Unsupervised Learning and Dimensionality Reduction

1. Requirement

Python, scikit-learn, scipy.

2. Datasets

Iris dataset

The Iris dataset is a multivariate dataset with 150 datapoints and 4 features, which can be used for classification or clustering. The 4 features are sepal length, sepal width, petal length and petal width, and the target contain 3 classes: setosa, vesicolor, and virginicia.

Breast Cancer dataset

The breast cancer dataset is another multivariate dataset with 569 datapoints and 30 features. All 30 features are numeric, and their definition can be found here: https://scikit-learn.org/stable/datasets/index.html#breast-cancer-dataset. The target contains 2 classes: malignant, benign

Given the number of different test combinations for this assignment, datasets must be 1) simple enough to allow modeling to finish within a reasonable timespan, and 2) multivariate to allow dimensionality reduction. Both Iris and Breast Cancer datasets satisfy the requirement. Another value of using these two datasets is the ease of visualization. Since Iris data only has 4 features, after dimensionality reduction it can be shown on a 2D plot, as a visual check for the quality of dimensionality reduction.

3. Clustering

Two algorithms are tested in this section: K-Means and Expectation Maximization (EM). Since K-Means involves distance calculation, features for both data points are transformed to be mean = 0 and standard deviation = 1.

5-fold cross-validation is performed for all calculations. Figures 3.1 and 3.3 are the average scores of cross-validation. Other plots, such as scatter plots, are generated by the model from cross-validation to the entire dataset.

3.1 K-Means

The key parameter to choose in K-Means is the number of clusters (k). The easiest and most straightforward way to test if data points are well clustered is by calculating the distance of each points to its cluster centers. Theoretically, when every single data point is its own cluster, the distance reduces to 0. However, this is impractical and deviates from the goal. The goal is to find a probable cluster number so that 1) each cluster contains a substantial amount of points, 2) data points in each cluster are close to its own cluster center while 3) are distant away from its neighboring clusters.

In K-Means, inertia is a metric that measures the summation of squared distance of data points to its cluster center. For every number of clusters, inertia can be computed. Figure below shows inertia variation with number of clusters for both datasets.

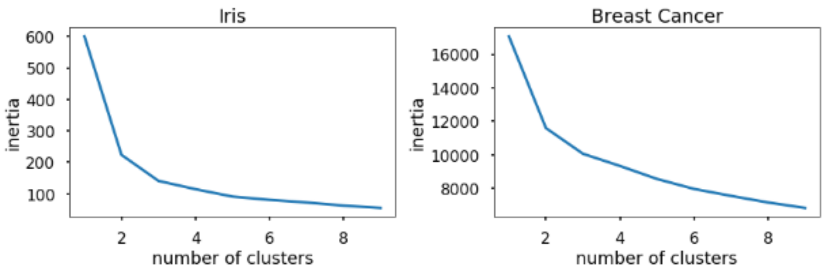


Figure 3.1 inertia variation with number of clusters

In both cases, inertia exhibits an 'elbow' shape as number of clusters increases. A good choice of number of clusters is when 'elbow' happens. However, this rule-of-thumb is quite subjective. For instance, the target of Iris contains 3 classes, but 'elbow' occurs at 2 or 3 (after trying some other hyperparameter combinations, for instance, 'k-means++' vs. 'random', different max iterations, etc., the observation does not change). Similarly, 'elbow' happens when number of clusters is 2 or 3 for Breast Cancer, even though it is less apparent than Iris. This aligns with the 2 classes in the dataset target.

I then choose the number of clusters for each dataset to be the same with the number of target classes. The confusion matrices are presented below.

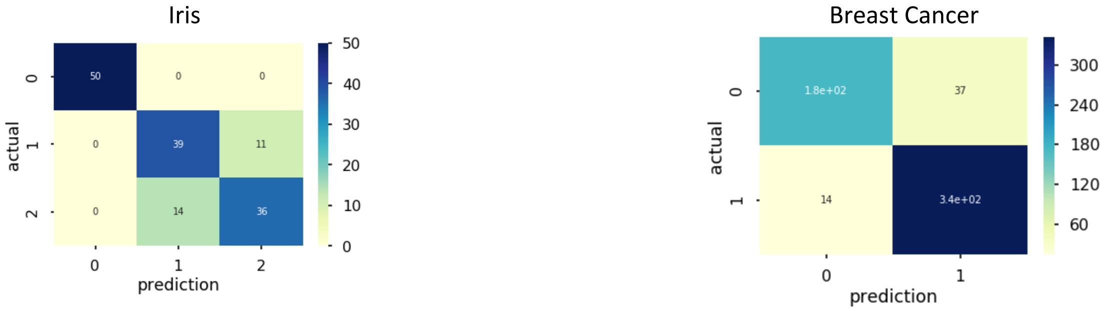


Figure 3.2 confusion matrix for actual labels vs. K-Means prediction

For Breast Cancer dataset, false positives and negatives are quite low compared to the dataset size, and accuracy score is 0.91. For Iris dataset, the accuracy score is 0.81. This is due to relatively high false positive and negative counts for class 1 and 2, even though there is no misclassification for class 1. It is difficult to visualize clusters with 4 features (Iris), but after dimensionality reduction is performed in the following section, it becomes clear why the misclassification would happen between class 2 and 3.

3.2 Expectation Maximization

In Scikit-Learn, EM is applied in Gaussian Mixture model for clustering. Unlike K-Means, AIC and BIC curves are commonly used to help determine the best number of clusters. Both AIC and BIC are a function the max likelihood function, which measures how good a model can present data points. The difference between AIC and BIC can be found in the links: https://en.wikipedia.org/wiki/Bayesian_information_criterion, https://en.wikipedia.org/wiki/Akaike_information_criterion.

