Dimensionality Reduction and Clustering

Ian R. Stewart

October 22nd, 2018

Fall 2018 – COSC 528

Project 2

Abstract

Dimensionality reduction and clustering was performed in an unsupervised fashion on 2017 College Ranking criteria from the Wall Street Journal. Exploring the dataset proved that several data features were not useful due to non-unique attributes or large missing data points, resulting 55 Universities being analyzed with 54 features. Singular-value Decomposition is utilized to reduce the dimensionality of the data and calculate the appropriate number of singular values or principal components to use for the analysis. To meet 95% of the total variance explained, 15 principal components are used with 8 clusters. Two clustering algorithms are employed to cluster the data: k-Means and Expectation Maximization.

1 Introduction

A common task faced by high school seniors and academics alike is comparing higher education institutions to one another. For the high school senior, the task is primarily driven by the desire to select a set of universities to apply to that meet certain criteria, such as specific degree programs, tuition, location, athletic programs, religious affiliation, or cost of living. Academics, such as university professors or administration, analyze criteria of an institute for other reasons, such as increase profitability, increase the institution's ranking, or recruit high-level students and facility. For both these cases, a common place to turn to is the Wall Street Journal's (WSJ) College Ranking Report. The annual WSJ report provides multiple criteria for much of the U.S. domestic universities and calculates an index value used to compare or rank the universities. This project focuses on clustering collegiate data and compare or contrast the cluster arrangements. The analysis provided in this report utilizes the 2017 WSJ U.S. College Ranking data for 57 universities with 64 features. A k-Means clustering algorithm is created to cluster the data into groups based on specific important features. Prior to employing the cluster algorithm on the data, the data should be explored to verify the data is correct and complete, as well as identify the level of dimensionality to use. The details for the implementation of these techniques and results are discussed later in this report.

1.1 Dimensionality Reduction

Prior to arranging, a dimensionality reduction method is implemented to reduce the number of features required to perform the analysis. In an ideal scenario, feature selection and/or extraction should not be required as separate process, where the classification or regression should be employed in a manner that use and removes features, respectively, as necessary. This case can be true for low dimensionality data, but by removing the irrelevant data features from the analysis can potentially strengthen the subsequent analysis by:

1. reduce computation time due to decrease in size of matrices;
2. simpler models provide more robust results on smaller datasets;
3. increase understanding of the dataset by keeping the features that impact the data; and,
4. potential increase in graphical representation of the original data as the analysis can be visualized in a lower dimension.

To analyze the usefulness of each feature to decrease the dimensionality of the data, principal component analysis (PCA) is utilized. As an unsupervised method, PCA attempts to maximize the variance and uses the variance as the criteria to acquire the appropriate level of dimensions to use for subsequent data analysis. Essentially, PCA is an eigenvalue problem, where the eigenvectors, referred to as the principal components (PCs), provide, after the data is projected on to the eigenvalue, a more spread out or separated data. There are several methods to the PCs, but a conventional method solves for the eigenvectors of the covariance matrix from the original data. The resulting PCs are then sorted in a decreasing order from greatest to least in the amount of variance explained by each PC. That is, the eigenvector with the highest eigenvalue is the direction that the data is most spread out, which is ideal for clustering.

Rather than directly calculating the eigenvalues and eigenvectors for PCA, another method called singular value decomposition (SVD), returns the principal components and allows for the data is to be decomposed into eigenvectors and eigenvalues. To add context, let's say the data $\boldsymbol{X}$ is a $(N \text{ x } d)$rectangular matrix and can be represented as the product of three matrices: $\boldsymbol{V}$, $\boldsymbol{A}$, and $\boldsymbol{W}^{\mathrm{T}}$, as shown in the following equation.

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^{T}$$

The matrix $\boldsymbol{U}$ $(N \text{ x } N)$ contains the eigenvectors of $\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}$ in its columns, the matrix $\boldsymbol{S}$ $(N \text{ x } d)$ contains the singular values, or eigenvalues, along the diagonal, and the matrix $\boldsymbol{V}^{\mathrm{T}}$ $(d \text{ x } d)$ contains the eigenvectors of $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ in its columns. To transforms the data into the resulting PC space, the product of $\boldsymbol{U}$ and $\boldsymbol{S}$ is used. To prove this, point that PCA and SVD will result in equal solutions, a random (5 x 4) matrix was created, and two approaches were used: (1) Python's SVD function in SciPy linear algebra library (*scipy.linalg.svd),* and (2) Python's SciKit-Learn's decomposition library for PCA (*sklearn.decomposition.PCA*).

$$\boldsymbol{X} = \begin{bmatrix} 59 & 64 & 71 & 83 \\ 75 & 80 & 55 & 97 \\ 94 & 64 & 1 & 111 \\ 44 & 36 & 74 & 98 \\ 12 & 85 & 40 & 156 \end{bmatrix}$$

Using the SVD function to decompose the above matrix ***X*** into the three matrices in the SVD equation and subsequently multiplying the ***U*** and ***S*** matrixes, the following matrix is obtained.

$$\boldsymbol{X_{SVD}} = \begin{bmatrix} -1.085 & -0.420 & 0.644 & 0.169 \\ -0.201 & 0.500 & 1.046 & -0.147 \\ 0.134 & 2.108 & -0.701 & 0.033 \\ -1.406 & -1.251 & -0.911 & -0.073 \\ 2.559 & -0.937 & -0.099 & 0.018 \end{bmatrix}$$

Then using the *sklearn* decomposition library for the PCA function on the original data matrix and subsequently transforming the results into the PC-space, the following matrix is obtained.

$$\boldsymbol{X_{PCA}} = \begin{bmatrix} -1.085 & -0.420 & 0.644 & 0.169 \\ -0.201 & 0.500 & 1.046 & -0.147 \\ 0.134 & 2.108 & -0.701 & 0.033 \\ -1.406 & -1.251 & -0.911 & -0.073 \\ 2.559 & -0.937 & -0.099 & 0.018 \end{bmatrix}$$

Comparing the two resulting matrices shows the matrices are identical. This provides confidence in the usefulness and correctness of using the singular values from the SVD method.

