CS 4641 Project3 —— Unsupervised Learning

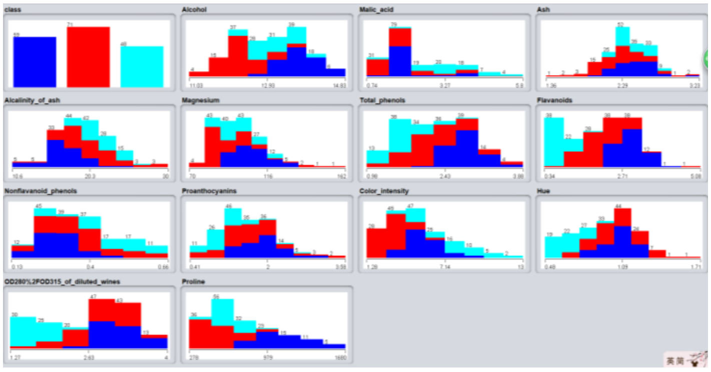
Name: CHANG Yingshan
GT Student ID: 903457645
GT Account: ychang363

Dataset Description

1. Wine (from python, sklearn.datasets.load_wine)

The data is the results of a chemical analysis of wines grown in the same region in Italy by three different cultivators. There are thirteen different measurements taken for different constituents found in the three types of wine.

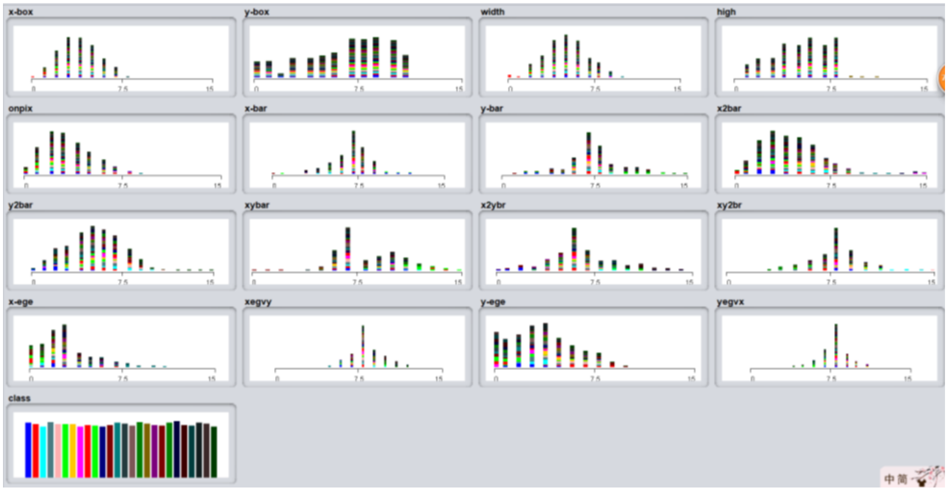
Classes	3
Samples per class	[59,71,48]
Samples total	178
Dimensionality	13
Features	real, positive



2. Letter recognition (Same as the one used in my project 1)

This is a classic dataset for a multi-classification task, whose objective is to identify each of a large number of black-and-white rectangular pixel displays as one of the 26 capital letters in the English alphabet. The character images were based on 20 different fonts and each letter within these 20 fonts was randomly distorted to produce a file of 20,000 images. 16 numerical attributes were measured for each image, which were then scaled to fit into a range of integer values from 0 through 15.

Num_classes	26
Samples per class	[789,766,736,805,768,775,773,734,755,747,739,761,792, 783,753,803,783,758,748,796,813,764,752,787,786,734]
Samples total	16000
Dimensionality	16
Features	Integer 0~15, positive

