and 9 features associated with chars and strings of capital chars. The objective is to classify the e-mail as spam (1) or ham (0).

To minimize issues with feature scale, the data for all problems were normalized and scaled to values between zero and one. This data transformation was applied to all problems and used for all algorithms in this assignment.

## 3. Tune the Hyper-Parameters and Classify

For the classification, a Neural Network was used for all experiments in this assignment. The sklearn hyper-parameters: `'activation'`, `'learning_rate'` and `'solver'` were optimized by applying a ten-fold cross validation using sklearn's GridSearchCV. The number of hidden layers for the Neural Network were not included in the GridSearch, that is, the number of hidden layers was left at the default setting (`sklearn.neural_network.MLPClassifier: hidden_layer_sizes: tuple, length = default (100,)`). To account for algorithmic processing efficiency, the run-time of both the dimensionality reduction and classification were captured. It should be noted that the computer running the experiments was not exclusively restricted to running experiments, e.g., there were other operations and multi-tasked operations occurring during the experimentation. This could have impacted the time, or at least leas to increases in the deviation of observed run-time values. Also the optimal hyper-parameters for classification change as per the number of dimensions. This also had an impact on the variance of observed classification run-time.

To avoid bias in generalizing to the test data when classification (Supervised Learning, Neural Network) was performed, the problem was partitioned to an 80% test, 20% training for both clustering and dimensionality reduction. The reduced data was then partitioned into 50% test/training for classification.

## 4. DISCUSS THE RESULTS

To provide a baseline for this project, the two problems were classified with a neural network, without clustering and without dimensionality reduction. Classification was performed using all features, and using only two features (chosen at random). This results of this ad-hoc approach of feature reduction will be compared against both clustering and dimensionality reduction. The results in table show a large decrease in test accuracy, while the learning curve for IRIS show a need for more training data using ad-hoc feature reduction.

**Table 1**: Summery of Accuracy for classification Neural Network.

| **Table 1**<br>**Summary of Accuracy** | **Nrl Net** | | **Nrl Net** | |
|---|---|---|---|---|
| | **IRIS** | **SPAM** | **IRIS** | **SPAM** |
| **# features** | **All** | **all** | **(0,3)** | **(0,14)** |
| **CLF Run Time (s)* | ***28*** | ***205*** | ***25*** | ***155*** |
| *Accuracy Train %* | ***97*** | ***92*** | ***96*** | ***67*** |
| *# Misclassified* | ***2*** | ***205*** | ***12*** | ***748*** |
| *# Correct* | ***73*** | ***2095*** | ***63*** | ***1552*** |
| *Accuracy Test %* | ***97*** | ***91*** | ***84*** | ***67*** |

* Time for classification includes 10-fold cross validation.

**Figure 1a**: Plots of data classification using only 2 features. The shaded regions denote decision boundaries. The classification of the test data is depicted in the plot by circles.

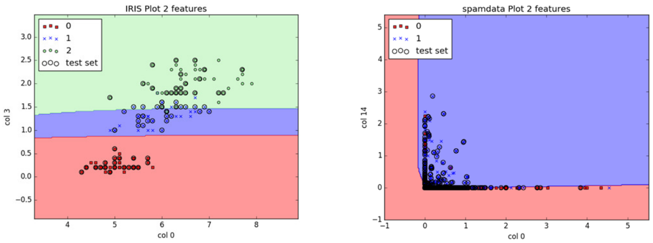
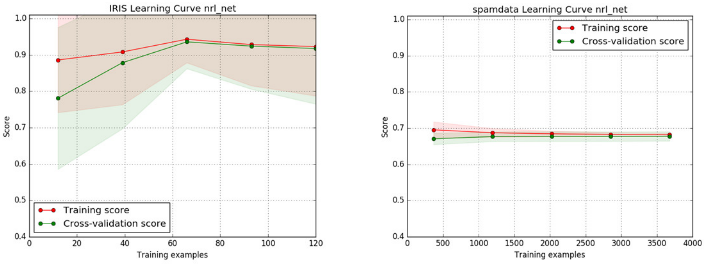


**Figure 1b**: Learning curves for classifying the problems with a Neural Network that uses only two features.



### 4.1. Clustering: k-means, Expectation Maximization (EM)

The first clustering algorithm for discussion is k-means. K-means identifies clusters using an iterative process of assigning and optimizing cluster centers within the data. The optimization step minimizing the sum-of-squares (Euclidian) distances between points within a cluster and the cluster center. When the number of clusters is fixed to k, k-means clustering is an optimization problem: find the k cluster centers and assign the points to the nearest cluster center, such that the squared distances from the cluster are minimized. The optimization problem is known to be NP-hard, and thus the common approach is to search only for approximate solutions. Drawbacks of k-means: k must be specified in advance, bias toward clusters of approximately similar size. Like hill climbing, k-means is susceptible to local minimum, thus, is dependent on where the initial centers for the clusters are placed.