For Gaussian Mixture model, one important parameter, other than the number of clusters, that affects AIC and BIC score is the covariance type. Covariance type determines the shape of how models should grow. In Scikit-Learn, four types are supported: full, tied, diag, spherical. For the definition of these types, please see https://scikit-learn.org/stable/auto_examples/mixture/plot_gmm_covariances.html#sphx-glr-auto-examples-mixture-plot-gmm-covariances-py. All 4 covariance types are evaluated in this report.

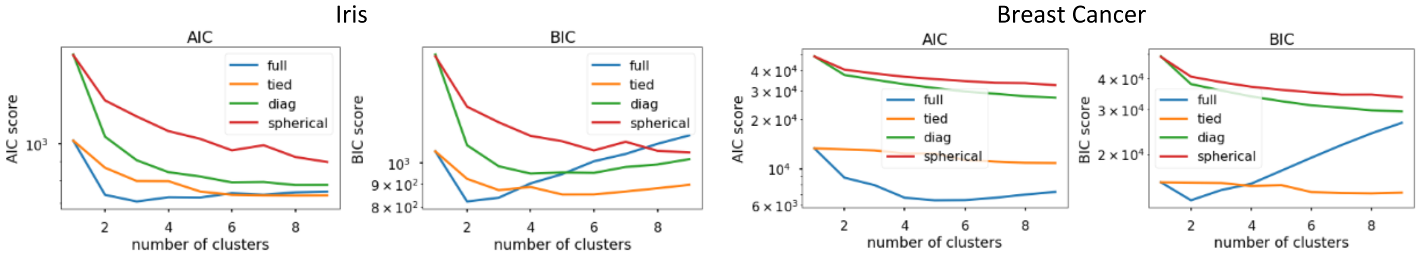


Figure 3.3 AIC and BIC variation with number of clusters, for both datasets

Both AIC and BIC are linearly, negatively correlated with the max likelihood function. Therefore, when the model better expresses sample data points (higher likelihood function), AIC and BIC should be lower. The rule-of-thumb to determine the best number of clusters is to find the lowest point of the 'U' shape curve.

The shape of the curve is heavily affected by the choice of covariance type. For Iris, only type 'full' and 'tied' exhibit 'U' shape, and only type 'full' has the lowest point at the reasonable position (2 or 3 clusters), while type 'tied' has the lowest point at much higher clusters. Note that AIC and BIC scores for 2 or 3 clusters are not quite different, suggesting a vague boundary between two classes. This observation agrees with Figure 3.2, where high false positives and negatives exists between class 2 and 3.

Only type 'full' exhibits 'U' shape for Breast Cancer. However, the lowest point for AIC curve happens at 5 clusters, which does not agree with the number of classes (2). In this case, only BIC curve finds the correct number of clusters. Confusion matrices for both datasets with the correct metrics (AIC for Iris, and BIC for Breast Cancer) and covariance type ('full' for both) are presented below.

Iris

Breast Cancer

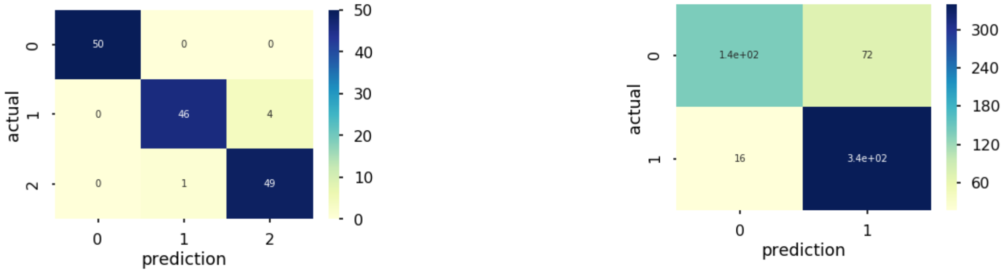


Figure 3.4 confusion matrix for actual labels vs. Gaussian Mixture prediction

Gaussian Mixture with EM has a higher accuracy (0.97) than K-Means (0.81) for Iris. False positives and negatives for class 1 and 2, which are quite significant for K-Means, now drop to almost 0. However, the accuracy for Breast Cancer is 0.85, lower than K-Means (0.91). False positives almost double using Gaussian Mixture with EM, which becomes the primary drive to lower the accuracy score.

The performance discrepancy on two datasets shows there is no conclusion on which algorithm is better for clustering. Algorithm selection is data sensitive and it is a trial-and-error process.

4. Dimensionality Reduction

PCA, ICA, Random Projection and Random Forest are applied for dimensionality reduction.

4.1 PCA

The key parameter to determine the number of principle components is explained variances and also eigenvalues. The relative value of a principle component is proportional to its importance, and low value principle component can be discarded. Figure 4.1 shows the explained variance and eigenvalues for two datasets.