Upon calculating these values, the amount of variance explained by each PC can be used to calculate the appropriate of number PCs or dimensions to use for the k-Means portion. A common approach is to choose an amount of variance to be explained by the lower-dimensioned matrix, say 90%, or choose a parameter where the variance slope changes drastically, called the *Elbow Method*. Both approaches use a *Scree graph,* which plots the variance explained per each PC, or eigenvector, kept for analysis. The graph looks similar to an inverse function (i.e. $1/x$), where the first PC contains the highest variance explained and decreases as the number of PCs employed increases. This graph is created later in this report and used to choose the appropriate number of the PCs to utilize.

1.2 Clustering Algorithms

The k-Means algorithm is a common clustering algorithm that fit k-number of clusters to a dataset. The k-Means clustering technique calculates the best centroids essentially in two steps: (1) assign data to a cluster based on current or initialized centroid, and (2) update centroid based on updated current assigned of data within clusters. This approach is then repeated over several iterations until the centroids no longer move or update based on a predetermined tolerance or convergence parameters. A datum is attributed to a particular cluster if the distance to the cluster's centroid is less than the distance to any other cluster centroids. A key function within this algorithm is the distance calculation, where a common approach is use the mean-square-error. To add context and clarity to the description, the algorithm is implemented as follows:

1. *Initialize cluster centroids;* $\mu_1, \mu_2, \dots, \mu_k$
2. *Repeat until converged:*

    a. For every i, calculate minimum distance to clusters.

    $$\text{cluster label for } x_i = C_i = \min\left(\left[x_i - \mu_j\right]^2\right)$$

    b. For every j, update centroids:

    $$\mu_j = \frac{\sum_{i=1}^{m} 1\{C_i = j\} x_i}{\sum_{i=1}^{m} 1\{C_i = j\}}$$

The k-Means method is straightforward and is an excellent starting point to implement a clustering algorithm due to its usefulness and simplicity. In this project, the distance is calculated by using the absolute values of the difference is used, as that is more robust approach than a more conventional squared regression procedure.

Another method is also implemented in this project, which is essentially a k-Means approach applied to a particular naïve Bayes model, called Expectation Maximization (EM). The EM method is an iterative method, similar to the previous k-Means approach, that calculates the maximum-likelihood estimates to parameters in a statistical model, which a Gaussian distribution is assumed in this case. The method alternates between an expectation step that creates the likelihood function and a maximization step that computes the statistical parameters by maximizing the previous likelihood function from the expectation step. The resulting estimated parameters are used for the next iteration of the expectation step and a convergence is checked similar to the k-Means method.

Using a Gaussian mixture as the generative model for clustering, the data is assumed to be distributed by the Gaussian function containing two parameters, the mean value and a standard deviation. The aim for this approach is to estimate the unknown, or latent, variables (i.e. which cluster a datum belongs to) based on current iteration cluster centroid, cluster standard deviation, and the datum position. This task is relatively simple if the analyst knows which points in the data belong to which distribution but can be nontrivial is the perhaps more realistic scenario where no prior or limited knowledge, such as the model parameters, is known a priori (e.g. proverbial, chicken and egg problem). To add context to this approach, let's assume an analyst is examining a set of data with two distributions, or clusters, **A** and **B**. The analyst can calculate the probability that the data $\boldsymbol{x}_i$ belongs to cluster **B** by using Bayes rule and the gaussian probability, shown in the following equations.

$$P(\boldsymbol{x}_i|\boldsymbol{B}) = \frac{1}{\sqrt{(2\pi\sigma_{\boldsymbol{B}}^2)}} \exp\left(-\frac{(\boldsymbol{x}_i - \mu_{\boldsymbol{B}})^2}{2\sigma_{\boldsymbol{B}}^2}\right)$$

$$P(\boldsymbol{B}|\boldsymbol{x}_i) = \frac{P(\boldsymbol{x}_i|\boldsymbol{B})P(\boldsymbol{B})}{P(\boldsymbol{x}_i|\boldsymbol{B})P(\boldsymbol{B}) + P(\boldsymbol{x}_i|\boldsymbol{B})P(\boldsymbol{A})}$$

The EM method begins by placing *k* Gaussian distributions randomly in the data-space, or PC space, then for each points in the data given the current distribution parameters, what is the degree of likelihood that the points came from the distributions (i.e. $P(\boldsymbol{B}|\boldsymbol{x}_i)$ equation). Once the probabilities have been computed, these update values adjust or re-estimate the distribution parameters (e.g. mean and covariance).

A key attribute of using EM is how the data are assigned to clusters is calculated, in that a value can range from 0 to 1 (i.e., probability) for multiple clusters and the highest value is chosen, sometimes referred to as *soft clustering*. Compare this method to k-Means where the value is 0 or 1 and forces the data to a cluster, referred to as *hard clustering*. Soft clustering can have overlapping clusters and provides strength of association between clusters and instances.