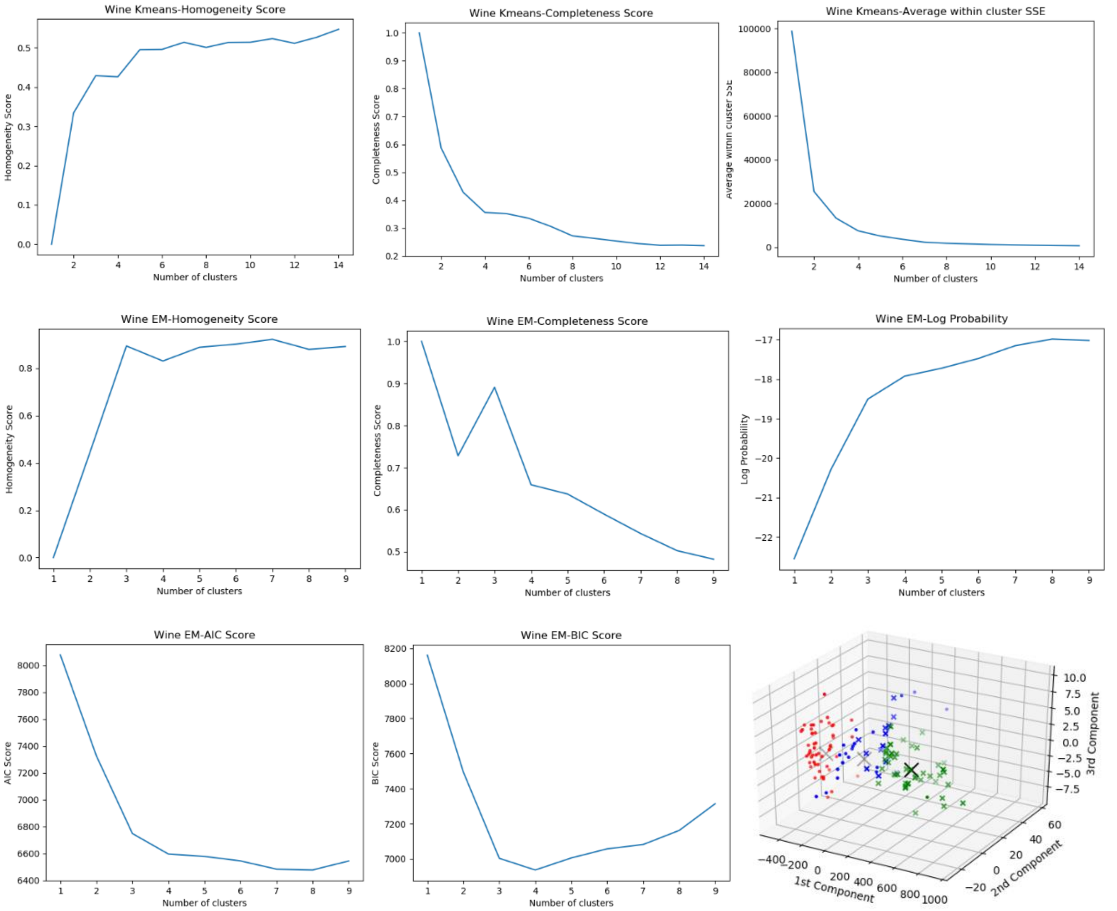


Clustering

1. **K-Means**
    a) Overview
    K-Means is implemented using *sklearn.cluster.KMeans*. Euclidean distance is used. The clustering algorithm will run multiple times with K being an integer value in [1,14] and [1,30] for Wine and Letter-recognition respectively. There will be detailed analysis of choosing the appropriate K later according to the outputs.
    b) Evaluation method
        i. Homogeneity: describes how each cluster contains only members of a single class.
        ii. Completeness: describes the degree in which all members of a given class are assigned to the same cluster.
        iii. SSE: within cluster sum-of-squares error
2. **Expectation Maximization (EM)**
    a) Overview
    EM algorithm is an iterative method to find maximum likelihood estimates of parameters. Here we assume that data points in every cluster follow a Gaussain distribution. Therefore, the parameters involved are mean and variance. EM is implemented using *sklearn.mixture.GaussianMixture*.
    b) Evaluation method
        i. Homogeneity describes how each cluster contains only members of a single class.
        ii. Completeness: describes the degree in which all members of a given class are assigned to the same cluster.
        iii. Log likelihood
        iv. Akaike Information Criterion (AIC): gives an estimate of a model performance on a new, fresh dataset. AIC is given by the formula: AIC = -2 * loglikelihood + 2 * d, where d is total number of parameters. The lower AIC score signals a better model.
        v. Bayesian Information Criterion (BIC): When fitting models, it is possible to increase the likelihood by adding parameters, but doing so may result in overfitting. The BIC resolves this problem by introducing a penalty term for the number of parameters in the model. The penalty term is larger in BIC than in AIC. The lower BIC score signals a better model.
3. **Elbow Method**

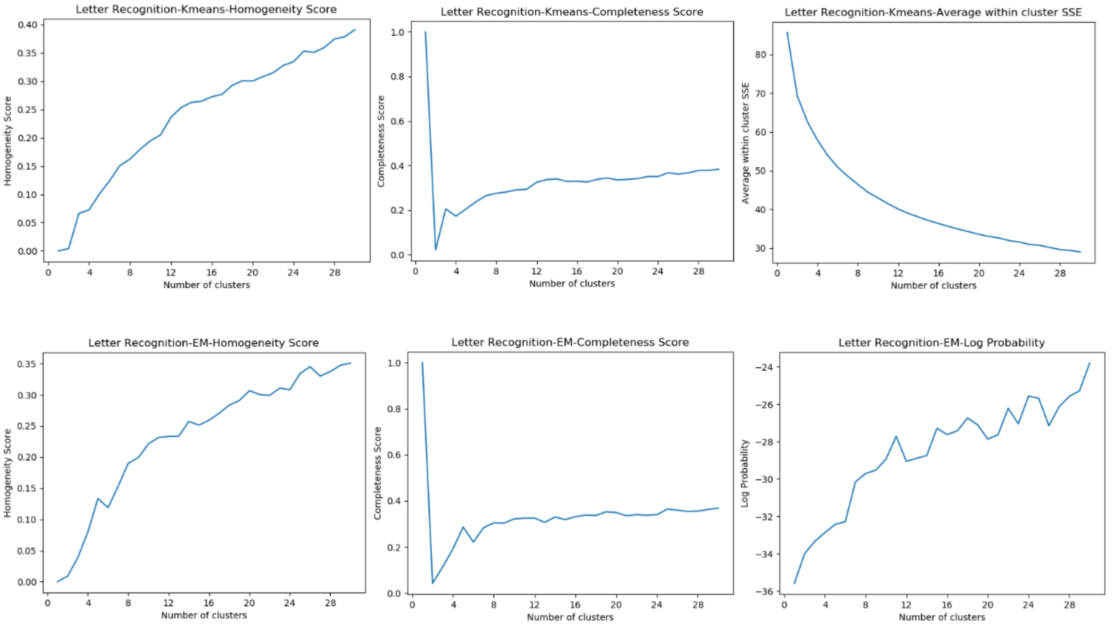
Elbow Method looks at the percentage of variance explained as a function of the number of clusters. One should choose a number of clusters so that adding another cluster doesn't give much better modeling of the data. More precisely, if one plots the percentage of variance explained by the clusters against the number of clusters, the first clusters will add much information (explain a lot of variance), but at some points the marginal gain will drop, giving an “angle” in the graph. The number of clusters is chosen at the point after which the curve obviously becomes flatter. For the subsequent analysis, I will use elbow method to find the appropriate number of clusters according to an algorithm’s output.