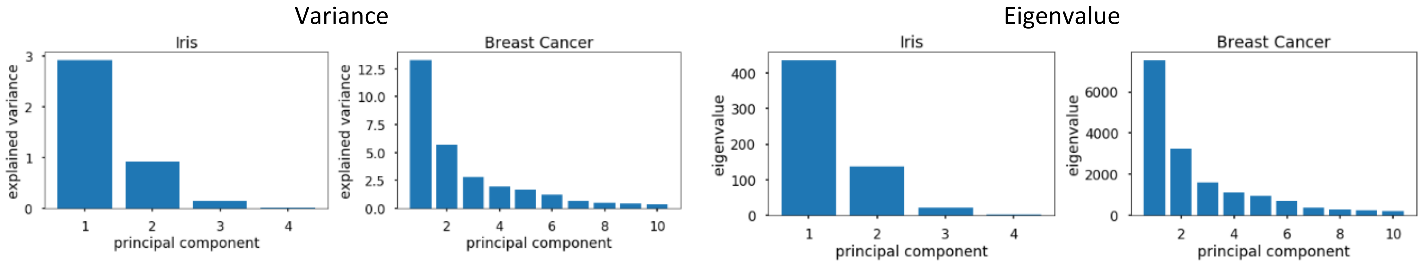


Figure 4.1 explained variance for principle components

Explained variance and eigenvalue are different measurement of matrix transformation, but the relative distribution among different principal component are the same. For Iris, the first two principle components have explained variance much larger than the other two, suggesting the original dimension of Iris (4) can be reduced to 2. For Breast Cancer, however, it is unclear which principle components are the dominating ones. Other than the first principle component, there is no significant difference in the variances for components 2 to 6. It is arguably that the first 6 components can be used after the dimensionality reduction.

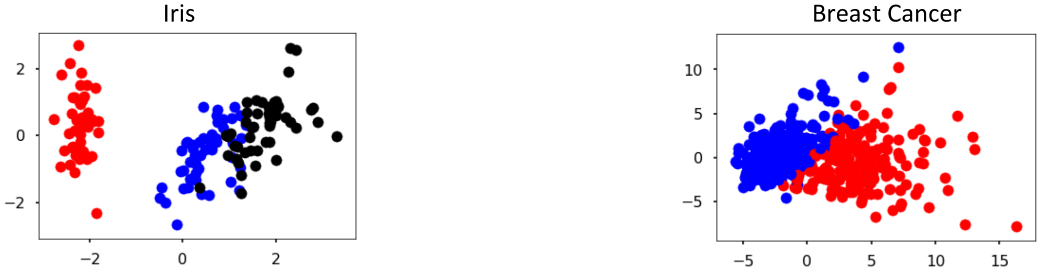


Figure 4.2 scatter plot with target class after dimensionality reduction using PCA

For both datasets, the original dimensions are reduced to two for visualization. This is inappropriate for Breast Cancer since Figure 4.1 shows 2 principle components are not enough to describe the original dataset. Figure 4.2 for Breast Cancer is just for visualization and demonstration. For Iris, 2 principle components successfully separate the dataset into two clusters: the red cluster represents class 0, and blue and black clusters represent classes 1 and 2. Recall that K-Means can identify perfectly class 0, but has some misclassification for class 1 and 2. Figure 4.2 gives the visual evidence: class 1 is separable from the rest of the data points, while there is no clear boundary between classes 1 and 2. For Breast Cancer, 2 principle components are insufficient to cluster the data points; however, the 2 classes are divided very nicely without much overlap.

4.2 ICA

The idea of ICA is to separate multivariate dataset to independent components. One key assumption for ICA is that the components after projection are non-Gaussian. To measure non-Gaussianity, Kurtosis is often used. In this section, kurtosis for each component is calculated and compared for each dataset.

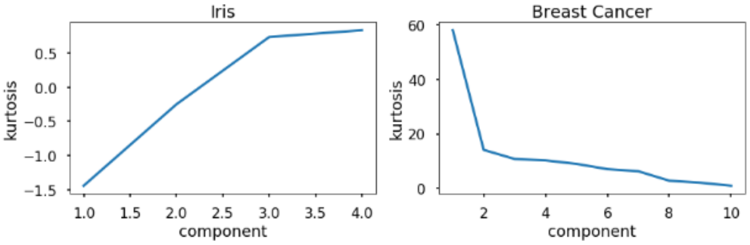


Figure 4.3 kurtosis for different ICA components

When kurtosis is 0, it means the distribution is gaussian. A negative kurtosis means the distribution is 'flatter' than a normal distribution and a positive kurtosis means the distribution is 'sharper' than a normal distribution. The goal is to pick components that has a kurtosis away from 0.

For Iris, the corresponding kurtosis for 4 components are: -1.44, -0.25, 0.73, 0.83. components 1, 3, and 4 are relatively away from 0 compared to 2nd component. Therefore, components 1, 4 or 1, 3, 4 can be selected as the final independent components. Note that the component 2 cannot be selected not only because its kurtosis value is close to 0, but also because adding component 2 to the final selection would make the dimension of the projection the same with original data, violating the goal for dimensionality reduction.

For Breast Cancer, all kurtoses are greater than 0. The first component has a much higher kurtosis than others, and kurtoses for components 2 to 7 has no significant difference. This observation agrees with PCA.

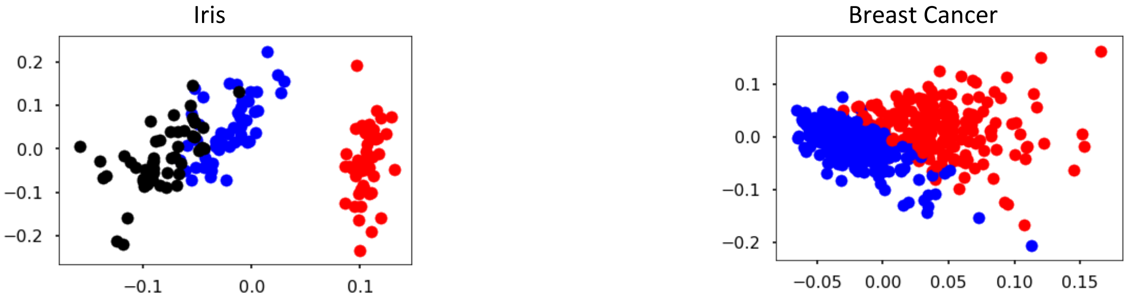


Figure 4.4 scatter plot with target class after dimensionality reduction using ICA

Only 2 components are used for both datasets for visualization purposes. Compared with Figure 4.2, no additional information is given by ICA than PCA. In fact, Figure 4.4 is the same as Figure 4.2 with different data orientations. It shows for these two datasets, PCA and ICA basically identify very similar data projection.