2 Data Exploration

This project uses a dataset from the 2017 WSJ U.S. College Ranking data for 57 universities, including the University of Tennessee, with 64 features in total. The feature labels are provided in the following table.

Table 1: Feature labels for WSJ College Rank data.

Feature Labels					
1	IPEDS#	23	Carm R1	44	HBC
2	% Hisp Total Students	24	2017 US News top 65	45	2014 Med School
3	Vet School	25	Academic Support Expenditures	46	% Grad Enroll
4	Six-year graduation rate	26	ACT/ SAT Avg	47	Fresh Admit Rate
5	% Freshmen Retention	27	Endowment per Student FTE	48	% Bachelors
6	% Doct/ Profess	28	Total E&G Expend	49	E&G / St. FTE
7	State Approp Rev	29	Tuition/Fee Rev	50	% Rev from State
8	% Rev from Tuit/Fees	30	% from State / Tuition	51	Endowment
9	Total Degrees	31	Student Faculty Ratio	52	Total Revenue
10	Total Faculty	32	ARU Faculty Awards	53	Wall St. Jourl Rank
11	ST. FTE	33	Total Research Expenditures ($000)	54	Total Expend
12	Endowment / St. FTE	34	Total Research Exp - Med School Exp ($000)	55	% UG Age 25 +
13	AG Research ($000)	35	Total Tenure /Tenure-Track Facutly	56	Faculty Academy Memb
14	Faculty FTE	36	% UG with Loans	57	Full-time Students
15	% Total Age 25 +	37	% Full-Time	58	GR Enroll Age 25 +
16	UG Total Enroll	38	GR Total Enroll	59	Doctoral Degrees
17	Part-time Students	39	UG Enroll Age 25 +	60	ACT/ SAT 75%
18	Bach Degrees	40	Masters Degrees	61	% Blk Total Students
19	Profess Degrees	41	ACT/ SAT 25%	62	Total Faculty
20	(State/ Tuit)/ St. FTE	42	Total Tenure /Tenure-Track Facutly	63	Endowment Figure
21	Total Enroll	43	Student Services Expenditures	64	Med School Res $
22	% UG Pell Grants				

When exploring the data further, three universities contained multiple missing data points. These three universities (University of Pittsburgh, University of Delaware, University and Colorado) were dropped from the data. Several data features contained non-numeric and missing values, which must be handled appropriately. Specifically, the following features contain values with missing or non-numeric values:

1. 2014 Med School,
2. Vet School,
3. Endowment,
4. Wall St. Jourl Rank,
5. Endowment / St. FTE,
6. AG Research ($000),
7. Faculty Academy Memb,

8. Profess Degrees,
9. Med School Res $,
10. Academic Support Expenditures, and
11. Student Services Expenditures.

The following steps were performed regarding the previous eleven features (in order):

1. The *2014 Med School* and *Vet School* features were changed to values 0 or 1, depending on if the University contained the particular school or not (i.e., *pre-clin* values were given values of 1).

2. The *Endowment* for Clemson University was missing and was replaced with a value obtained from the Clemson University financial reports for 2016-2017.

3. The *Wall St. Jourl Rank* were updated from the 2017 WSJ College Rankings obtained from the WSJ website. This feature contained twelve missing values in total and the university names were verified via the IPEDS # prior to verifying rank on 2017 WSJ ranking list.

4. The *Endowment / St. FTE* values were updated as the division of the listed *Endowment* amount by the *St. FTE* list values.

5. The *AG Research ($000)* was missing twelve values in total and also contained a large variance. Several of the missing values were research to obtained a value to insert for the University, but lack of data provided no help. Of note, a high variance is ideal for dimension reduction methods, such as PCA, but the feature was dropped due to large number of missing values.

6. The *Faculty Academy Memb* and *Profess Degrees* contained two and five missing values, respectively. Since the number of missing values is low, a plethora amount of time was spent to locate the missing values for both features, to no avail. Thus, the feature was dropped from further analysis.

7. The *Med School Res $* feature contained multiple missing values. This is expected as several universities do not have a medical program, but further investigation showed there were missing values from universities with medical programs. For this reason, the feature was dropped from the analysis.

8. *Academic Support Expenditures* and *Student Services Expenditures* features contained eight missing values and no data exploration efforts were able to locate the values of any of the missing values. These two feature were dropped from the analysis.

The last two features, *Total Faculty* and *Total Tenure /Tenure-Track Faculty*, were repeats, as these features were already accounted for in the data. Further investigation showed the values were different from the original feature in the dataset and the repeats (final two features) were dropped. Additionally, the first three features, *IPEDS #, Carm R1,* and *HBC,* were dropped as the *IPEDS* number is solely used to identify a University, *Carm R1* was equal for all Universities, and *HBC* only have two unique values.

In summary, the resulting data matrix contained 54 universities and 55 features, as the final feature of *Name* was dropped and used as the index for the matrix. Prior to performing dimensionality reduction, it is highly suggested to mean-center each feature in the data. If, for example, the variance of the data greatly vary then they can affect the PCs' direction. Furthermore, preprocessing the data to a mean-centered and z-standardized dataset allows for multiple features to be analyze on the same scale, where features with comparatively large units will not necessarily have a greater impact than smaller unit features.

