pchow7
Machine Learning - Fall 2015
Assignment #3

Introduction

For assignment #3, unsupervised learning methods for clustering and components analysis/dimensionality reduction were applied to “Online News Popularity” (News) and “Wine” dataset from the UCI machine learning repository. The key take-aways are the practical applications of both types of algorithms, as well as potential workflow for tying both approaches to real world datasets and questions. For this assignment, I flip flopped between languages and packages to extract the best visualization and tools for analysis but remained consistent when comparing results against each other

0. About the Datasets:

News are articles attributes with a continous target feature of total shares per article from Mashable.com and Wine covers composition of different wines with a target value for quality, both datasets are from the University of Irvine Machine Learning Repository (UCI-Repo).[1] For the purpose of these assignments, “shares” is synonymous to “popularity.”

News is interesting classification problem because it attempts to find the “DNA” of digital content popularity. The 59 attributes in the dataset are associated with and describe the article but none are explicitly related to popularity. The purpose of classification is to tease out unknown patterns of digital behavior/preference. The dataset contains a lot of attributes with unknown significance and could potentially produce noise, which can lead us astray.

In assignment #3, reusing this data set for dimension reduction is especially interesting because this dataset did not respond well to supervised learning, aka the model accuracy from assignment were low and never exceeded more than high 70% accuracy. I did in fact take a stab at reducing the dimensions from this dataset to reduce the complexity and run time of the models. This portion of assignment #3 will engage in a more formal version of what I attended in assignment #1. I will finally observe how to increase the accuracy of predicting probability, as well as more precisely decreased the size of the dataset for application of a neural network (and other supervised learning techniques).

The second dataset is the UCI wine dataset. The datasets I used thus far related to interactions of features related to a success action. The wine dataset is an exploratory exercise for discovering what makes “good wine” with features that inherent describe and are attributes to its quality. There is much more propensity for Wine dataset to produce better learning models. During these assignments, I hope to gain more exposure to different and domain knowledge.

[1] Kelwin Fernandes - INESC TEC, Porto, Portugal/Universidade do Porto, Portugal. Online News Popularity Dataset: http://archive.ics.uci.edu/ml/datasets/Online+News+Popularity (last accessed September 9, 2015)

A. Online News Popularity Dataset

1. Unsupervised clustering (with all features of @ dataset)

Description of input and output: Input original Datasets (News-Original and Wine-Original) to clustering methods to output with clustering labels: News-Kmeans1 and News-GMM1

On both datasets, k-means clustering and expectation maximization were performed to find the "best" grouping for each of the datasets given screen graphs and composition of each cluster.

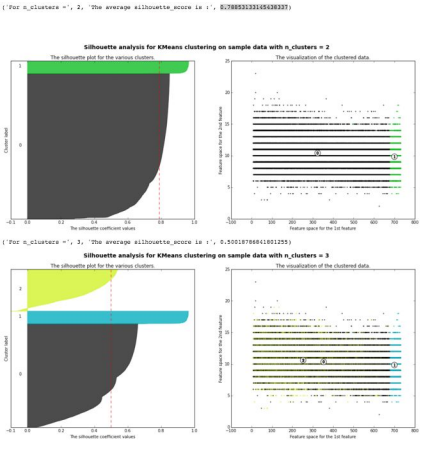
k means clustering is a method of separating groups of features in a dataset to minimize inertia or the sum of squares within a cluster of points. Kmeans typically uses Euclidean distance to measure distance between clusters, other distances may be applicable but there is guarantee of convergence for different measure. With enough time kmeans will always converge.

One of the first questions I have for each datasets is what is the most efficient way via unsupervised learning models to split the rows. For News, assignment #1 labels were applied "manually" by allocating a group by range of number of shares only and did not consider other features. The purpose for News in assignment #1 was to discover patterns and attributes that contributed to higher popularity. Wine came with original target labels from research.

After running **News** through KMeans, the clusters are compared with visualizations of how attributes are distributed within each cluster. Visualizations are good ways to view many different attributes at once. As seen below, every k data cluster greater than 2 has significant overlap in points in the left most section. This result is not surprising, it was equally as challenging to discover rules in assignment #1 and supervised learning algorithms accuracy rates were low. In assignment #1, I could not say with confidence what made a popular article.

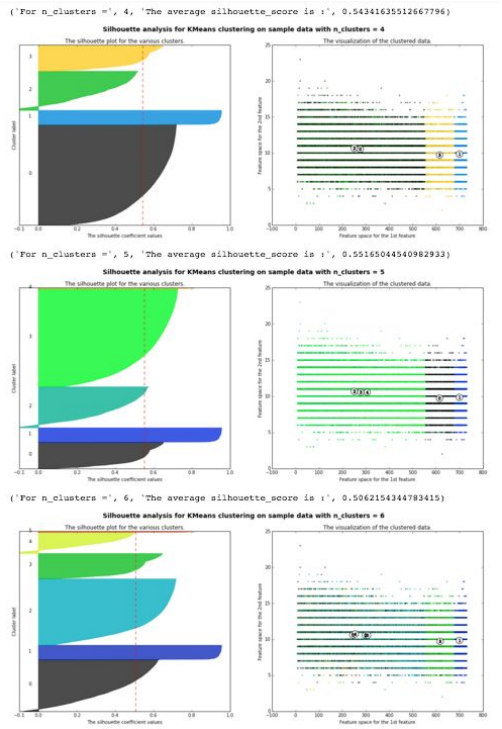
# of clusters	Silhouette
2	0.789
3	0.500
4	0.543
5	0.552
6	0.506

News silhouette analysis compared k clusters silhouette values. Values in the silhouette coefficients closer to +1 means the sample is farther away from the neighboring clusters, a value of 0 is when the sample is on or very close to the decision boundaries between clusters, and -1 indicates that values may have been given to the wrong clusters. We can judge the average silhouette values from all points overall and compare to find the "best" value of k.