4.3 Random Projection

There is no algorithm-specific metrics to evaluate the performance of Random Projection. However, since dimensionality reduction projects high-dimensional data to lower dimensions, there is information loss during this process. The information loss, or can be quantified by computing the difference between original dataset and the reconstructed dataset by projecting its lower-dimension projections back to the original dimension.

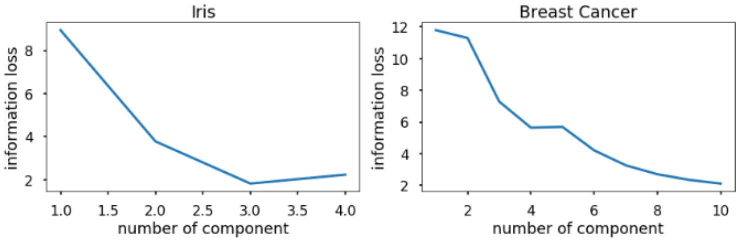


Figure 4.5 reconstruction loss for Random Projection

Note that for random projection, results are different every time random projection is performed, if random state is not fixed (see section 7 for 500 random projection instances). Therefore, results for Random Projection presented in this report are just one instance of almost unlimited possible outcomes. Figure 4.5 only shows one possibility. The information loss is the lowest when there are 3 components for Iris. On the other hand, information loss keeps dropping with number of components for Breas Cancer. There are two 'elbows': one at 4 components and one at 6 – 7 components.

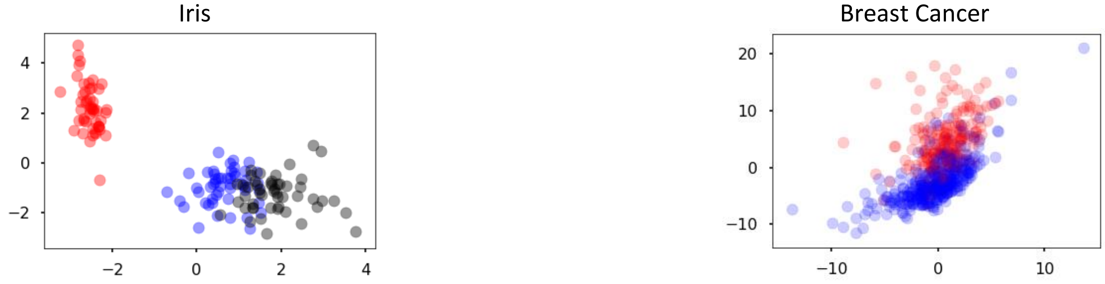


Figure 4.6 scatter plot with target class after dimensionality reduction using Random Projection

Only 2 components are used for both datasets for visualization purposes. Data points show more overlaps than PCA or ICA. Random Projection does not guarantee the hyperspace maximize data variance (as PCA) or non-Gaussianity (ICA), therefore it is no surprise that data points do not have the best clustering projection after applying Random Projection. As Figure 4.6 shows, there are some overlaps for classes 1 and 2 points for Iris, and the boundary is not as clear as PCA or ICA for Breast Cancer.

4.4 Random Forest

Random Forecast is used for classification problem. It analyzes feature importance to help reduce features that are less important. The fundamental difference between Random Forest's feature importance analysis and the other algorithms (PCA, ICA, and RP) is that Random Forest does not project the original dataset to a new, lower-dimensional hyperspace. Random Forest only ranks feature importance so that features with less importance can be discarded in future analysis. As a result, Random Forest does not change the physical meaning of the original features. On the other hand, once original data is projected to a new hyperspace, the new features would lose their physical meaning.

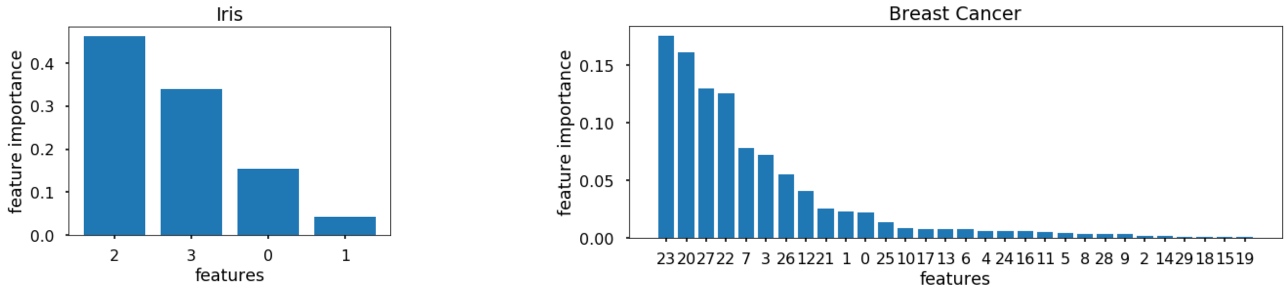


Figure 4.7 feature importance for Random Forest

For Iris, features 2 and 3 have the dominant importance in classification, followed by features 0 and 1. This observation agrees with PCA and ICA, even though for Random Forest no dimension projection is performed. For Breast Cancer, the first 8 features: 23, 20, 27, 22, 7, 3, 26, 12 are much more important than the other features. One important benefit of feature importance analysis is that after removing unimportant features, the remaining features do not lose their physical meaning.