3 Data Analysis

The 54 x 55 data matrix was analyzed by using dimensionality reduction schemes and k-Means clustering. Recall that conventional methods for PCA utilize the covariance matrix for find the eigenvalues and eigenvectors. For this reason, the covariance matrix overlaid with a heatmap is provided in the following Figure.

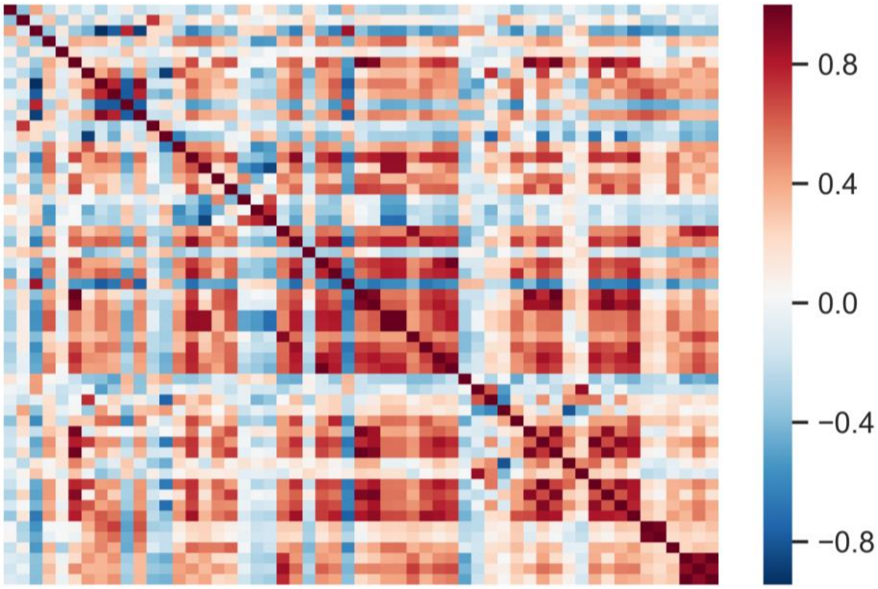

Figure 1: Covariance matrix heatmap of the final 54 x 55 data matrix.

From the covariance figure, an analyst typically examines for strong relationships, either highly correlated or uncorrelated. In this case, strong correlations are at the extremes of the scale, -1 (dark red) or +1 (royal blue). Moving forward, the mean-centered and standardized data matrix $\boldsymbol{X_{ST}}$ was decomposed via SVD, resulting in three matrices with corresponding shape:

$$\boldsymbol{X_{ST}}(54 \text{ x } 55) = \boldsymbol{U}(54 \text{ x } 54) * \boldsymbol{S}(54 \text{ x } 54) * \boldsymbol{V^T}(54 \text{ x } 55)$$

The variance for each singular value, provided in the $\boldsymbol{S}$ matrix, can be used choose the appropriate number of PCs during the clustering algorithm. The following tables provides the variance for each PC and the cumulative variance for the first twenty PCs.

Table 2: Variance for the first twenty principal components.

No.	Variance	Cumulative Variance
1	39.90%	39.90%
2	11.29%	51.19%
3	9.64%	60.83%
4	7.37%	68.20%
5	5.12%	73.32%
6	4.35%	77.67%
7	3.46%	81.13%
8	2.80%	83.93%
9	2.61%	86.54%
10	2.02%	88.56%
11	1.86%	90.42%
12	1.61%	92.02%
13	1.10%	93.12%
14	1.05%	94.18%
15	0.80%	94.97%
16	0.70%	95.68%
17	0.64%	96.32%
18	0.56%	96.88%
19	0.43%	97.31%
20	0.38%	97.69%

As expected, the first PC contains the largest variance and decreases as the number of PCs increases. Recall that a well-known method for deciding on the number of components or dimensions to use for analysis can be chosen by predetermined variance explained value (e.g. 90%) or by creating a *Scree graph* and employ the *Elbow method*. The *Scree graph* is shown in the following figure.