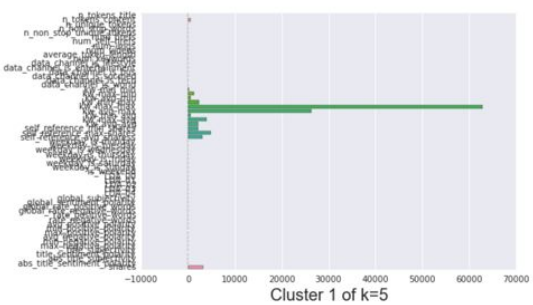
The kmeans algorithm is able to handle large datasets very well but not as good as handling irregular shapes and is not a normalized measure. The overlapping of clusters make it enticing to apply other distance measures, beside Euclidean, and different

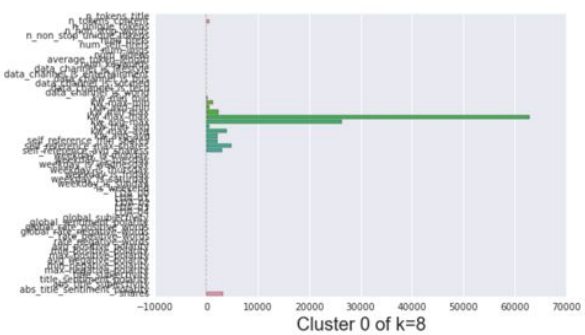
shapes to be represented inside the dataset. For the original dataset without dimensionality reduction or transformation of the data (i.e. scaling or normalization) 2 clusters provides the best separation between clusters. Here we can see the trade off with silhouette analysis, where more distinctiveness means less cluster and ability to extract insights. Choosing clusters for this data set is dependant on the desired outcome. Even with more number of clusters, distinctiveness is preserved and can still provide insights for article content and publishing. As expected and seen in assignment #1, the features in this dataset are difficult to separate.



Below are descriptions and averages by feature for cluster configurations 2, 3 and 5. I also manually created labels for 2,3 and 5 groupings based on ranges of shares. It difficult to make sense of the clustering. Applying my business question on what attributes in an article makes it worth sharing, I compared the means of feature from different n_clusters. For 2 clusters, articles that are shared more on average (group 0) have greater values of keyword features, number of shares from referenced hyperlinks, which denotes a relationship between current article and related articles on the site, and contains more hyperlinks and images. For 5 clusters, the most shared group are more likely to have been posted on Tuesday or Friday, greater tokens in its title, and generally negative content and less positive words. Three (3) clusters show similar patterns with 2 and 5 clusters with difference in the weekday an article is published and channel likelihood. Here even with the highest silhouette score, k=2 clusters is not very informative. Thus, I would be more likely to opt for up to 5 clusters to extract better business insights.

Feature	2 clusters	5 clusters	3 clusters
description of group distribution (group # and number of records)	**0 36458** (greatest avg shares) 1 3186 (Group 0) This group of articles has **highest avg shares**, more negative popularity, less positive words/more negative words, more inclined to be data or entertainment channel. These articles have more hyperlinks to itself and overall and more images, where content and title length are on average longer.	**Group 3 with 21500;** Group 2 with 9004 (highest avg shares), Group 0 with 5728, Group 1 with 3185, Group 4 with 227 **MOST**: n_token_title, kw_max_avg more likely to be tuesday for friday, avg_negative_polarity **LEAST** global sentiment polarity, rate_positive_words, least likely to be socmed	**0 25952** (greatest avg shares) 2 10506 1 3186 Group 0) **highest avg shares** More likelihood that article is published on Sat, Sun, or is_week day (least likely to be published on Wed, Thur, Friday, or Monday). More likely to be in channel data, entertainment. Higher average values for content tokens, keywords, hrefs, self_refs,images, and video.

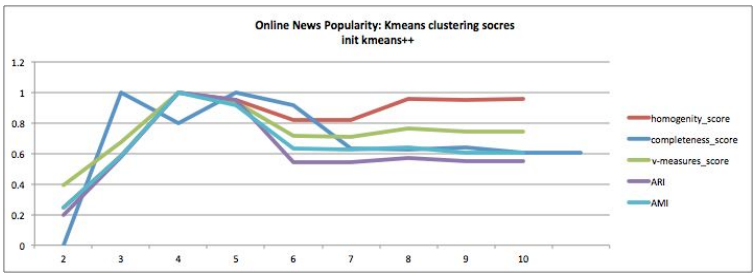


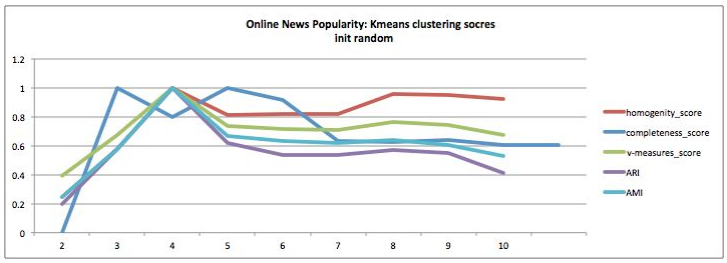


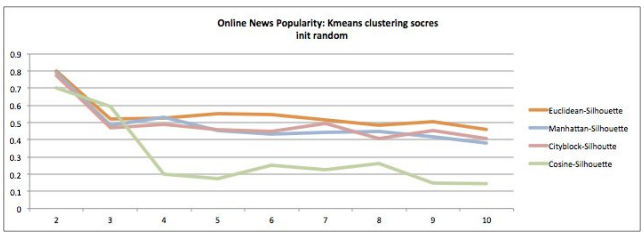
Additionally, the clusters distributions from for k=5 and k=8 are compared side by side. After k=5, the highest shares are no longer being captured inside the largest cluster. Between k=5 and k=8, k=8 most popular group has ⅓ less records (3185 records) than k=5 (9004 records). But these distributions are very similar, which denotes that common distributions pop up after k=5 and are doing it's job to separate out the nuances.