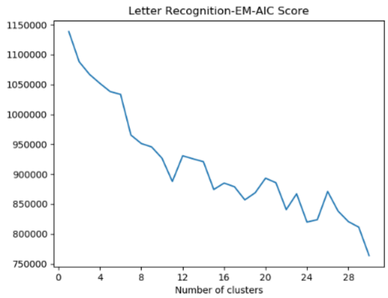
4. Clustering Result for Wine Dataset

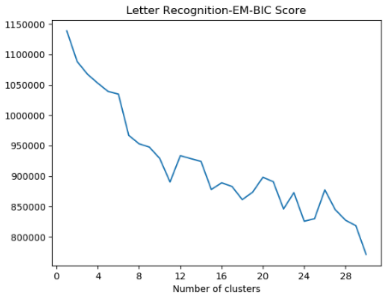


For almost all plots, the Elbow Method indicates that cluster=3 gives the best modeling of the data. Noticeably, we even see peaks at cluster=3 in the plots for EM-Homogeneity and EM-Completeness, which makes “cluster=3” particularly different from other choices.

5. Clustering Result for Letter-recognition Dataset





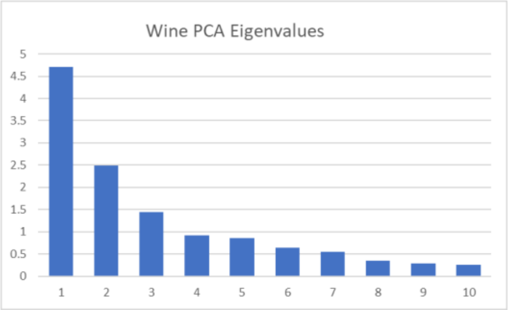


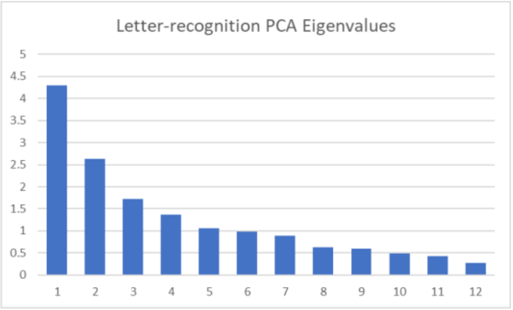
Since the number of actual classes for this dataset is relatively larger, it is harder to determine the "best" number of clusters and there may be variances across different algorithms. The KMeans-SSE curve is pretty smooth with no apparent "angle". As for homogeneity and completeness score, we notice that they keep on increasing even if the number of clusters has risen above 26. This can be interpreted as the algorithm recognizes more than one "font" of the same letter. Thus, adding more clusters means letting the model further differentiate letters appearances in detail. In the plots for EM-Log-Probability, EM-AIC, EM-BIC, we can see "spikes" around cluster=24 or cluster=25, which means adding the 24th or 25th cluster boosts the model performance in a more significant way than others.

Dimensionality reduction

1. Principle Component Analysis (PCA)

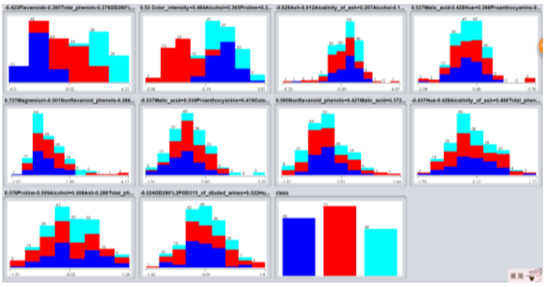
PCA uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of linearly-uncorrelated variables. The first principal component should account for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. The eigenvalue involved in orthogonal transformation conceptually represents the amount of variance accounted for by a factor.

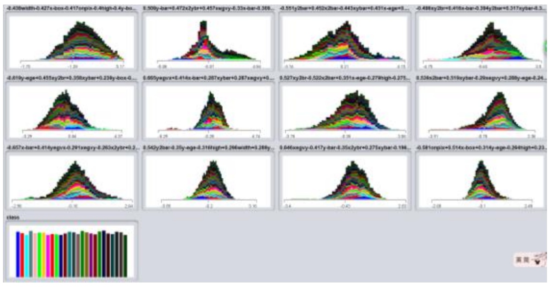




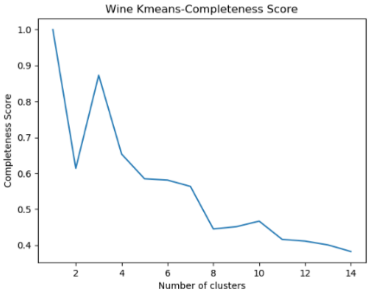
By analyzing the eigenvalue for each principal component, we can identify which components have greater importance in classifying the data and which have less. Those components with extremely small eigenvalues are also noteworthy because they may even add more noise to the model and negatively impact the model performance.