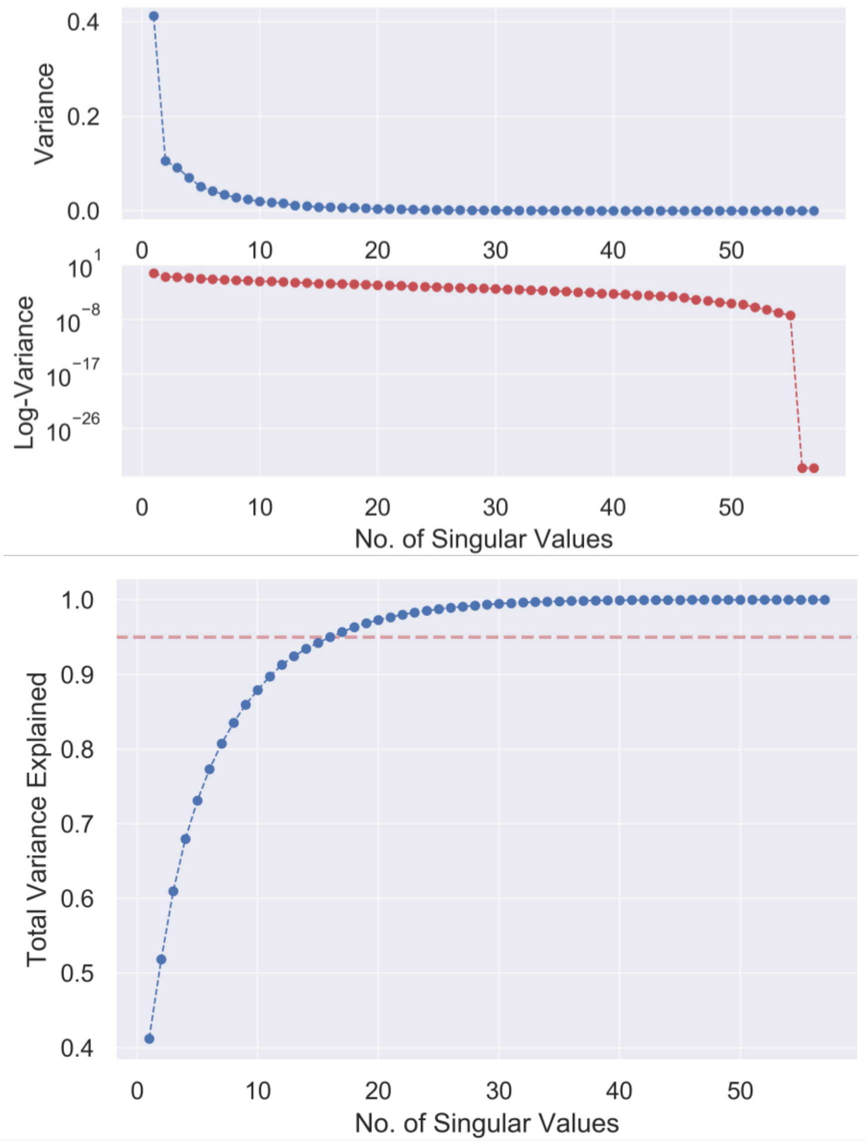

Figure 2: (*Top*) Variance as function of singular value for both linear and log-variance. (*Bottom*) Cumulative variance explained.

For this analysis, a total variance explained value of 95% will be used to determine the number of PCs to use. This value corresponds to approximately 15 PCs (see Table 2). To use this many PCs, a function (*getValues*) was created to calculated the PCs using SVD and extract the number of PCs, or columns, requested. In an attempt to add context to the reader, the first two PCs are plotted versus one another in the following figure. The first PC, shown along the x-axis, has a range of roughly $[-10, +15]$ while the second PC, along the y-axis contains a range of approximately $[-6, +8]$. This is expected, as the largest variance should be captured by the first PC; that is, the larger the range in values results in large variance.

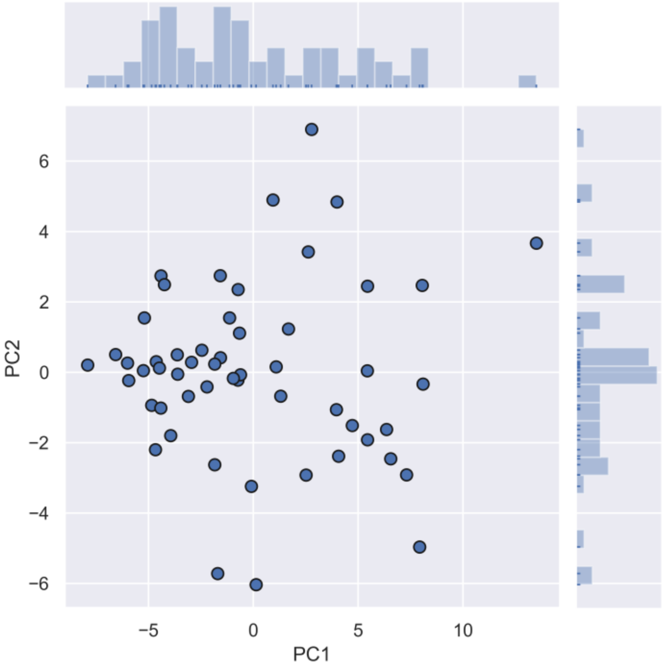

Figure 3: PC1 and PC2 showing initial separation of data.

Using the first two PCs, the k-Means class created (*K_means*) was used to cluster the data into one (for a sanity check), two, and three clusters. The results are provided in the following Figures with the centroids of each cluster shown by an uppercase *X* and the University of Tennessee shown by an orange star.

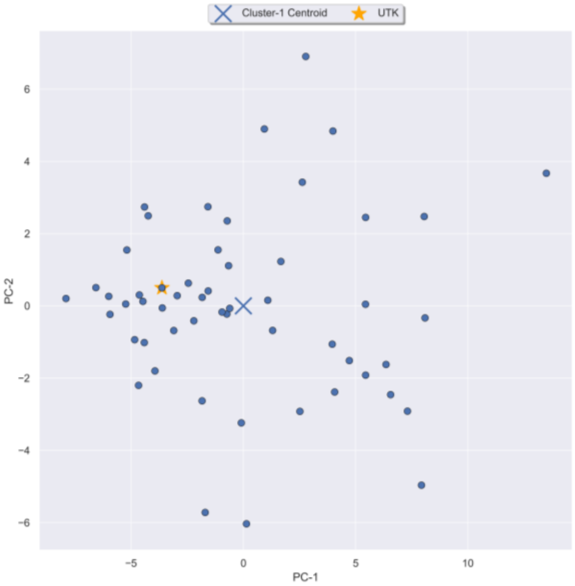

Figure 4: Initial one-cluster testing of data.