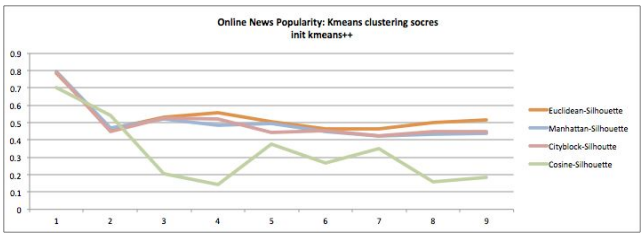
Unsupervised learning is important based on assignment #1 models, the features weighted the greatest have the magnitude in the dataset. Similarly in assignment #1, the base model was heavily influenced by "kw_" or key word metrics and "self_reference" metrics. As the number of clustered increased for News it became harder and harder to overcome larger values, which may be why silhoutte values go down.

Assignment #1 labels compared to scikit-learn applied kmeans clustering labels performed very poorly and turned out to be: 2 clusters = 2463 times or 6% of the time, 3 clusters = 727 times or 2% of the time, 5 clusters = 700 times or 1.5% of the time. The low rate of similarity shows that this dataset is much more complex than the human eye and could not be distributed by range of one features. For News, unsupervised learning with the upcoming feature selection exercise are very important to developing the right analysis. Instead of a label, shares within News becomes an influential feature.

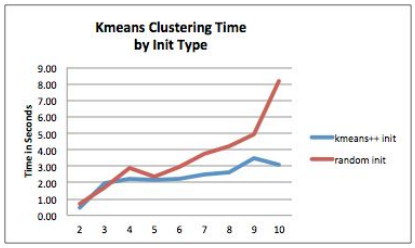






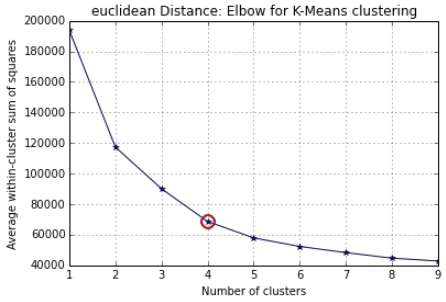


Very importantly for clustering is choosing the "right" number of groups to represent the data. Kmeans is also sensitive to the way cluster centers are initiated as a result of local minima within the data. Different ways of initiating the centers were compared, either randomly and kmeans++ in the graphs above. Kmean++ initialization in scikit-learn starts with k centroids in distance location, while random initialize could start k centroid anywhere, even very close to another centroid. These graphs depict trades off between choosing the k cluster to address the problem at hand.

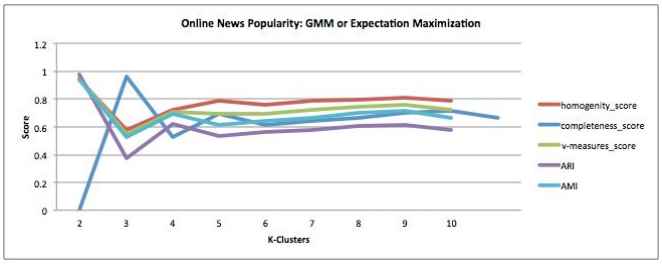


Homogeneity and completeness score are very similar between initialization methods. Homogeneity describe the clusters containing a single class, while completeness score describes the task of allocation a

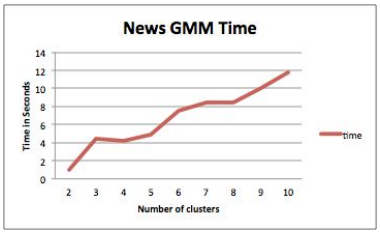
cluster to each data point. Clustering with 8 or more cluster show that each cluster will have only a single member of the class given the homogeneity score. ARI (adjusted rand index) and AMI (adjusted mutual information) shows the level of similarity with and agreement between the "true" labels and "predicted" labels, respectively. Given, there is low confidence in the original labels AMI and ARI do not sway my judgement in choosing clusters. Form the graphs, 4 clusters are optimal for kmeans++ and random initialization. Then within the same analysis, comparing the silhouette scores between different k clusters with various initialization methods shows the sensitivity of this measure by initialization. Time for initializing the algorithm by K increased more quickly when random initialization was used, compared with more steady increase of time with kmeans++. It is "cheaper" for processing time to run kmeans clustering with more clusters with kmeans++ initialization.

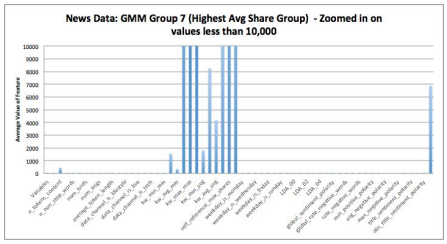


The **Elbow Method** was applied to look at the variance explained within each cluster or average within-cluster sum of squares. The optimal number of clusters from the Elbow Method is where margin of information gains start to drop with the addition of another cluster. Using the Elbow method for this dataset, the optimal number of clusters is 4. The Kmeans algorithm in scikit learn measures distances from centroids with Euclidean distance. I looked at the Elbow method from scipy using alternative distances but it did not show improvements in choosing the right K and for evaluating clusters for this dataset.