Expectation Maximization (EM), which was implemented using a Gaussian Mixture Model (GMM) for this project, is also an iterative algorithm, with Expectation and Maximization steps. Like k-means, EM exposes clusters in the data. However, where k-means tends to find clusters of comparable spatial profile, EM allows clusters to have different shapes.

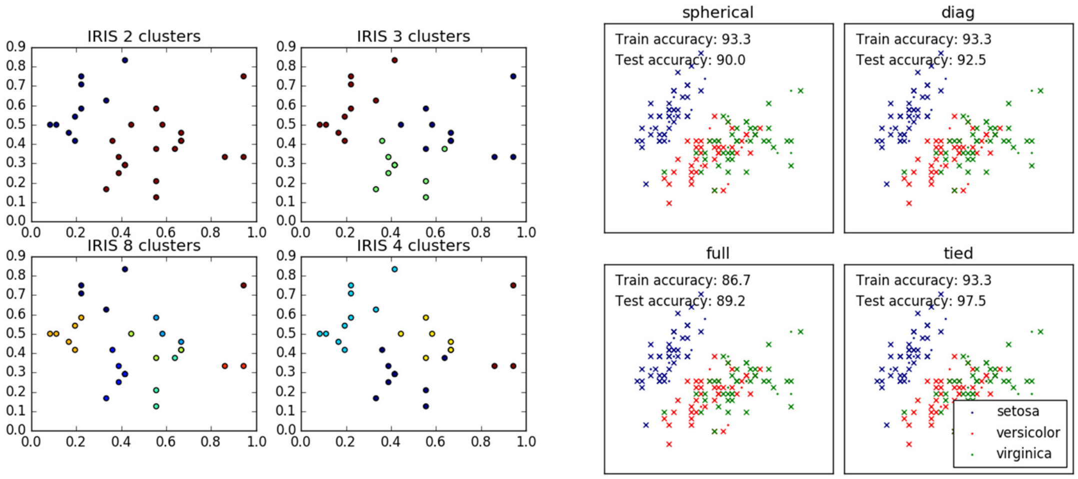
**Table 2**: Summary of results from clustering and expectation maximization. For this experiment, the objective was to observe the performance of k-means across a range: 2 ≤ k ≤ # features. The upper bound for k (# features) is not a good heuristic, it was selected to demonstrate the significance of choosing a 'good' value for k.

| **Table 2**<br>**Clustering** | **Num Clstr 2** | **Num Clstr 3** | **Num Clstr 8** | **Num Clstr 4**** | **Num Clstr 58**** | **EM** | **Notes** |
|---|---|---|---|---|---|---|---|
| **IRIS:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.025*<br>*70*<br>*66* | *0.026*<br>*30*<br>*29* | *0.042*<br>*47*<br>*43* | *0.033*<br>*20*<br>*23* | | *0.011*<br>*93*<br>*98* | **k-means**: K=2 performs the best, although the performance is not linear as k increases.<br>**EM cov='tied'**: Best performance. |
| **SPAM:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.126*<br>*62*<br>*59* | *0.122*<br>*50*<br>*49* | *0.184*<br>*9*<br>*11* | | *0.555*<br>*0*<br>*0.33* | *0.019*<br>*63*<br>*60* | **k-means**: K=2 performs the best, the performance decreases linearly as k increases.<br>**EM cov='spherical'**: Best performance. |

* Accuracy estimated using labels from problem.

** There values were obtained from the number of features for the problem.

**Figure 2**: k-means plots (left) of IRIS for 2 ≤ k ≤ # features. EM plots for covariance: [`'spherical'`, `'diag'`,`'full'`, `'tied'`]. Note the clustering appears more pronounced for EM. K-means, k=2 appears to generate the best clusters.



### 4.2. Dimensionality Reduction

There were some challenges for obtaining interesting results when applying dimensionality reduction to the problems. Values for quantifying the dimensionality reduction methods were captured: Eigenvalues, Kurtosis and Reconstruction Error. The results for both problems were similar: PCA Eigenvalues for the first two dimensions were dominant. With increasing dimensionality: ICA Kurtotic distributions plateau at a strongly leptokurtic (tall and narrow curve) value; NMF and RP reconstruction error approach zero.