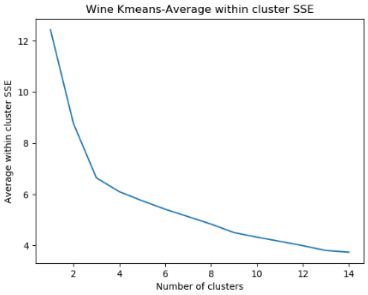
The following two images show the two datasets after PCA transformation.

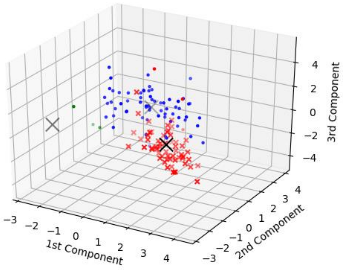


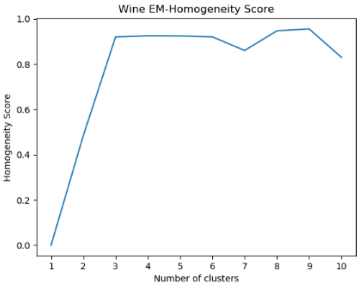


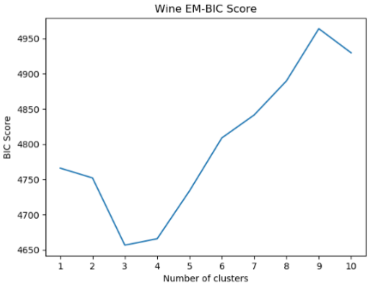
- ➢ <u>PCA clustering analysis on Wine dataset.</u>

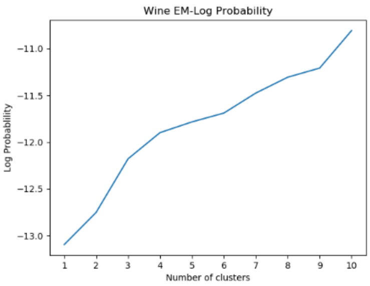






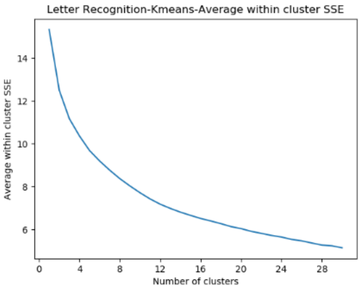


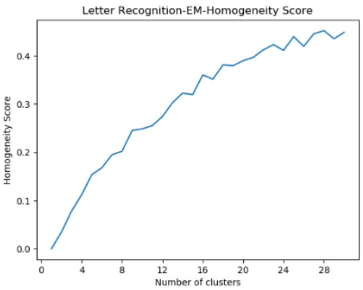


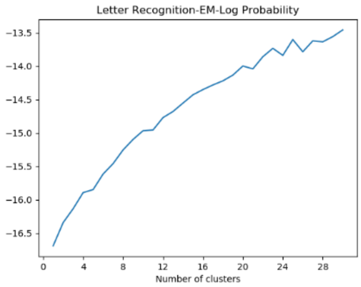


We can see apparent "signals" defined by Elbow Method, indicating that cluster=3 should be the best choice. It is also clear that since the KMeans-SSE and EM-BIC are significantly lower than the scores on the original dataset. Moreover, in stark contrast to the model trained on original data, whose EM-Log-Probability seems to converge at -17, this time EM-Log-Probability is continuously rising above -11 as the number of clusters grows, showing no signs of convergence at the current stage, which also indicates that PCA has a great positive effect.

- ➢ <u>PCA clustering analysis on Letter-recognition dataset</u>



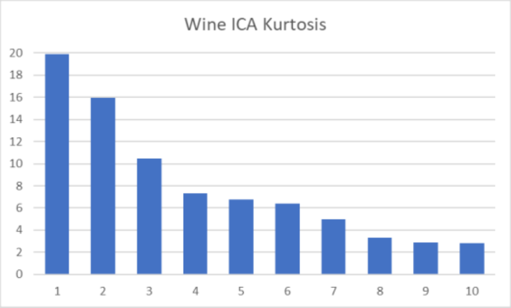


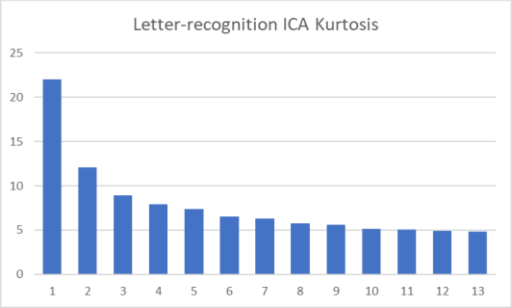


The KMeans-SSE curve is again very smooth, giving no useful information regarding the appropriate number of clusters. However, we can see "spikes" at 23 & 25 in the other two plots. Given the actual number of class is 26, observing this result is quite reasonable. Comparing with the results of original dataset, applying PCA has significantly reduced Kmeans-SSE and EM-Log-Probability.