Expectation maximum (scikit learn GMM) "soft clusters", where there is no guarantee for the clusters to converge. GMM was run on News with the same cluster sizes as above, output scores based on kmeans clusters as the original labels or "true labels" and GMM as predicted labels. GMM clusters starts to level off after about 8 clusters. The graphs show at k = 3 that GMM and Kmeans label the data points the same. For GMM there is no time where all clusters will contain a single member. (homogeneity score). Adjusted rand score (ARI) is 100% similar between GMM and Kmeans at k=2 but fluctuates and levels off around 60-70% after k=4 clusters, which the approaches for distributing the groups between algorithms are different, where GMM has softer boundaries. I also applied 8 clusters for GMM to get labels for supervised learning neural networks in part #5 of this assignment.



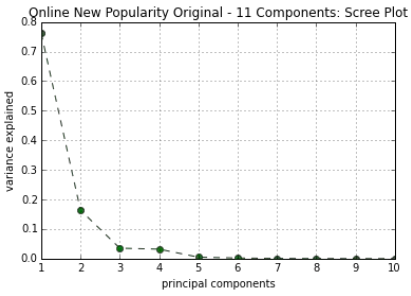


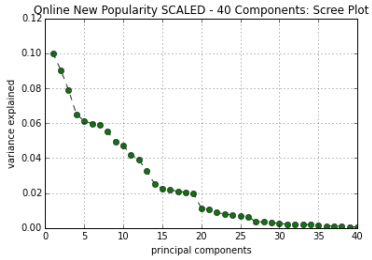
Then looking comparing the highest average share number group from the EM/GMM cluster, a similar shape appears as above with kmeans. The groupings are based on the very large values but for GMM zooming in abs title sentiment polarity showed up in this group more than it did of highest share value group for kmeans. Between kmeans and GMM,

there are slight differences in applications but here the data is forcing the groups to have similar distributions.

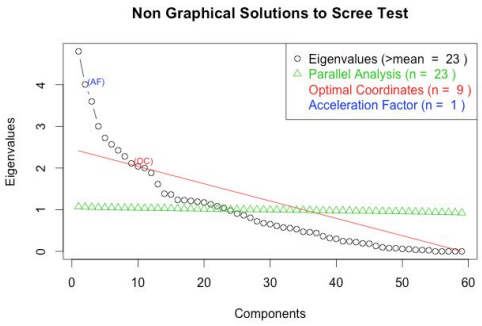
2) Dimensionality reduction

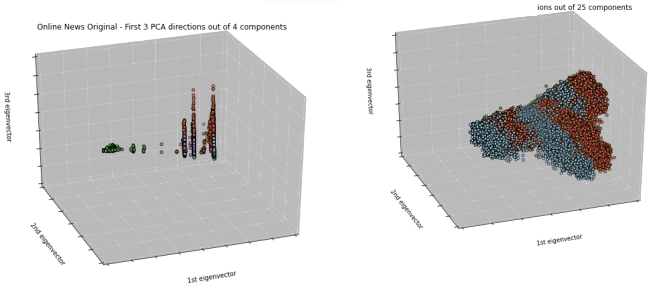
Description of input and output: Input original Datasets (News-Original and Wine-Original) to dimensionality reduction methods to output reduced data for unsupervised clustering in part #3 of this assignment, outputs are News-PCA1 (c=20), News-ICA1 (c=16), News-RCA1 (c=10), News-ECA1 (c=9) and Wine-PCA1, Wine-ICA1, Wine-RCA1, Wine-ECA1





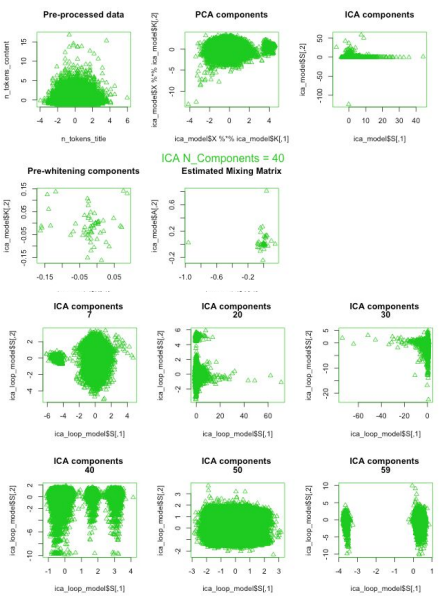
This is an exciting task for Online News Popularity since it's 59 features did not perform well with the models in assignment #1. **Principle Component Analysis** (PCA) is a very popular way to reduce the dimensions of data set, which can help visualize the information, reduce computation time, and cut out noise from the data set. The algorithm projects features to reduce residual sum of squares (increase the variance) and find a space where all features are orthogonal. Comparing the scree plots between the Online News Popularity original dataset with 11 components and scaled dataset with 40 components show the impact of scaling the data. The graphs show that the most cumulative variance explained for original/unscaled PCA is up to 4 components and scaled PCA is up to 20 components and thus the number of components that should be kept for further analysis. Output from R nFactor package with scaled data, see "Non Graphical Solutions to Scree Test" graph, shows **optimal coordinates** are with 9 features (elbow of this scree plot) and eigenvalues are meaningful until around n= 23. In **News**, articles from the same site are being compared so the differences across features and articles can be scaled to maintain information. On the contrary, if articles were from different websites and this was a competitive analysis keeping the original data would be more important. PCA assumes equal noise for all features, in contrast to ICA.