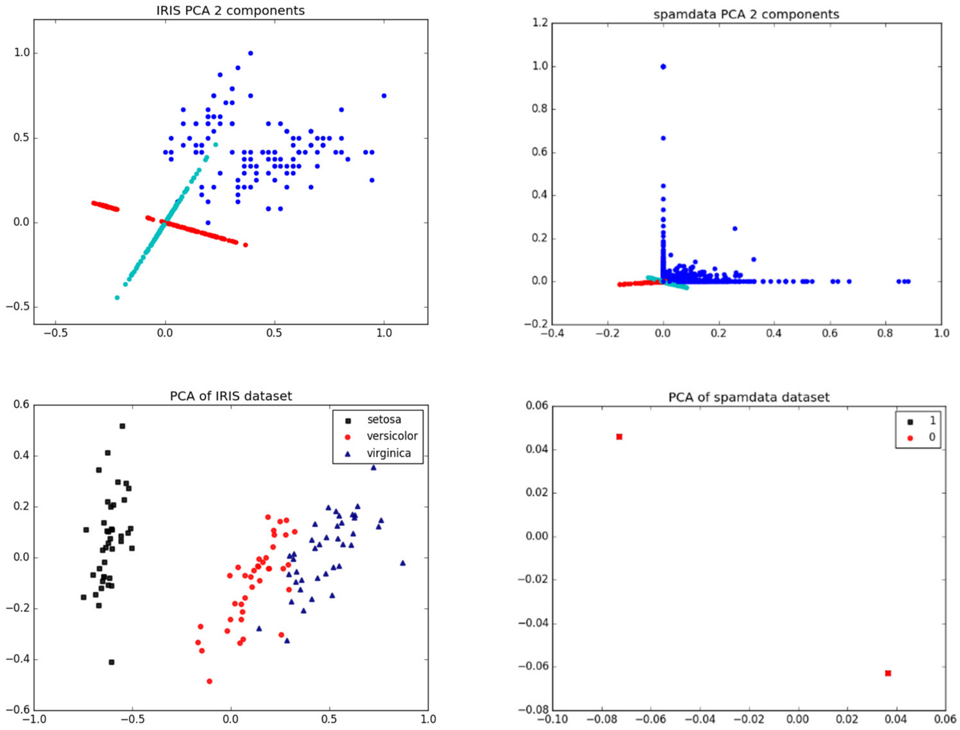


**Figure 3**: The first two principal component vectors are plotted against the original data points (top). Note the principal components are axis parallel to the SPAM data.

The data is transformed using the first two principal components and plotted (bottom). Note the well define clusters for both problems. Interestingly, the data collapsed for the PCA transformation for SPAM.

### 4.3. Clustering with Dimensionality Reduction

This experiment is a repeat of the experiment in section 4.1, but with dimensionality reduction applied prior to clustering. The data was reduced using the first two, then three components.

**Table 3a**: Summary of results from clustering and expectation maximization on dimensionally reduced problem using the first two components from PCA.

| **Table 3a**<br>**Clustering/PCA with (n=2)** | **Num Clstr 2** | **Num Clstr 3** | **Num Clstr 8** | **Num Clstr 4**** | **Num Clstr 58**** | **EM** | **Notes: The number of components DR keeps=2.** |
|---|---|---|---|---|---|---|---|
| **IRIS:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.026*<br>*0*<br>*0* | *0.026*<br>*4.2*<br>*3.1* | *0.046*<br>*0*<br>*0* | *0.03*<br>*38*<br>*49* | | *0.009*<br>*96*<br>*96* | **k-means+DR**: Performs worse than k-means.<br>**EM+DR**: Performs on par with EM. |
| **SPAM:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.033*<br>*56*<br>*59* | *0.059*<br>*49*<br>*50* | *0.116*<br>*22*<br>*22* | | *0.429*<br>*2*<br>*3* | *0.014*<br>*54*<br>*57* | **k-means+DR**: Performs on par with k-means.<br>**EM+DR**: Performs on par with EM. |

* Accuracy estimated using labels from problem.

** There values were obtained from the number of features for the problem

**Table 3b**: Summary of results from clustering and expectation maximization on a dimensionally reduced problem using the first three components from PCA.

| **Table 3b**<br>**Clustering/PCA with (n=3)** | **Num Clstr 2** | **Num Clstr 3** | **Num Clstr 8** | **Num Clstr 4**** | **Num Clstr 58**** | **EM** | **Notes: The number of components DR keeps=3.** |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| **IRIS:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.03*<br>*75*<br>*64* | *0.03*<br>*0*<br>*7.3* | *0.04*<br>*8*<br>*7* | *0.03*<br>*17*<br>*17* | | *0.008*<br>*96*<br>*99* | **k-means+DR**: Performs better than k-means when #clusters=2.<br>**EM+DR**: Performs better than EM. |
| **SPAM:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.06*<br>*56*<br>*59* | *0.07*<br>*80*<br>*78* | *0.115*<br>*21*<br>*20* | | *0.5*<br>*2*<br>*1* | *0.02*<br>*55*<br>*56* | **k-means+DR**: Performs better than k-means when #clusters=3.<br>**EM+DR**: Performs worse than EM. |

* Accuracy estimated using labels from problem.