2. Independent Component Analysis (ICA)

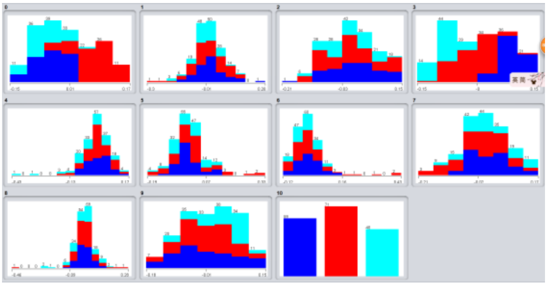
ICA tries to reveal hidden factors that underlie the data. In the model, variables are assumed to be linear mixtures of some unknown and mutually independent latent variables, and the mixing procedure is also unknown. ICA aims at factoring out those underlying independent components.

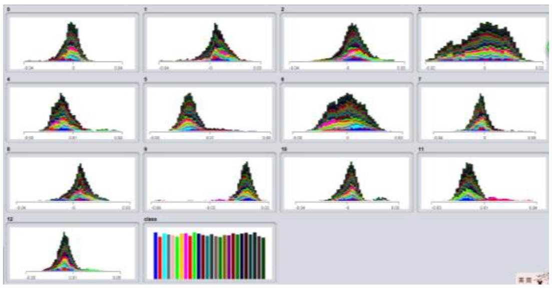




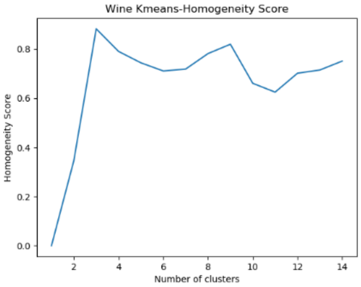
It has been shown that a necessary condition for ICA to work is that the hidden signals be non-Gaussian. Therefore, when finding each independent component, ICA will find the direction of the largest non-Gaussianity. We use kurtosis to measure the degree of non-Gaussianity. A good independent component should have large positive kurtosis, whose distribution is characterized by a "spiky" probability density function. Components with kurtosis close to zero do not contribute much worthy information to classification task.

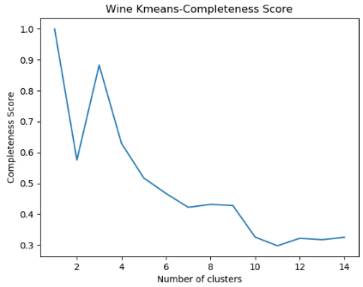
The following two images show the two datasets after ICA transformation.

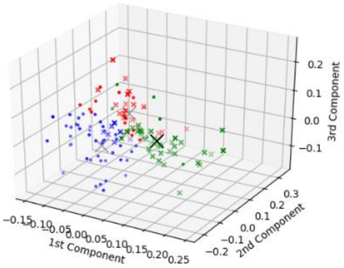


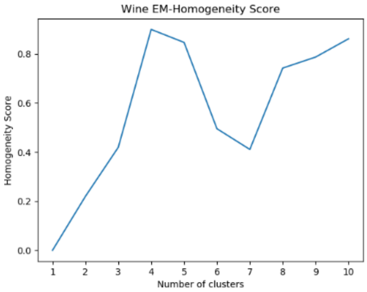


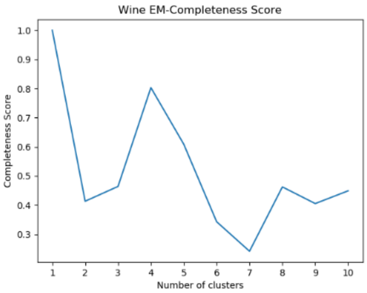
- ➢ ICA clustering analysis on Wine dataset





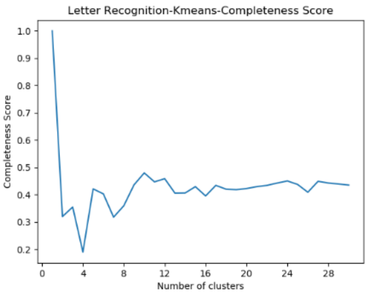


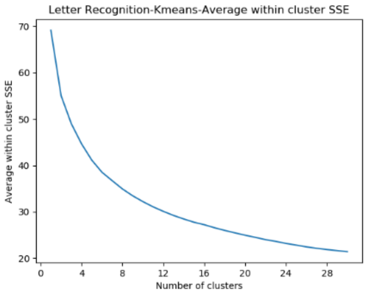


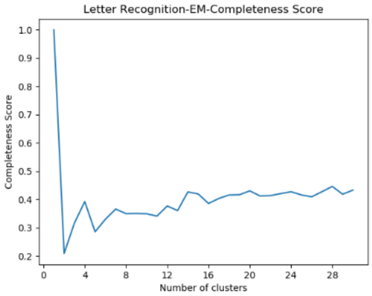


Clustering analysis is applied to the dataset transformed by ICA with 10 components. It is interesting to note that homogeneity and completeness scores of Kmeans show spikes at 3,9, while these two scores of EM show sharp angles at 4,8, which indicates that both algorithms find some reasonable ways to classify the data in more detail (split the dataset into more than 3 categories).