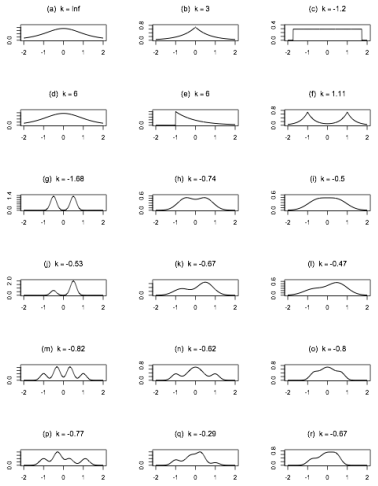




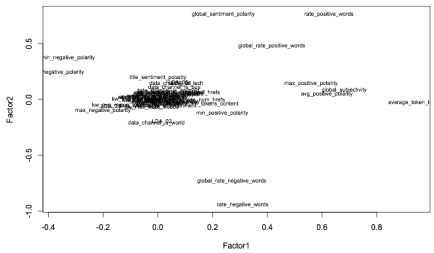
Visualizing PCA with original and scaled data shows the transformation of PCA and effects of scaling on analysis and relationships between components. The different 3rd shape of the data are discernible between the different configuration. It looks like for original data, information is being pulled towards the largest influencing features compared with scaled points cluster together and are partitioned within the dataset. Another view in R of the PCA components with princomp with labels on unscaled data, since for scaled data values are smashed together and unreadable.

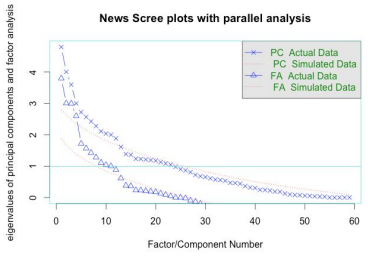
Independent Component Analysis (ICA) is maximizing independence of features in the dataset, where PCA is maximizing variance. The fundamental assumption of ICA is to discover hidden variables within the dataset given the visible datasets. ICA is taking the approach of unmixing certain patterns and projections from the data. This makes ICA seem very attractive for the News dataset since it has a great deal of noise and/or difficult pattern to discern. Below are graphs that show, on the right plots the densities and kurtosis for 18 of Source Signal Distribution within **News** data set or patterns that ICA (parallel) is finding underlying in the dataset. The green graphs on the left sides illustrates the components of FastICA, which shows the pre-processed data (including pre-whitening), aligned PCA component, and resulting ICA component. ICA components are plotted in comparison to n_components = 7, 20, 30, 40, 50. It looks for the more features for ICA, the more ICA is able to differentiate noise and find underlying patterns. The application of ICA are majority in audio and visual, where the decomposition of separation of noise and audio/photography have very different outcomes than the News dataset. Components n between 20-30 are visually separating the data in an effective way. Components at 7 and 40 still look like unrecognizable clumps of points. 20 are chosen to go forward to ANN.





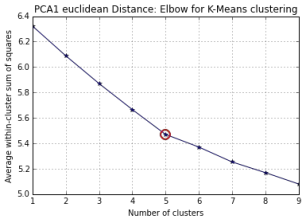
Random Components Analysis (RCA) or Random Projection generates random directions/matrix when reducing dimensions without a condition. RCA is known to carry some classification but is likely to have more n_components than PCA. Advantages to RCA are a cheaper cost of execution and computation speeds. It is difficult to visualize this portion of the assignment. In running RCA many times, different distribution of points appeared like the different variations in ICA points, see above green graph. They were different enough but most of the shapes and patterns repeated many times over.

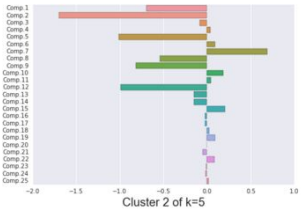


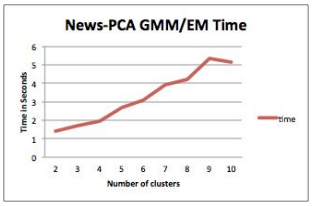


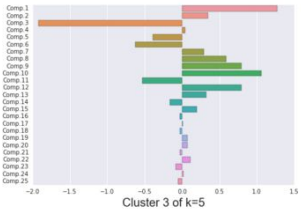
Exploratory Factor Analysis (EFA) generates a maximum likelihood for each feature with the goal to find relationships between variables. The graph above displays with EFA mapped relationships between the variables in News, which is also the output of the model. This approach is less likely to result in factor scores. Scree plot below for News EFA suggests that 9 factors should be kept and applied in the next section. Ideally for end

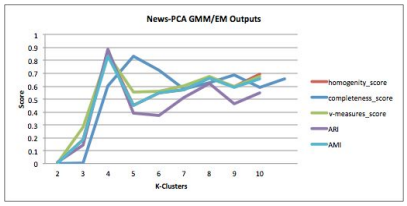
analysis on News, I would remove the features that were correlated with each other the most, aka the blog of text graphs above.