** There values were obtained from the number of features for the problem

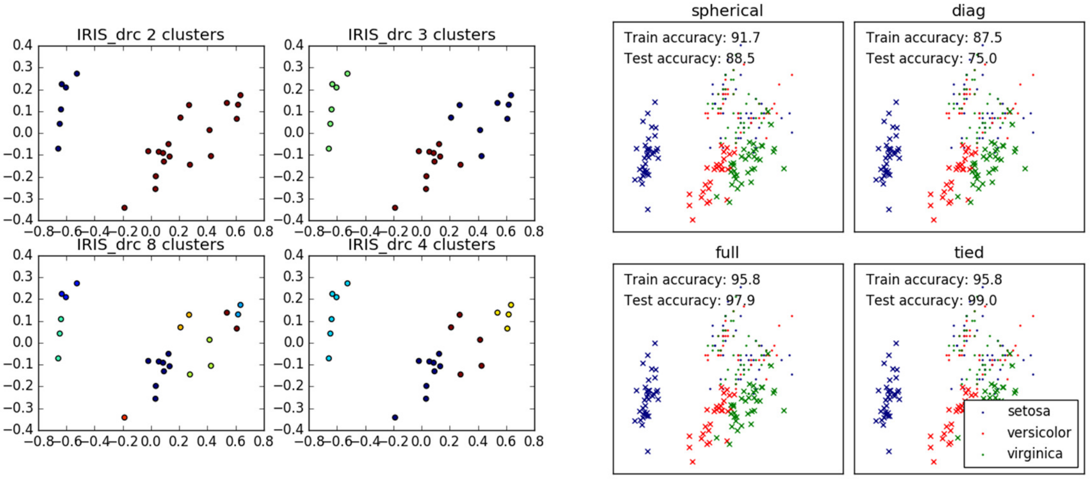


**Figure 3b**: k-means plots (left) of IRIS for 2 ≤ k ≤ # features. EM plots for covariance: [‘spherical', ‘diag’,‘full’, ‘tied’]. PCA was used prior to clustering to reduce the data using the first three components. Note the clustering appears more pronounced for both k-means and EM when DR is applied first.

**Table 4a**: Summary of results from clustering and expectation maximization on a dimensionally reduced problem using the first two components from ICA.

| **Table 4a**<br>**Clustering/ICA with (n=2)** | **Num Clstr 2** | **Num Clstr 3** | **Num Clstr 8** | **Num Clstr 4**** | **Num Clstr 58**** | **EM** | **Notes: The number of components DR keeps=2.** |
|---|---|---|---|---|---|---|---|
| **IRIS:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.014*<br>*0*<br>*1* | *0.017*<br>*88*<br>*76* | *0.042*<br>*17*<br>*32* | *0.03*<br>*21*<br>*19* | | *0.009*<br>*96*<br>*96* | **k-means+DR**: Performs better than k-means with #clusters=3.<br>**EM+DR**: Performs on par with EM. |
| **SPAM:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.059*<br>*81*<br>*79* | *0.06*<br>*78*<br>*78* | *0.12*<br>*34*<br>*34* | | *0.61*<br>*7*<br>*5* | *0.01*<br>*44*<br>*43* | **k-means+DR**: Performs better than k-means with #clusters=[2, 3].<br>**EM+DR**: Performs worse than EM. |

* Accuracy estimated using labels from problem.

** There values were obtained from the number of features for the problem

**Table 4b**: Summary of results from clustering and expectation maximization on a dimensionally reduced problem using the first three components from ICA.

| **Table 4b** **Clustering/ICA with (n=3)** | **Num Clstr 2** | **Num Clstr 3** | **Num Clstr 8** | **Num Clstr 4**** | **Num Clstr 58**** | **EM** | **Notes: The number of components DR keeps=3.** |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| **IRIS:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.03*<br>*4*<br>*4* | *0.03*<br>*88*<br>*68* | *0.04*<br>*8*<br>*7* | *0.03*<br>*8*<br>*13* | | *0.008*<br>*96*<br>*99* | **k-means+DR**: Performs on par with k-means.<br>**EM+DR**: Performs better than EM. |
| **SPAM:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.03*<br>*19*<br>*21* | *0.04*<br>*78*<br>*78* | *0.11*<br>*11*<br>*10* | | *0.314*<br>*1*<br>*1* | *0.014*<br>*55*<br>*58* | **k-means+DR**: Performs better than k-means when #clusters=3.<br>**EM+DR**: Performs worse than EM. |

* Accuracy estimated using labels from problem.