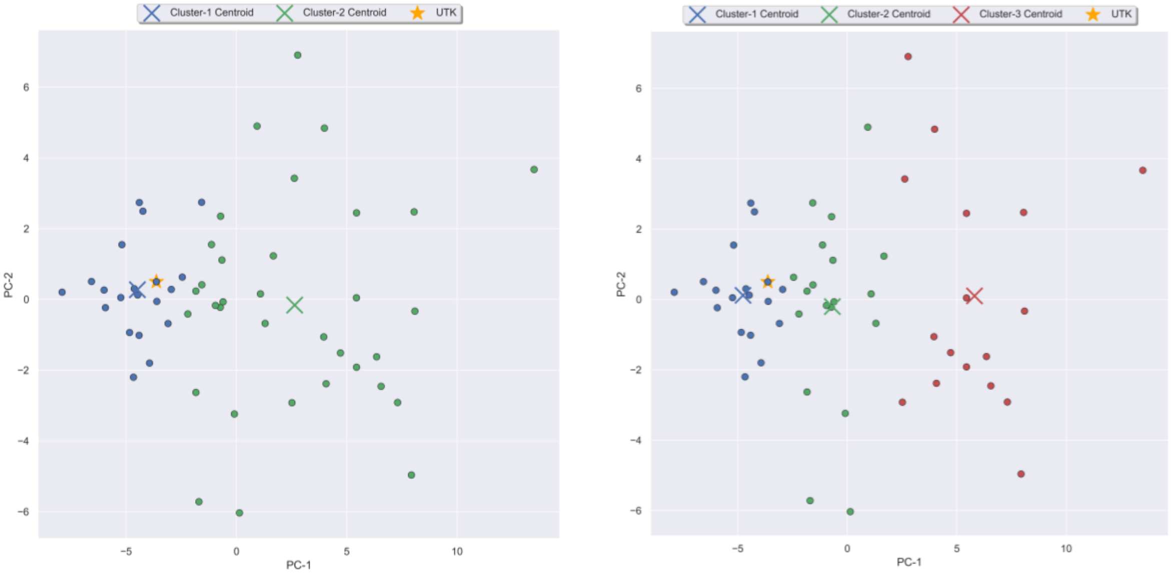

Figure 5: (*Left*) Two clusters and (*Right*) three clusters using k-Means for the first two PCs.

Implementing the created k-Means algorithm for one, two, three, and four clusters on the first two PCs, the algorithm converged in one iteration for one and two clusters, two iterations for three clusters, and six iterations for four clusters. The iterations continued until a maximum of 1,000 were completed or a convergence of 0.01% obtained (i.e. convergence calculated as percentage of total change in centroids between iterations).

The minimal intercluster distance and maximal intracluster distance are utilized as figures of merits for the clustering algorithm. The former calculates the minimal distance between points in different clusters, while the latter calculates the maximal distance between distinct points within a single cluster. The ratio, referred to as the Dunn index, of the minimal intercluster distance to the maximal intracluster distance provides a single value to evaluate the clustering, where generally the greater the ratio the better. For the first two PCs of the data and up to four clusters, the minimal intercluster and maximal intracluster distances, respectively, are provided along with the Dunn Index in the following Table.

Table 3: Figures of Merit for created k-Means on first-two PCs.

k-Clusters	Minimal Intercluster	Maximal Intracluster	Dunn Index
2	7.145	17.862	0.400
3	4.095	12.938	0.317
4	4.125	11.204	0.368

For the two, three, and four k-Means clustering on the first two PCs, the University of Tennessee – Knoxville is grouped in the same (blue) cluster as the following the universities:

- Clemson University, Auburn University, Iowa State University, University of California - Santa Barbara, Louisiana State University, University of Kentucky, Mississippi State University, University of Alabama, University of Arkansas, University of Mississippi, University of South Carolina, University of Oklahoma, University of Nebraska, Colorado State, Univ. of Kansas, U. of Massachusetts, University of Oregon, SUNY - Stony Brook, and University of California-Riverside.

If we choose a higher dimensional data matrix and higher number of clusters, it is expected to observe smaller groupings. To calculate the appropriate number of clusters to use, an *Elbow method* was attempted to be used, as shown in the following Figure with the sum of square difference as a function of the number of clusters. From this analysis, there is not apparent *Elbow* feature, thus this method was not used.

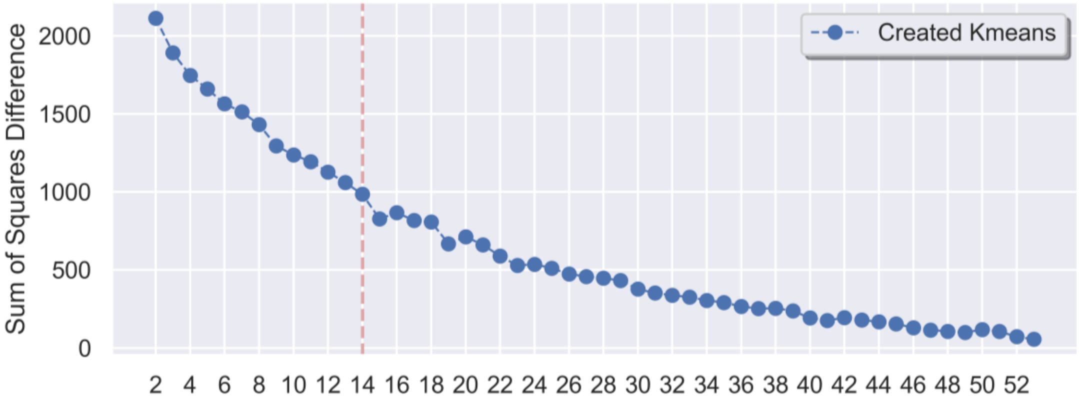

Figure 6: Error and number of clusters for elbow method analysis.