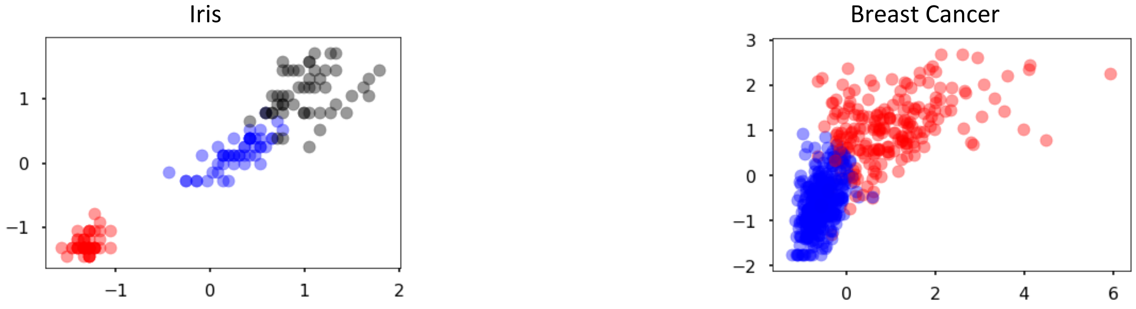


Figure 4.8 scatter plot with x and y axis being the most important features

Figure 4.8 shows data points distribution for Iris and Breast Cancer, using only the most two important features as axes. It is not surprising to see for Iris distinct clusters are present. Class 0 (red dots), similar to PCA, ICA and Random Projection, is distant from classes 1 and 2 (blue and black dots). On the other hand, classes 1 and 2 form a single cluster, even though additional features might be needed to separate them.

There is no distinct separation between the two classes for Breast Cancer, which agrees with the observations in the previous sections. Note again the axes on Figure 4.8 are original features than projected components. It shows even with two different approaches, the conclusions for clustering do not differ from each other.

5. Dimensionality Reduction and Clustering

There is no universal guideline on how many dimensions original dataset should be reduced to. Based on the above analysis, the final dimensions are tabulated below.

Number of components				
	PCA	ICA	Random Projection	Random Forest
Iris	2	3	3	3
Breast Cancer	6	7	6	8

Dimensionality algorithms are applied to the dataset in the first place, and then K-Means and Expectation Maximization is applied to the new dataset for clustering. The clustering results are compared with the class labels. The results are shown in Figures 5.1 and 5.2.

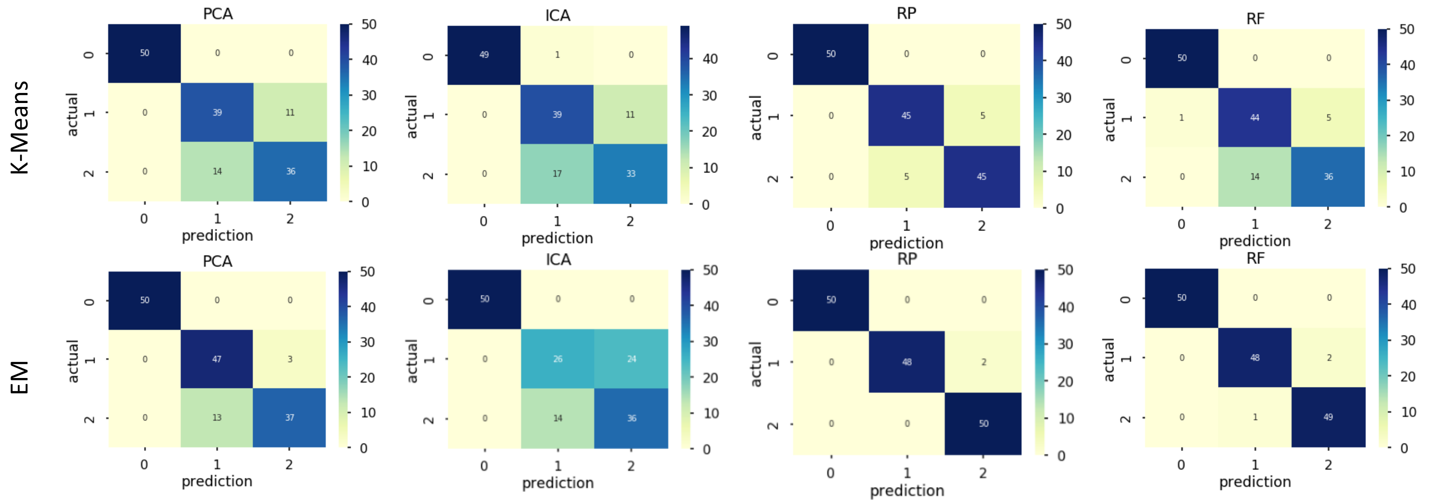


Figure 5.1 confusion matrices for K-Means and EM on Iris, after dimensionality reduction is performed

Class 0 in Iris target has its own cluster that separates from the other two classes. As a result, for all confusion matrices class 0 has the highest accuracy with at most one misclassification. The real challenge is to differentiate classes 1 and 2, especially for K-Means. Compared to the clustering results without dimensionality reduction (Figure 3.2), dimensionality reduction with Random Projection and Random Forest allows for a significant improvement over the original result, while receives no improvement with PCA and ICA. For Iris, Random Projection and Random Forest manage to maintain the key information for clustering while successful reduce the feature noise that would otherwise affect clustering quality.