** There values were obtained from the number of features for the problem

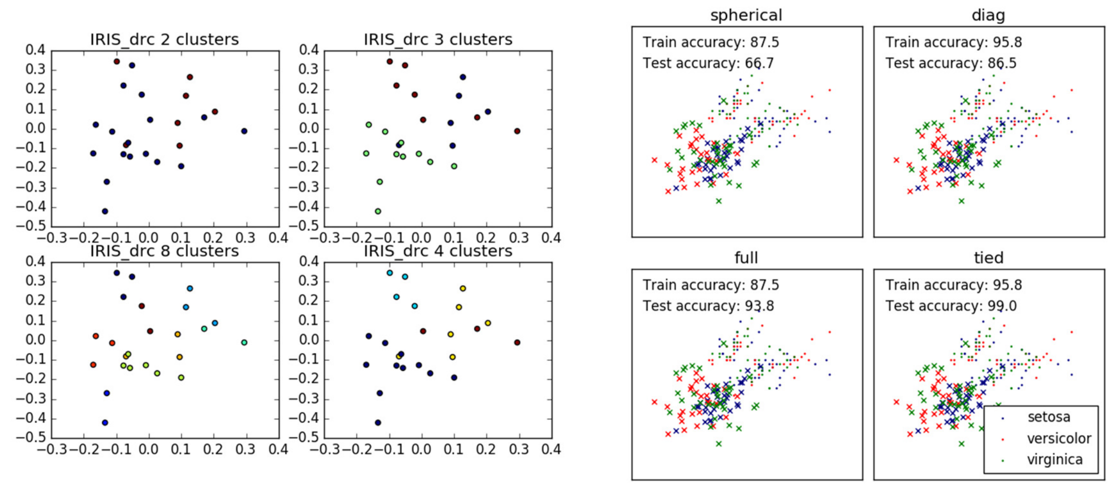


**Figure 4b**: k-means plots (left) of IRIS for 2 ≤ k ≤ # features. EM plots for covariance: [‘spherical', ‘diag’,‘full’, ‘tied’]. ICA was used prior to clustering to reduce the data using the first three components. Note the clustering appears about the same for both k-means and EM when DR is applied first.

**Table 5a**: Summary of results from clustering and expectation maximization on a dimensionally reduced problem using the first two components from Randomized Projections (RP).

| **Table 5a** **Clustering/RP with (n=2)** | **Num Clstr 2** | **Num Clstr 3** | **Num Clstr 8** | **Num Clstr 4**** | **Num Clstr 58**** | **EM** | **Notes: The number of components DR keeps=2.** |
|---|---|---|---|---|---|---|---|
| **IRIS:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.02*<br>*42*<br>*24* | *0.029*<br>*46*<br>*42* | *0.041*<br>*8*<br>*2* | *0.023*<br>*17*<br>*14* | | *0.009*<br>*96*<br>*96* | **k-means+DR**: Performs worse than k-means.<br>**EM+DR**: Performs on par with EM. |
| **SPAM:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.012*<br>*57*<br>*58* | *0.072*<br>*44*<br>*45* | *0.19*<br>*28*<br>*27* | | *0.41*<br>*1*<br>*2* | *0.009*<br>*54*<br>*56* | **k-means+DR**: Performs on par with k-means with #clusters=2.<br>**EM+DR**: Performs worse than EM. |

* Accuracy estimated using labels from problem.

** There values were obtained from the number of features for the problem

**Table 5b**: Summary of results from clustering and expectation maximization on a dimensionally reduced problem using the first three components from Randomized Projections (RP).

| **Table 5b**<br>**Clustering/RP with (n=3)** | **Num Clstr 2** | **Num Clstr 3** | **Num Clstr 8** | **Num Clstr 4**** | **Num Clstr 58**** | **EM** | **Notes: The number of components DR keeps=3.** |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| **IRIS:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.02*<br>*75*<br>*63* | *0.03*<br>*25*<br>*34* | *0.04*<br>*0*<br>*0* | *0.03*<br>*17*<br>*23* | | *0.006*<br>*96*<br>*96* | `k-means+DR`: Performs on par with k-means when #clusters=2.<br>`EM+DR`: Performs on par with EM. |
| **SPAM:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.06*<br>*43*<br>*46* | *0.1*<br>*57*<br>*55* | *0.2*<br>*26*<br>*26* | | *0.5*<br>*4*<br>*3* | *0.01*<br>*53*<br>*55* | `k-means+DR`: Performs on par with k-means when #clusters=3.<br>`EM+DR`: Performs worse than EM. |

*   Accuracy estimated using labels from problem.