- ➢ ICA clustering analysis on Letter-recognition dataset





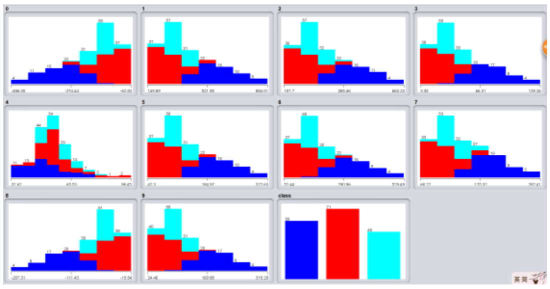


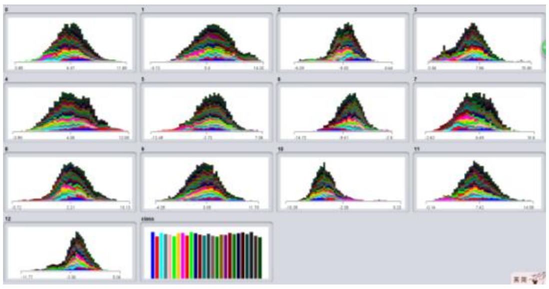
For Letter-recognition, it is not obvious to tell the good cluster number. Judging from the plots, we see peaks around 24 and 28, but they are not very notable. The KMeans SSE does not improve much. One possible explanation is that it is not appropriate to think of the data as a mix of independent dignals in this case.

3. Random Projection (RP)

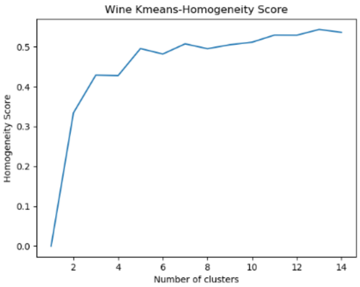
RP projects the data onto a random lower-dimentional space. RP is known for being significantly computationally cheap.

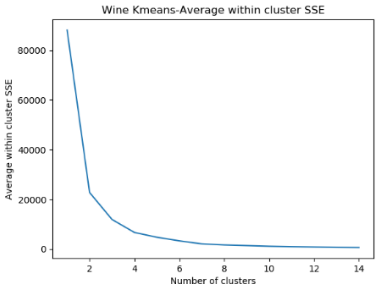
The following two images show the two datasets after RP transformation.

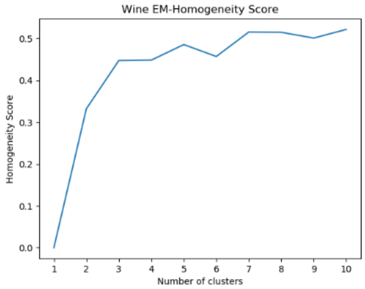


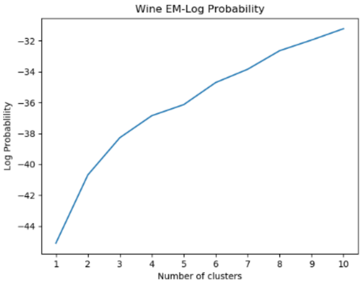


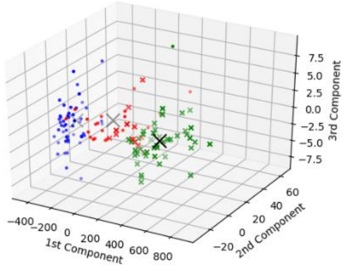
- ➢ RP clustering analysis on Wine dataset





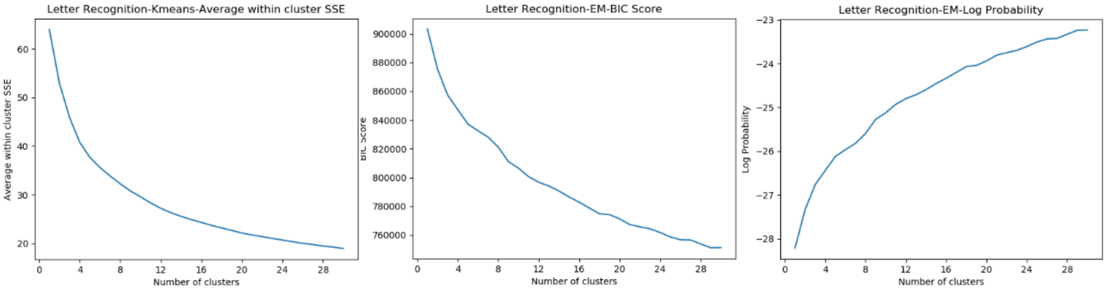






I used RP to reduce the wine dataset dimensionality to 8. Via Elbow Method, we can decide that the appropriate cluster number is 3, which is consistent with the previous analysis. However, after RP transformation, the KMeans-SSE does not reduce much and the EM-Log-Probability is even worse than the original dataset. Perhaps this illustrates the trade-off between computational simplicity and goodness of an algorithm.

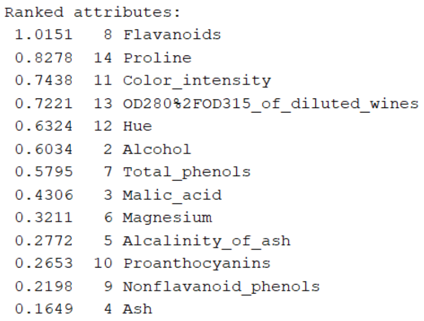
- ➢ RP clustering analysis on Letter-recognition dataset



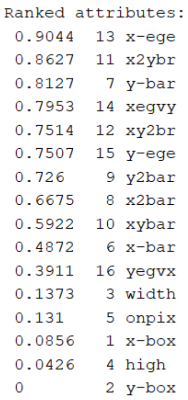
For Letter-recognition, I reduced the dimensionality to 13 by RP. In contrast to the clustering result on RP-transformed Wine data, it turned out that KMeans-SSE, EM-BIC and EM-Log-Probability have all improved, albeit to a small degree. Thus, it is reasonable to assume that, compared to other state-of-art algorithms that might be sensitive to disturbance, RP is less erroneous on complex datasets that are expected to contain more noise and uncertainty.