Clustering results based on Expectation Maximization (Figure 3.4, no dimensionality reduction) already has a very high accuracy. It is therefore difficult to check if there is any accuracy, precision or recall score improvement after dimensionality reduction is performed. High accuracy is maintained by dimensionality reduction with Random Forest and Random Projection. However, dimensionality reduction by PCA and ICA in fact deteriorates the clustering accuracy, with more misclassification between classes 1 and 2. As shown in many of the scatter plots such as Figures 4.2 and 4.4, classes 1 and 2 are prone to be misclassified due to limited discriminating features. Projection by PCA and ICA further reduces discriminating features and results in a more intertwingled classes 1 and 2 cluster.

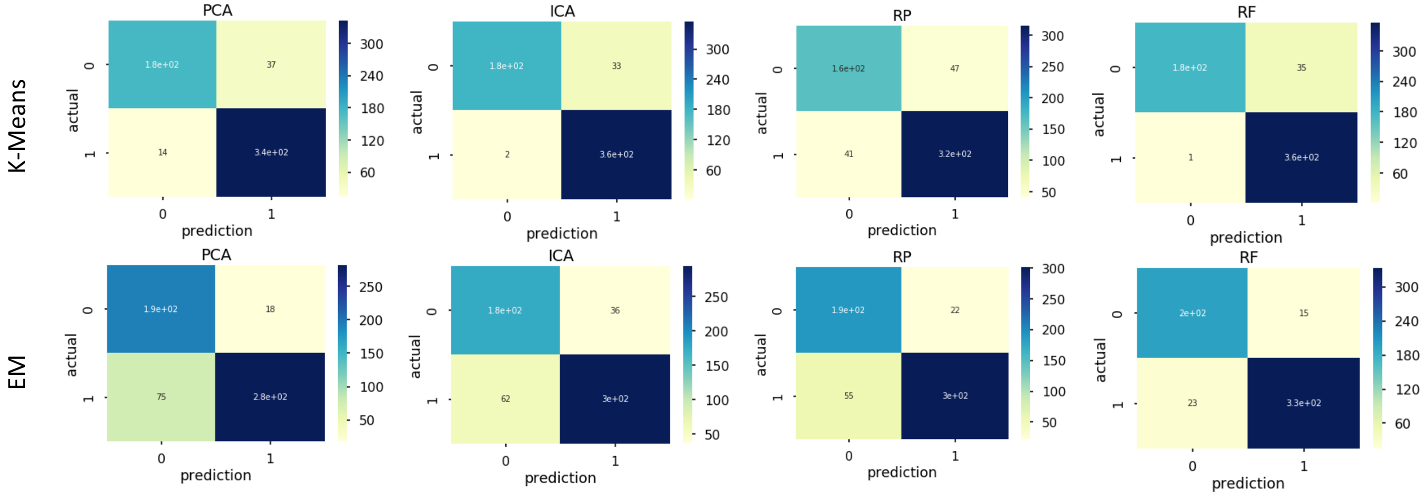


Figure 5.2 confusion matrices for K-Means and EM on Iris, after dimensionality reduction is performed

For Breast Cancer dataset, the only algorithm combination that has noticeable improvement over the original result (Figures 3.2 and 3.4) is ICA+K-Means. The other combinations either show no improvement (e.g., RF+K-Means with less false negatives but more false positives), or more false positives and negatives (e.g., ICA+EM, RP+EM). In general, dimensionality reduction not only does not help the clustering problem but also deteriorate model accuracy. The Breast Cancer dataset contains 30 features, and dimensionality reduction projects those features to 6-8 dimensional hyperspace, depending on different algorithms. Even though those 6-8 features contains most of the data information, there is inevitably information loss during this transformation. The decrease in accuracy after dimensionality reduction shows the negative effect of information loss on clustering.

6. Dimensionality Reduction on Assignment 1 Problem

Otto dataset from Assignment 1 is used again. Otto dataset is a multiclass classification dataset contains 46725 points and 20 features. All 4 dimensionality reduction/feature selection algorithms are used to evaluate feature/component importance.