** There values were obtained from the number of features for the problem

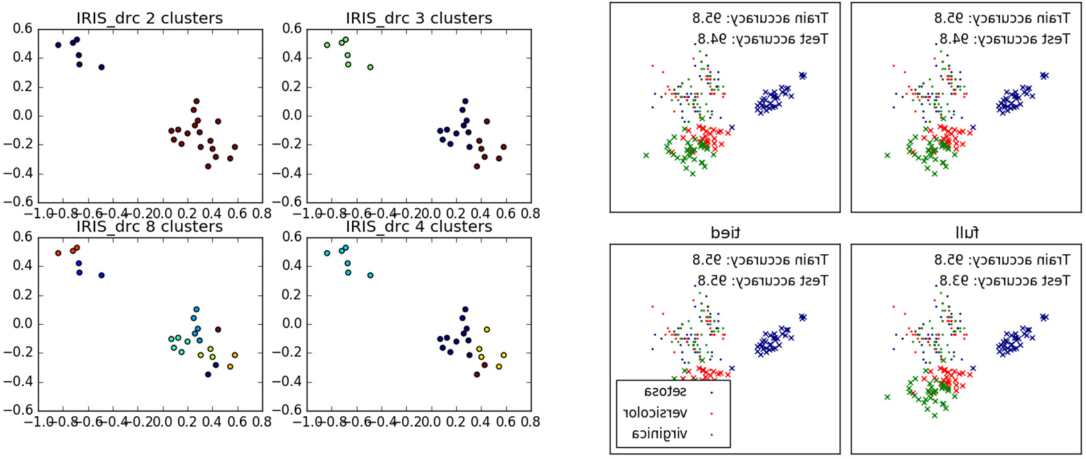


**Figure 5b**: k-means plots (left) of IRIS for 2 ≤ k ≤ # features. EM plots for covariance: [`'spherical'`, `'diag'`,`'full'`, `'tied'`]. RP was used prior to clustering to reduce the data using the first three components. Note the clustering appears about the same for EM, but more pronounced for k-means when DR is applied first.

**Table 6a**: Summary of results from clustering and expectation maximization on a dimensionally reduced problem using the first two components from Non-Negative Matrix Factorization (NMF).

| **Table 6a**<br>**Clustering/NMF with (n=2)** | **Num Clstr 2** | **Num Clstr 3** | **Num Clstr 8** | **Num Clstr 4**** | **Num Clstr 58**** | **EM** | **Notes: The number of components DR keeps=2.** |
|---|---|---|---|---|---|---|---|
| **IRIS:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.02*<br>*0*<br>*0* | *0.026*<br>*25*<br>*29* | *0.04*<br>*0*<br>*0* | *0.026*<br>*0*<br>*7* | | *0.006*<br>*96*<br>*96* | `k-means+DR`: Performs worse than k-means.<br>`EM+DR`: Performs on par with EM. |
| **SPAM:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.04*<br>*54*<br>*58* | *0.05*<br>*76*<br>*76* | *0.13*<br>*5*<br>*6* | | *0.46*<br>*3*<br>*3* | *0.02*<br>*53*<br>*57* | `k-means+DR`: Performs better than k-means with #clusters=3.<br>`EM+DR`: Performs worse than EM. |

*   Accuracy estimated using labels from problem.

** There values were obtained from the number of features for the problem

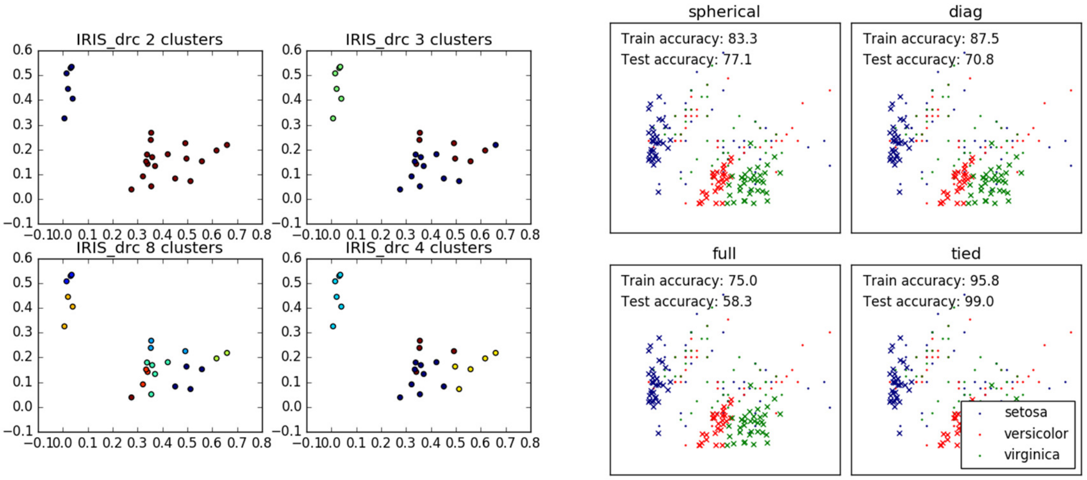
**Table 6b**: Summary of results from clustering and expectation maximization on a dimensionally reduced problem using the first three components from Non-Negative Matrix Factorization (NMF).

| **Table 6b**<br>**Clustering/NMF with (n=3)** | **Num Clstr 2** | **Num Clstr 3** | **Num Clstr 8** | **Num Clstr 4**** | **Num Clstr 58**** | **EM** | **Notes: The number of components DR keeps=3.** |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| **IRIS:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.02*<br>*75*<br>*64* | *0.03*<br>*17*<br>*18* | *0.04*<br>*4*<br>*6* | *0.03*<br>*21*<br>*22* | | *0.008*<br>*96*<br>*99* | **k-means+DR:** Performs on par with k-means when #clusters=2.<br>**EM+DR:** Performs better than EM. |
| **SPAM:**<br>*Run Time (s)*<br>**Accuracy Train %*<br>**Accuracy Test %* | *0.054*<br>*44*<br>*41* | *0.05*<br>*54*<br>*57* | *0.14*<br>*18*<br>*19* | | *0.4*<br>*1*<br>*1* | *0.02*<br>*51*<br>*53* | **k-means+DR:** Performs on par with k-means when #clusters=3.<br>**EM+DR:** Performs worse than EM. |

* Accuracy estimated using labels from problem.