4. Information gain

Information Gain evaluates the attributes by measuring the information gain respecting the class. According to the ranked attributes, we can drop those attributes with small info gain to reduce noise. The following images show the ranked attributes for Wine and Letter-recognition calculated by Weka.



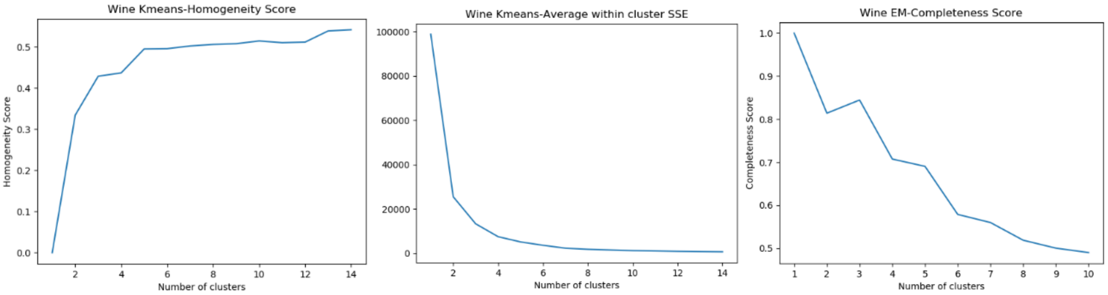
```
Ranked attributes:
 1.0151    8 Flavanoids
 0.8278   14 Proline
 0.7438   11 Color_intensity
 0.7221   13 OD280%2FOD315_of_diluted_wines
 0.6324   12 Hue
 0.6034    2 Alcohol
 0.5795    7 Total_phenols
 0.4306    3 Malic_acid
 0.3211    6 Magnesium
 0.2772    5 Alcalinity_of_ash
 0.2653   10 Proanthocyanins
 0.2198    9 Nonflavanoid_phenols
 0.1649    4 Ash
```

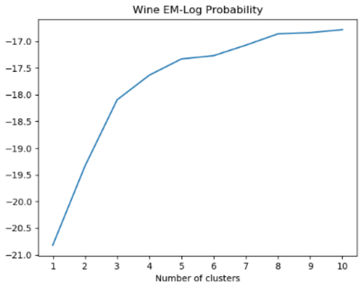


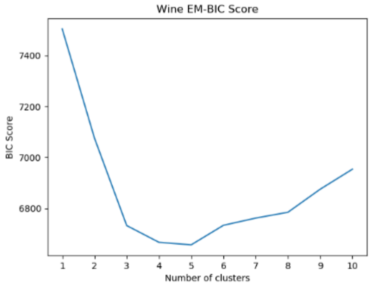
```
Ranked attributes:
 0.9044   13 x-ege
 0.8627   11 x2ybr
 0.8127    7 y-bar
 0.7953   14 xegvy
 0.7514   12 xy2br
 0.7507   15 y-ege
 0.726     9 y2bar
 0.6675    8 x2bar
 0.5922   10 xybar
 0.4872    6 x-bar
 0.3911   16 yegvx
 0.1373    3 width
 0.131     5 onpix
 0.0856    1 x-box
 0.0426    4 high
 0         2 y-box
```

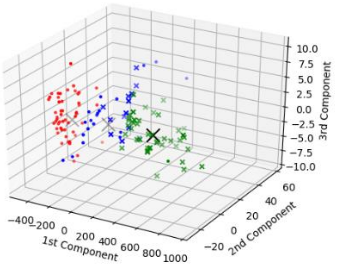
Clustering analysis for Info Gain is done by dropping the attributes with info gain less than 0.3 and re-run the two clustering algorithms.

- ➢ Info Gain clustering analysis on Wine dataset



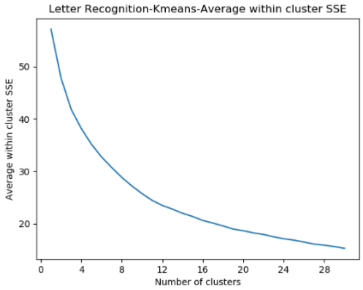


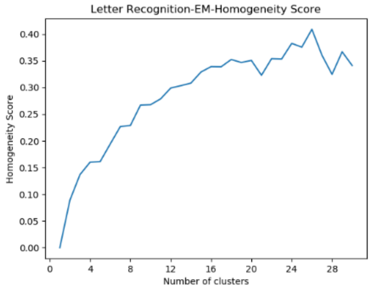


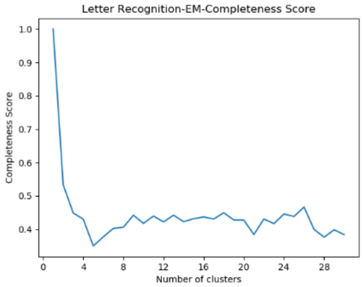


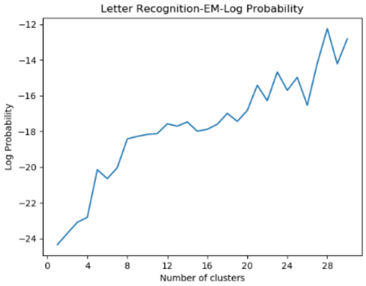
There are noticeable signals derived from homogeneity and completeness curves for choosing cluster=3. The EM-BIC has decreased by dropping the least-significant attributes, while the KMeans-SSE and EM-Log-Probability didn't improve much. The clustering results are close to the original dataset because we do not transform the data but only select a sub-set.