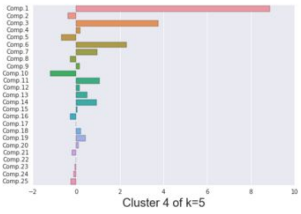








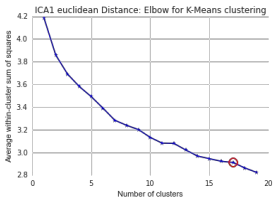


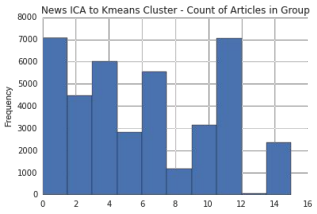


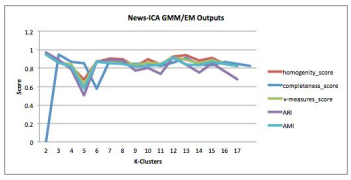
3. Unsupervised learning on reduced data. Output cluster labels for #4

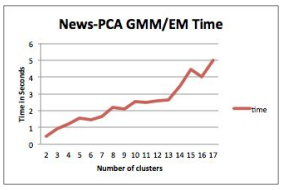
On the News-PCA-Kmeans data, the silhouette analysis is not similar to clusters in #1 since pre-processed with PCA projected features to be orthogonal and thus inherently connected and harder to separate. PCA directions between each feature is the same, as opposed to ICA. The visualization of components by group show that dimensionality reduction incorporates negative values so there is a larger working range for clusters to differentiate. Not all clusters pictured. Clustering with preprocessed data can bring out nuances in the dataset but more difficult to explain to the business. The elbow method provides that the optimal number of k_cluster = 5.

News-PCA-GMM results deviated even more than when GMM was run without dimensionality reduction, signaling that clusters are different. News-PCA-GMM k=8 clusters are optimal. Time starts to decline after 9 clusters, where it had hit it's optimal point.

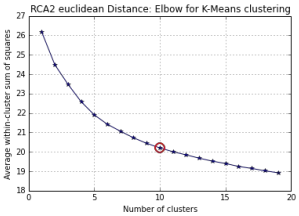


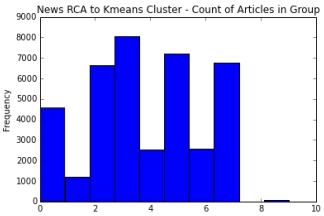


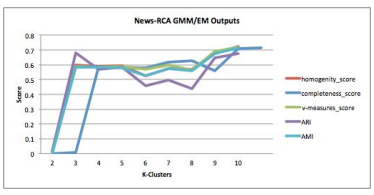


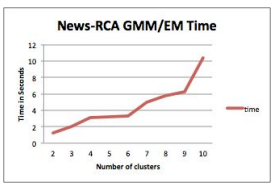


News-ICA-Kmeans - kmean clustering elbow method denoted more k-clusters for partitioning the dataset, k=16 and the distribution of the clusters show the distribution. Similar to PCA above, the cluster distributions are a mix of negative and positive values across components. News-ICA-GMM is classifying the dataset on average 85% (homogeneity score) which means that ICA k means and ICA GMM/EM are more similar than with the PCA dataset. Time starts to cost more after k=15 clusters.



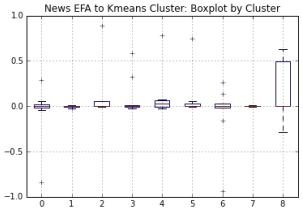


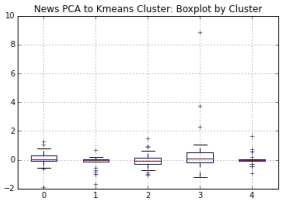




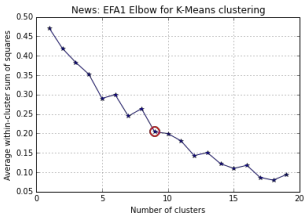
News-RCA-Kmeans showed k=10 are optimal clusters. Regardless of k, the kmean cluster allocation for RCA produces empty or group with very few members. GIven this algorithm is random and not

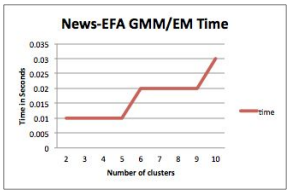
bounded by a condition this makes sense. Not the best approach for News because empty groups cannot be interpreted. Overall I see the take-away where RCA maintain some of the distribution seen in ICA and PCA but not all. RCA-GMM time wise grows more and more inefficient after k=9 clusters but the measures do not level off after 10. This is expected since RCA is not bounded so it can be across the board with distributions.

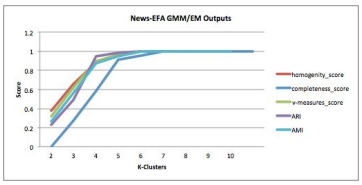




News-EFA-Kmeans scree plot shows that k=9 is optimal for maximizing the variances within each cluster. I didn't show all the boxplots for this section. EFA average values per cluster, given there are twice as many cluster as PCA, has greater range of variance. News-EFA-GMM scores converge to 100% after k=6.