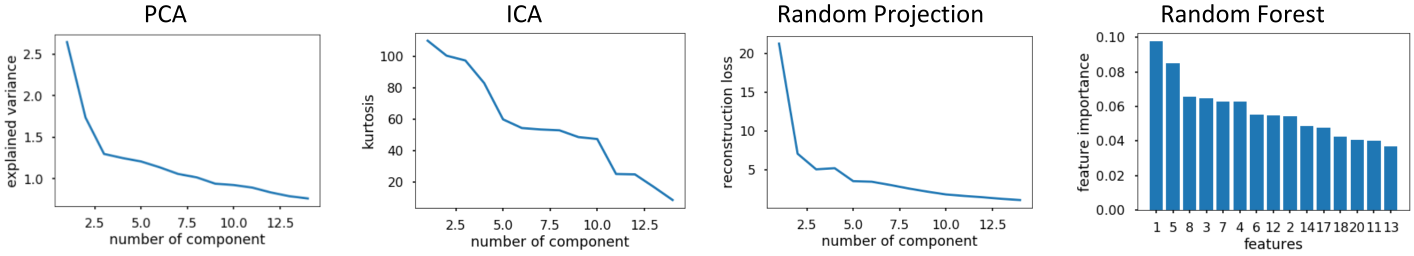


Figure 6.1 feature/component importance

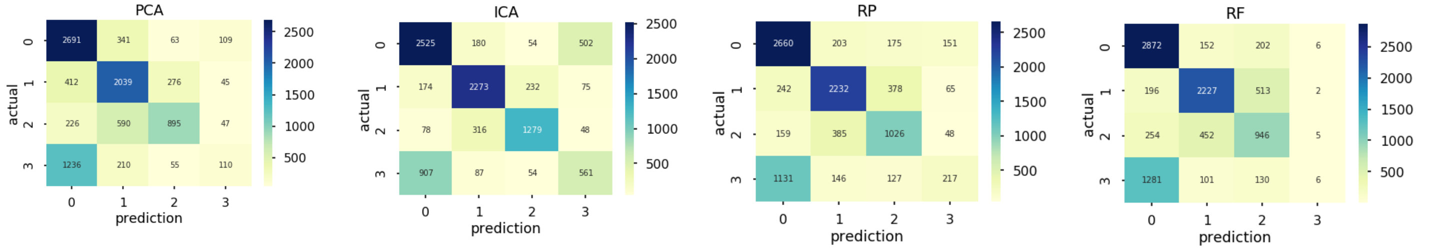
Explained variance calculated by PCA exhibits an elbow shape, where the turning point happens at 3 components. This does not agree with ICA, where the elbow happens at two locations: 5 and 11 components. Random Projection has elbow at 5 components. In Random Forest, features 1 and 2 have much higher importance than the rest of features. Another sudden drop of feature importance happens at feature 4 (6th feature).

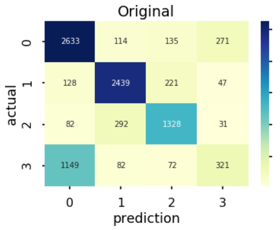
There is no consistency among different algorithms, therefore different feature numbers are selected for each algorithm. Neural network is reran based on the following dimensionality reduction strategy:

	Number of components			
	PCA	ICA	Random Projection	Random Forest
EEG	3	5	5	6

Figure 6.2 shows the confusion matrices of actual observation vs. neural network prediction, for the evaluation dataset of Otto. All neural network settings are kept the same for all experiments. Notice that darker color means higher number. Dimensionality reduction has been performed before neural network modeling.

For all algorithms, the key issue is the misclassification between classes 0 and 3: most of class 3 data points are categorized as class 0. This, however, cannot be attributed to dimensionality reduction. Neural network prediction based on the original data still suffers from high misclassification between classes 0 and 3. Dimensionality reduction does not help mitigate this issue. On the other hand, accuracy with dimensionality reduction is lower than that without dimensionality reduction (the diagonal cells are darker on the confusion matrix of original data than the others). It therefore shows in this case, dimensionality reduction in general 1) does not provide additional information to separate classes that are prone to be mislabeled, but 2) lose some information during projection that could otherwise be used to help separate other classes.





	Original	PCA	ICA	RP	RF
Accuracy score	0.71	0.68	0.73	0.66	0.65

Figure 6.2 confusion matrices for EEG dataset, after dimensionality reduction is performed

The only exception in Figure 6.2 is the accuracy score of ICA + neural network, which is higher than neural network based on the original data. At least from 3 datasets I tested in the report (Iris, Breast Cancer, and Otto), Random Projection always produces datasets that allow for a better clustering. To better understand Random Projection, 500 instances of random projections to 6 components are created for Breast Cancer dataset, and K-Means are applied to the new dataset. Then the accuracy score is compared with the original data.

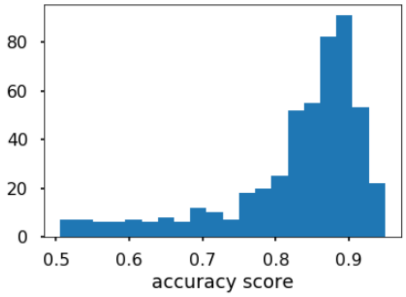


Figure 6.3 accuracy score distribution for 500 random projections, Breast Cancer data

Accuracy score from K-Means on original dataset is 0.91. Figure 6.3 shows a small portion (63/500) instance have accuracy score higher than 0.91, while the majority (437/500) are lower than 0.91. Out of those 437 instances, 126 have accuracy score lower than 0.8, with minimal score close to 0.5. Random Projection could potentially give bad results, even though statistically the majority of projections are reasonable and can lead to satisfactory clusters.

7. Clusters as New Features

After features are projected to a lower-dimensional space, clustering algorithms (K-Means and Expectation Maximization) are applied to find the best clusters. Results are below.

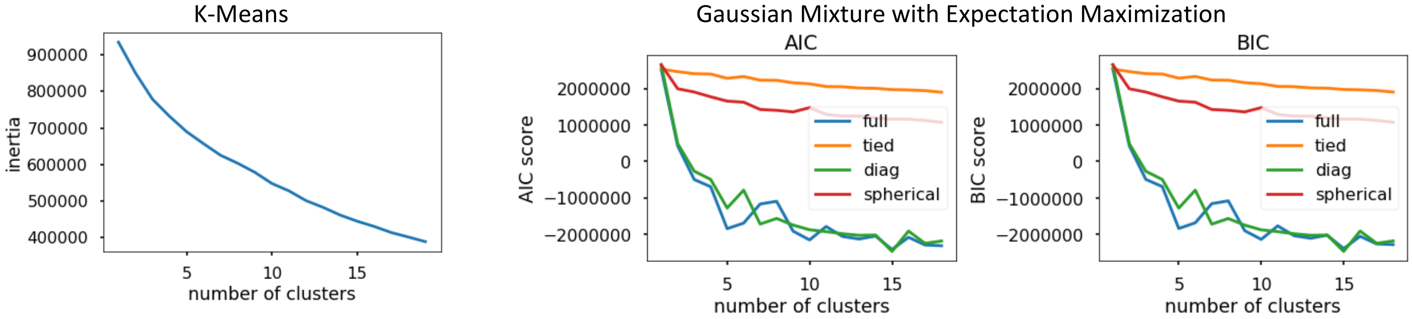


Figure 7.1 metrics for both algorithms to determine the best number of clusters

For K-Means, there is no apparent elbow to determine the best number of clusters. For Expectation Maximization, the ‘U’ shape is also missing for either AIC or BIC curves. Both curves continue to decrease as the number of clusters increases. Since the best number of clusters is difficult to determine, I use 5, 10, 15 for both algorithms to cover the cluster range. Accuracy and neural network runtime are reported below.