** There values were obtained from the number of features for the problem

**Figure 6b**: k-means plots (left) of IRIS for 2 ≤ k ≤ # features. EM plots for covariance: [`'spherical'`, `'diag'`, `'full'`, `'tied'`]. NMF was used prior to clustering to reduce the data using the first three components. Note the clustering appears about the same for EM, but more pronounced for k-means when DR is applied first.

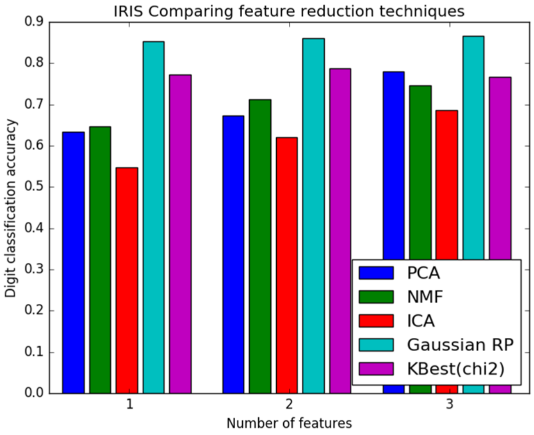


### 4.4. Classification on a Dimensionality Reduced Problem

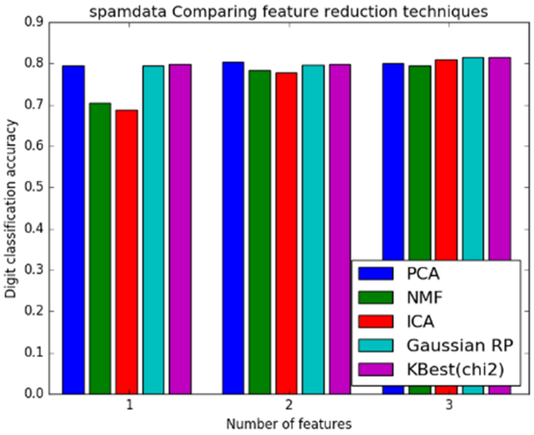
A pipeline in sklearn was utilized to evaluate, and apply the Dimensionality Reduction (DR) algorithms to a problem prior to classification. The pipeline combined DR: PCA, ICA, NMF, Gaussian RP and K-Best (chi2), then performed classification using a neural network with 10-fold cross validation. Recall the number of features (dimensions) in the problems: IRIS = 4, SPAM= 58. The range of dimensionality reduction for IRIS was 1 ≤ DR ≤ 3, and 1 ≤ DR ≤ 5 for SPAM. To baseline DR, a chi-square test was implemented to 'weed out' the features that are the most likely to be independent of class and therefore irrelevant for classification.

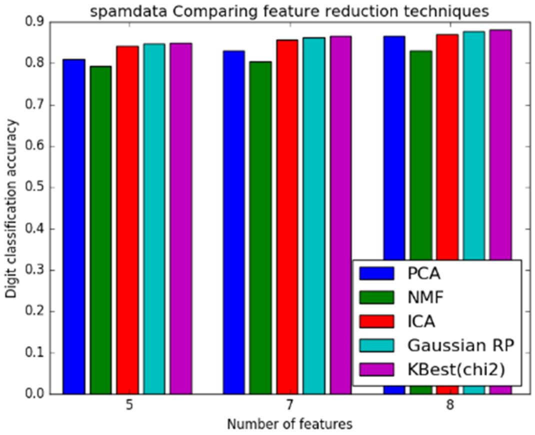
**Figure 1**: A comparison of feature (dimensionality) reduction methods. The graphical analysis presents the evaluation of DR algorithms across a range of feature reduction. Observe the accuracy of the classification as number of dimensions for the problem is increased. For IRIS, notice that the accuracy for classification improves for: PCA, NMF and ICA as more dimensions are passed to the neural network. Gaussian RP is the most accurate and consistent across all dimensions, and is superior to the baseline chi-square test. With three dimensions, PCA reaches the chi-square upper-bound as well.