During further analysis, as the number of clusters increased, the number of clusters with single values also increased starting around 9 clusters. To not overfit the data by applying several clusters too large, 8 clusters were used. Recall that for the mean-centered, *z*-standardized data, 15 PCs explained approximately 95% of the total variance. Thus, 15 PCs and 8 clusters are used. The following Figure of the first two PCs shows the results of the nine clusters on the 15 PCs data matrix; same as the prior Figures, where the University of Tennessee – Knoxville is shown with an orange star.

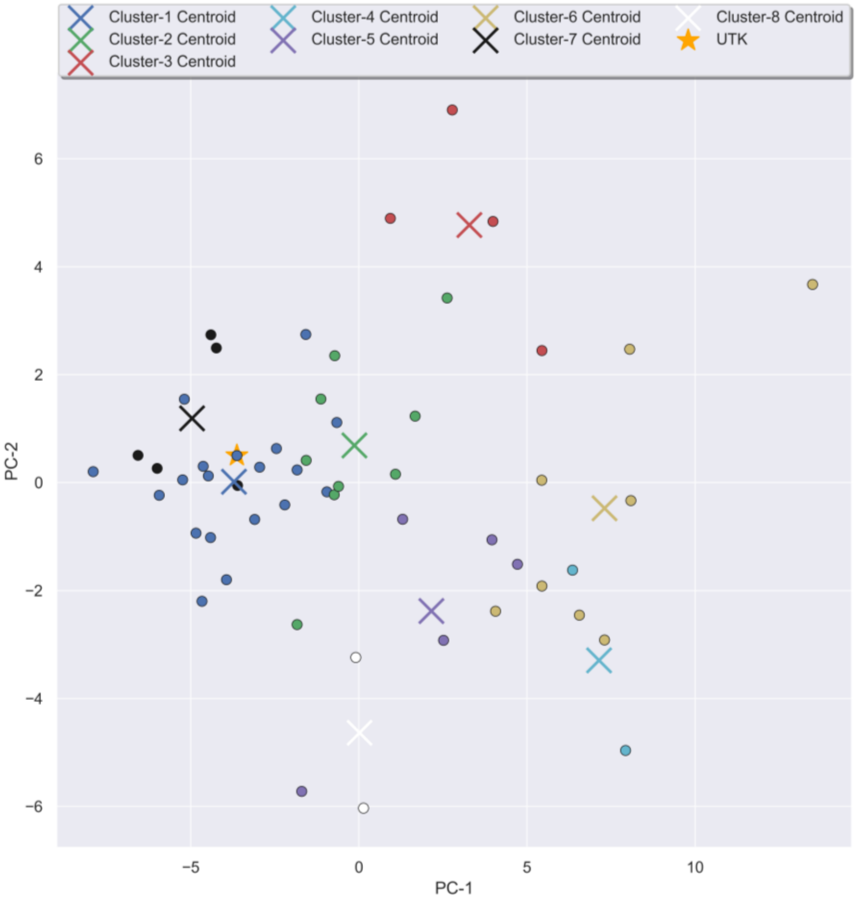

Figure 7: k-Means clustering on first 15 PC data matrix and 8 clusters.

The minimal intercluster and maximal intracluster distances were 4.977 and 14.405, respectively, resulting in a Dunn index of approximately 0.345. There are several other Universities in the same cluster as the University of Tennessee and are as follows:

- University of Tennessee – Knoxville, Auburn University, Iowa State University, Louisiana State University, University of Kentucky, University of Missouri, University of Alabama, University of Arkansas, University of Mississippi, University of South Carolina, University of Oklahoma, University of Nebraska, Colorado State, University of Kansas, SUNY – Stony Brook, University of Iowa, Buffalo University, and University of Utah.

If we only use first two PCs and eight clusters, rather than the 15 PCs chosen, the clustering should change drastically as less dimensions are used to separate the data into clusters. The following Figure provides the results.

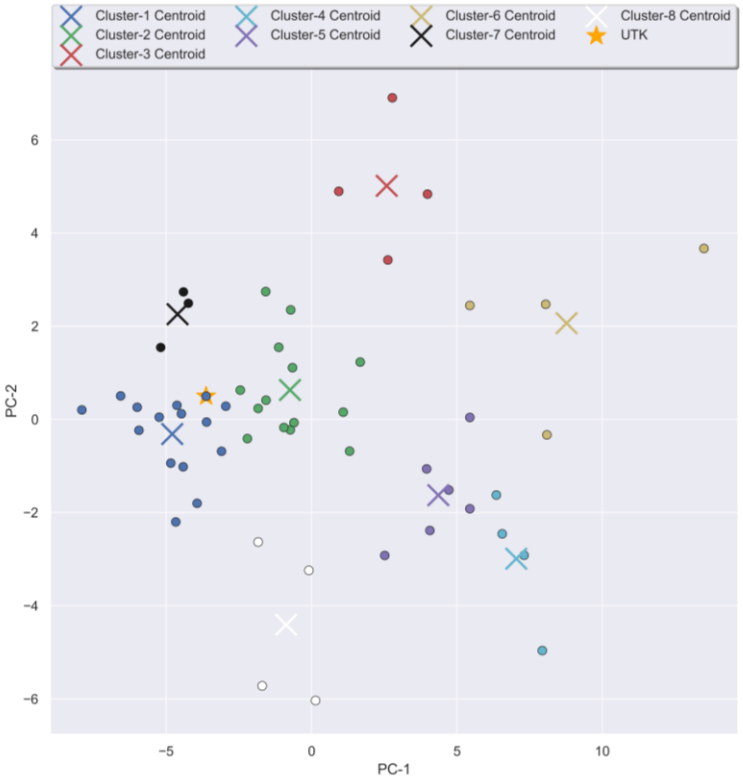

Figure 8: k-Means clustering on first two PCs using eight clusters.