	K-Means		Expectation Maximization	
	Accuracy	Runtime (s)	Accuracy	Runtime (s)
5 clusters	0.74	113	0.73	92
10 clusters	0.73	117	0.73	110
15 clusters	0.73	129	0.73	134

As a comparison, the accuracy score on the original data was 0.71, and neural network runtime was 90 seconds.

Having clustering results as extra features is proven to improve the neural network accuracy by 2%-3%, for both K-Means and Expectation Maximization. Clusters add additional information to the original dataset. These labels serve as pre-defined group indicator to help neural network quickly converge to the correct classification.

Model accuracy scores for different clusters are slightly different. 5 clusters generated by K-Means tends to have the best lift to accuracy score compared to 10 and 15 clusters. Having more clusters not only does not provide additional useful information to help neural network achieve higher accuracy, but in fact adding noise to the data and eventually reduces accuracy. Even with more training time, the accuracy scores from higher clusters (10 and 15) are in lower than with 5 clusters.

The argument above also holds true for Expectation Maximization. Even though the accuracy scores for all 3 cluster choices are the same, the training times differ quite significantly. The shortest training time (5 clusters) is 67% of the longest training time (15 clusters).

The trade-off for higher accuracy is training time. With more features, neural network uses more time for training. Note that the cluster labels do not just add one feature to the original dataset. The number of new features is the same as the number of unique labels. As a result, the number of features added to the original data is quite substantial given a lot of clusters. With more cluster labels, neural network has more weights to optimize, and therefore requires more time for training.

8. Summary

In this assignment I tested 2 clustering algorithms, 4 dimensionality reduction algorithms, and combination of those. I then applied clustering and dimensionality reduction algorithms to assignment 1 dataset, and compared the results with assignment 1

Clustering

1. ‘Elbow’ method is an effective method to select a reasonable cluster number, so that inertia is small but also allowing enough points for each cluster. The choice of ‘elbow’ is subjective, especially when ‘elbow’ is less apparent.
2. For Expectation Maximization, AIC and BIC curves are often used to determine a cluster number. Due to different covariance types, AIC and BIC curves behave differently. The number of clusters is chosen when AIC or BIC curve is at its minimum, which also indicates that the current model has the most likelihood to represent the data points.

Dimensionality Reduction

1. Variance and eigenvalue are two importance metrics to determine the number of principle components for PCA. Although they measure different matrix values, their relative distribution among all principle components are the same. The choice of number of components are subjective and sometimes could be unclear. For example in the Breast Cancer case, other than the first principle component, the value for others do not show significant difference.
2. Kurtosis, the key metrics to determine the number of independent components for ICA, can be both positive and negative. As long as the independent component has a kurtosis that is away from 0 (0 means gaussian distribution, which violates the consumption of independent component), it is a potential candidate to be kept.
3. The projection results differ every time when Random Projection is run. It does not guarantee the best components after projection, but it is a much faster algorithm compared to PCA and ICA.
4. Feature selection based on Random Forest’s feature importance differ from the other 3 algorithms since it does not project original dataset to a lower-dimensional space; instead, it only selects features with high importance. A great benefit is that features do not lose their physical meaning after feature selection, as compared to the other algorithms. This is of great importance where model explainability is required.

Dimensionality Reduction and Clustering

Dimensionality reduction either projects higher-dimensional dataset to a lower dimension (PCA, ICA and Random Projection), or ranks features so that only the most important ones will be selected (Random Forest). Such process could lead to a new dataset that might either improve or deteriorate the clustering results later on. On one hand, dimensionality reduction could reduce the features to only the ones that could maximize the inter-cluster distances, so that clustering algorithm can be performed with high accuracy. Moreover, any projection to lower dimensions would inevitably lose information. The information could be noise, which could help improve the clustering accuracy. When the information is useful one, however, clustering based on the new dataset might suffer accuracy loss.

In this assignment, both situations are encountered. On Figure 5.1, where Random Projection + K-Means helps improve clustering accuracy, ICA + K-Means sees a worse accuracy score. Note that for both experiments, the number of components are the same, therefore it eliminates the effect of different dimensions.

Dimensionality Reduction on HW1 Dataset

In general dimensionality reduction on HW1 dataset does not help improve clustering accuracy. The only algorithm (Random Projection) that help improve clustering accuracy is unlikely to be reproduced unless the same random seed is reused. It turns out the dataset suffers information loss after dimensionality reduction, which can otherwise be used to help cluster data points.

500 instances of random projection are performed, and clustering results after these instances are compared with that without random projection. K-Means on only a small portion of projected data is able to obtain an accuracy score better than on the original dataset. However, K-Means can get reasonable accuracy on the majority of projected data. Therefore in cases where the importance of efficiency surpasses the importance of accuracy, Random Projection is a great candidate.

Clusters as New Features

Clusters labels are used as additional features. This pre-processing provides cluster information that could later be used for classification, therefore in general helps improve neural network accuracy. In the experiment, the effect of the number of different clusters on accuracy does not vary significantly. For the range of clusters tested, they all have similar lift to the accuracy score. The biggest difference is model training time. With more clusters, more feature columns have to be created, and as a result, more weights have to be optimized when running neural network. Such extra work requires more training time, which is positively correlated with the number of clusters. In application, one should probably try to start with fewer clusters before moving to more, which should usually give a good balance between accuracy improvement and training efficiency.