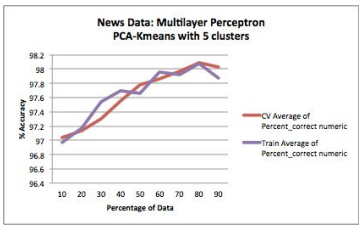


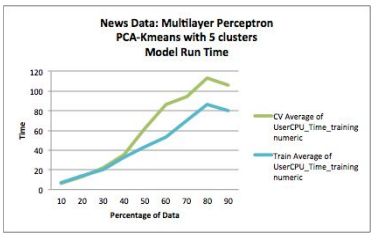


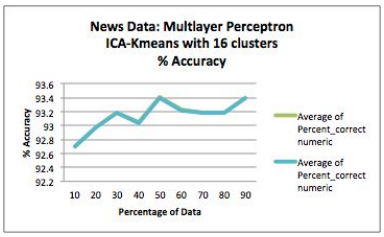
4. Dimensionality Reduced Data through Neural Networks

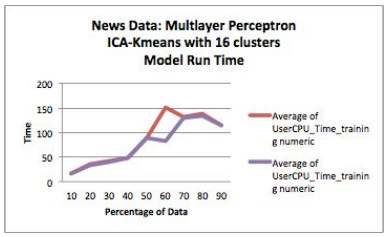
Overall the time to run neural networks (ANN) on dimensionality reduced data was less time than assignment #1 (A-#1). From graphs, training and CV accuracy rates were much closer to each other, where A-#1 there were often large variances in the training, testing, and cv scores. This is expected since the entire dataset were reduced at the same time. ANN this time around must balance overfitting, where in A-#1 the struggle was to tune any model to attain more accuracy. Time to run RCA and EFA were faster than other reduction datasets as well as much faster than assignment #1. Neural network running with ICA reduced data optimal use of data was much lower than other reduced data sets and A-#1. Although overall ICA accuracy is lower as compared with other reduced data.

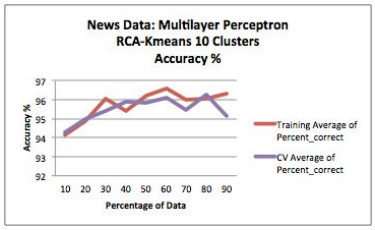
ANN for PCA provides the highest accuracy. ANN on ICA transformed data provides high level of accuracy but the most time for training. RCA reduced data performed better than ICA, however when group distribution did not always make sense. Even though RCA is property capturing the patterns and projections of the original data, it is not able to bring business sense to the problem. One concern with the results in these models are they are not as interpretable as original features, although PCA is able to reconstruction the original data.

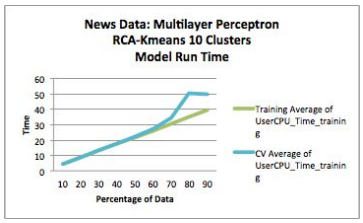


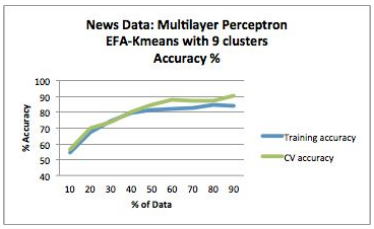


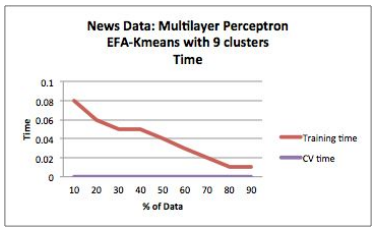






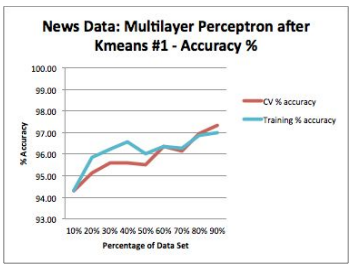


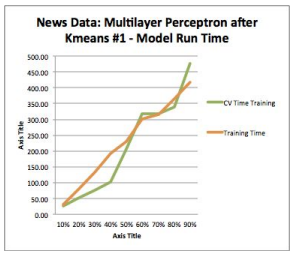


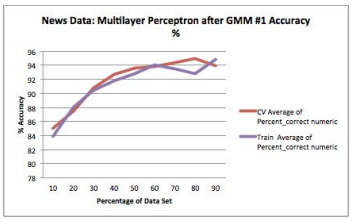


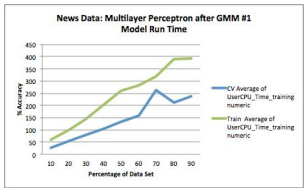
5. Neural Networks on Unsupervised Cluster Labels (from part #1)

Input from step #1: News-Kmeans1, News-GMM1, Wine-Kmeans1, Wine-GMM1



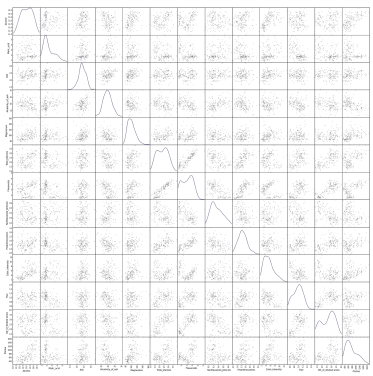






Neural Network (NN) algorithm from Weka was applied to Kmeans and Expectation Maximization (GMM) algorithms. The clustering labels from the algorithms were kept and run through NN. I removed the original labels as targets and shares remained in the dataset because they seemed to be the problematic part of assignment #1. The resulting graphs are provided below, which have much higher cross validation (cv) and training rates than assignment #1. This makes sense because the labels were created from the data itself, which could make this model stronger if the assumption is distributions of unseen data will be similar and makes this model prone to error when seeing new data because of the extremely high potential to overfit.

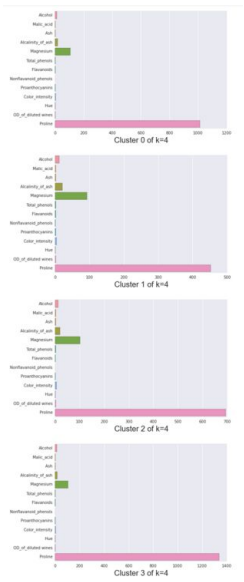
The graphs both show the optimal percentage of data needed to balance and variance is around 60% for application of neural networks. Results from the application of unsupervised learning is much more accurate than “eyeballing” the dataset. But in order to have total confidence in the model, analysis must be done on unseen or new data.