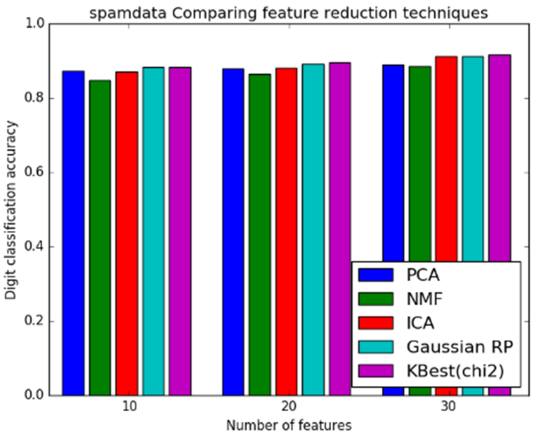
Similar, but not as dramatic results are seen in the SPAM problem. NMF and ICA show strong improvement in classification accuracy when the dimensionality is increased from one to two. All DR methods exhibit improved accuracy as the dimensionality is increased from two to ten. After ten dimensions, a very slow plateau of ~90% accuracy is obtained.

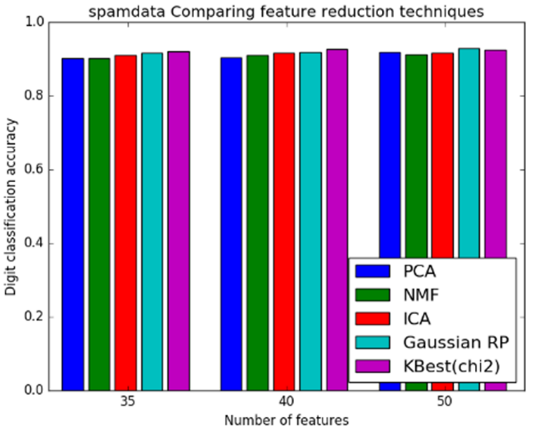


Compare the results of classification using dimensionality reduction to keep two dimensions with our ad-hoc feature reduction using two features for classification (table 1, figure 1). With the ad-hoc approach, the accuracy was: IRIS=84%, SPAM=67%. Although IRIS fared worse for all DR methods except for RP, SPAM performed better across all DR approaches. A greater than 90% classification was achieved with DR on the SPAM problem at ~10 dimensions, matching the results of classification without DR. Applying DR to IRIS was not as successful, as the 97% accuracy without DR was not achieved.









### 4.5. Classification Using a Neural Network on Clustered Data

The section of the project will use a neural network to classify data that has been clustered using: k-means and Expectation Maximization (EM). The approach was to treat the clustered data as features to the neural network. For k-means, since we have labeled data, a simple heuristic for defining the cluster size was to assign the number of classes to the cluster size. The EM feature in sklearn did not have a transformation method, so I used the `n_samples` attribute set to the size of the test set to generate a transformed data set for classification: `X = estimator.sample(n_samples=len(X_test))[0]`.

**Table 7**: Summary of results from classification using clustering and expectation for data reduction. The reduced data set was passed to a neural network for classification.

| **Table 7**<br>**CLF/Clustering** | **Num Clstr 2**** | **Num Clstr 3**** | **EM** | **Notes** |
|---|---|---|---|---|
| | | | | |
| **IRIS:**<br>**CLF Run Time (s)*<br>*Accuracy Train %*<br>*# Misclassified*<br>*# Correct*<br>*Accuracy Test %* | *57*<br>*92*<br>*6*<br>*54*<br>*90* | | *46*<br>*40*<br>*39*<br>*21*<br>*35* | **CLF+k-means:** Performs better than CLF+DR.<br>**CLF+EM:** Performs worse than CLF+DR. |
| **SPAM:**<br>**CLF Run Time (s)*<br>*Accuracy Train %*<br>*# Misclassified*<br>*# Correct*<br>*Accuracy Test %* | | *125*<br>*75*<br>*457*<br>*1383*<br>*75* | *114*<br>*55*<br>*793*<br>*1047*<br>*57* | **CLF+k-means:** Performs better than CLF+DR.<br>**CLF+EM:** Performs worse than CLF+DR. |

*  Time for classification includes 10-fold cross validation.
** There values were obtained from the number of classes for the problem

## 5. CONCLUSION

When clustering is performed over a dimensionally reduced problem, choosing the cluster size, dimensionality reducing algorithm and scope of dimensionality reduction relative to the problem is critical. Some clustering algorithms, EM for instance, may be more predictable over a range of dimensionality reduction. Others, like k-means, are less forgiving, giving wildly disparate results for initial cluster sizes and across dimensionality reduction. Classification using dimensionality reduction to reduce the feature space was shown to be more accurate than an ad-hoc approach for the SPAM problem, and performed better for randomized projections for the IRIS problem. Surprisingly, classification using k-means to reduce the feature space performed even better the dimensionality reduction. The methods used to measure dimensionality reduction: Eigenvalues, Kurtosis and Reconstruction Error showed that the optimal value for the number of dimensions to keep was approximate to the number of classes in the problem. Of course, with unsupervised learning, we would not have class information, but this could be estimated from the measurements, or from clustering.

[1] http://scikit-learn.org/stable/
[2] https://archive.ics.uci.edu/ml/datasets/Spambase