This results in a minimal intercluster and maximal intracluster distances were 2.578 and 8.143, respectively, resulting in a Dunn index of approximately 0.316. The groupings change as well, where the following universities are clustered with the University of Tennessee – Knoxville:

- Auburn University, Iowa State University, Louisiana State University, Mississippi State University, University of Alabama, University of Arkansas, University of South Carolina, University of Oklahoma, University of Nebraska, Colorado State, University of Kansas, University of Massachusetts, University of Oregon, and University of California – Riverside.

The EM method is used on the first two PCs for two, three, and four clusters. This method was used for five iterations or 0.01% convergence, calculated the same as the k-Means approach above. The EM method converged with one iteration for two and three clusters, and five iterations for four clusters. The first two PCs are plotted in the Figures for said number of clusters.

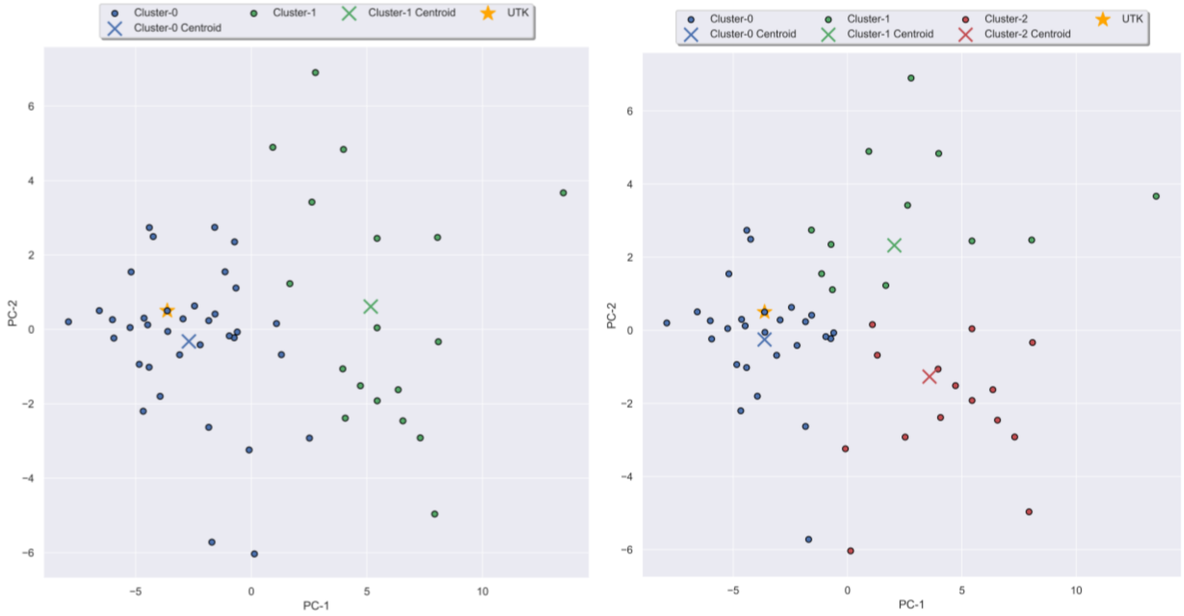

Figure 9: EM clustering method for first two PCs using (*Left*) two and (*Right*) three clusters.

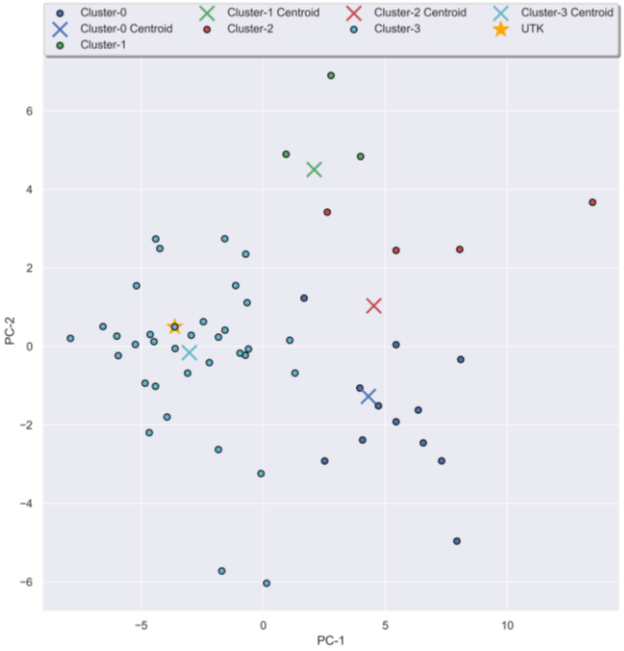

Figure 10: EM clustering method for first two PCs using four clusters.

Comparing the k-Means and the EM results show much similarities between the two clustering approaches, which is expected. The same general trend is equal for both, but the clusters are fit to different data points, which is apparent by the difference in cluster centroids between the k-Means (Figure 5) and EM (Figure 9).