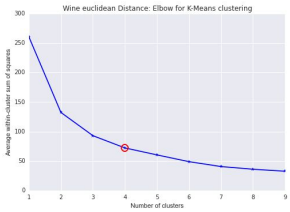


B. Wine Dataset

The Wine data start off is a much more distributed and diverse dataset, it’s scatter matrix shows discernible interactions between features. While the news data set scatter plot had a lot of binary features, which

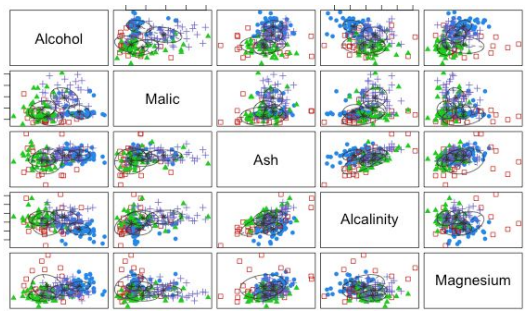
provided much less rich space for the clustering and dimensionality reduction algorithms to work in.

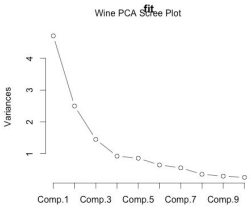


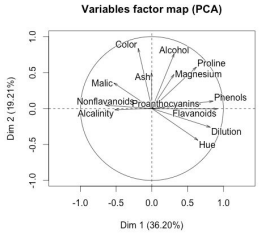


Kmeans graphs to the left show a scree plot with optimal number of clusters at 4, where the group 1 has the most number of wine. The values that we can observe with the greatest impact on the groups are Proline (also highest magnitude in value), Magnesium, Alkalinity of ash, and Alcohol content. Also in terms of quality, affect on taste and visibility, the impact of these features make a lot of sense. For example, average wine drinkers may be able to discern alcohol content the most so this value floats to the top. Finally, the 4 kmeans wine clusters are plotted to the left.

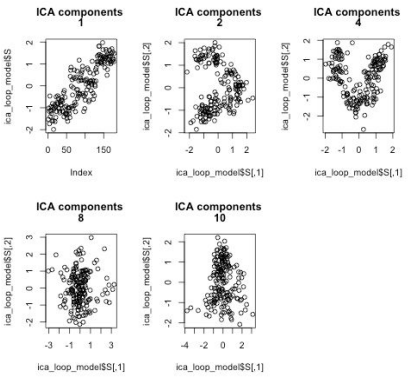
Expectation maximization (EM) clustering takes into account different shapes and irregularities of the dataset. It is apparent visually comparing the outcomes. Below is a portion of the EM scatter plot by feature by BIC value. You can see compared with finding areas of best fit, em applies various shapes, such as episodial, spherical, and diagonal to achieve cluster labels/maximize likelihood at every step.

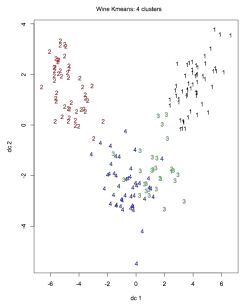


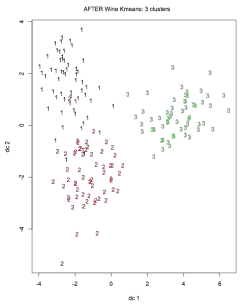




Applying different dimensionality reduction approaches provides a different way to transform the data for clustering and other analysis. PCA suggests 4 components, ICA (graph with different # of components is hard to distinguish, between 2-4 components look good), and ERA suggests 8 factors. The graph with variables factor map (PCA) provides a look into directional components of ICA (even if PCA is in the title), ICA focuses on direction while PCA aims for orthogonality of all components. Then with ICA model plots we can see the changing components with all and extracted components. Wine dataset does not seem to be as noisy filter through and change. Thus the transformation from PCA to ICA is not very dramatic. Exploratory Factor Analysis or Maximum likelihood factor analysis provides an easy output and provides the uniqueness scale of variables and suggested factors to include.







Reclustering from components analysis shows a clearer defined clustering in the wine data set. Kmeans recommended that 2-3 clusters for PCA-Wine, similar to pre-processing. The comparison with before and after clustering on wine, there is not a huge difference between the plots

Conclusion Clustering is a very rich and flexible algorithm and combined with components analysis can be very effective for discovering the space and business question. These were practical exercises in featuring tools to help transform and filter real life datasets. A huge impact for large datasets is by projecting down to lower dimensions and features, the speed of processing this data is cut down significantly. It is important to process it effective and then be able to read out the final results, thereby making PCA a very attractive means to shrink and then re-expand the dataset. Between Kmeans and GMM the algorithms took different approaches to applying clustering the dataset, as seen in comparisons especially with News dataset. However, as we projected down to lower dimensions, the labels started to converge. This is a good way to test data or hypothesis, where by applying different dimension reduction and transformation and compare different unsupervised learning to converge on the labels. Otherwise, understanding the domain is very important for choosing the right clustering technique and initialization.

In comparing the clustering labels before and after transformation, the clusters for News data were very different, as result of scaling and significantly transforming the space. On the contrary, the Wine data set clusters did not change as significantly. This is expected since the wine dataset is much less complex and large in comparison to News data.