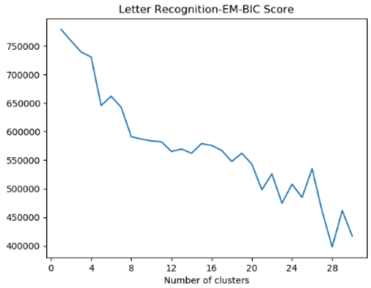
- Info Gain clustering analysis on Letter-recognition dataset









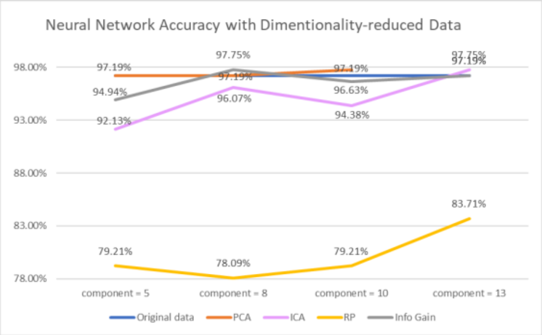


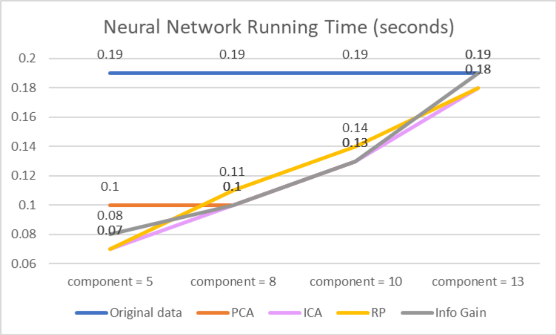
As can be seen from the plots for EM-Homogeneity, EM-Completeness and EM-BIC, there is a significant spike at cluster-26, which makes a lot of sense. It is also noteworthy that SSE is reduced and EM-Log-Probability has increased. Even though we just dropped some "useless" features and didn't do any reshaping or reconstructing work, the improvement is remarkable, indicating that Info Gain is a useful analysis tool to get rid of noisy and disturbing factors.

Neural Network Performance

1. Dimensional Reduction and Neural Network

In this part, we transform the data by four dimensionality reduction algorithms, with target number of components being 5, 8, 10, 13 respectively. Then, we run the neural network to classify the data and analyze the cross-validation accuracy as well as training time. The neural network is implemented by Weka with default parameters (learning rate = 0.3, momentum = 0.2)



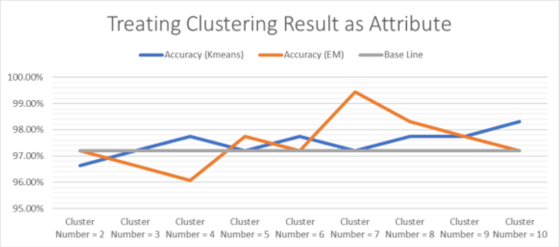


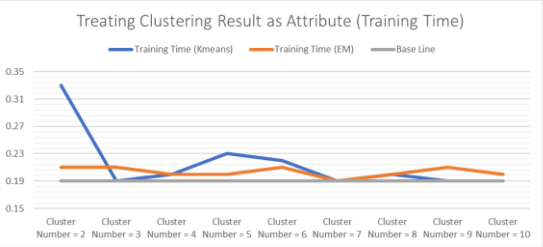
In terms of classification accuracy, PCA, ICA and Info Gain all have outperformed the original dataset in some cases, while RP is much worse. This could be attributed to the fact that, although RP performs well on high dimensional datasets, its output is highly unstable. Since our Wine dataset is regular and contains less noise, RP's drawback outweighs its advantage for this task.

In terms of training time, there is not much variance between different algorithms. It might be concluded that training time is highly related to the number of components (attributes).

2. Clustering and Neural Network

In this part, we introduce the clustering result as a new categorical feature and append it to the dataset. Then, we run the neural network to do classification and analyze accuracy as well as training time.





It is clear that adding clustering result as a new attribute increases accuracy, especially for the case where we added the clustering result of EM. And the improvement is more noticeable when cluster number is larger than the actual class number. This illustrates that the clustering algorithms can contribute more worthy information when they divide the dataset into finer categories. Meanwhile, adding clustering result does not affect the training time much, since we only add one more attribute.

Reference

https://github.com/danielcy715
https://github.com/Shally1130/CS7641-assignment3
https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_wine.html
http://www.cs.ucf.edu/~gqi/CAP5610/CAP5610Lecture13.pdf
https://www.quora.com/What-is-the-difference-between-PCA-and-ICA
https://web.engr.oregonstate.edu/~xfern/rpm_icml03.pdf