Ronald Randolph

CS 425

Project 2

10/23/2018

Dimensionality Reduction & Clustering

Data Exploration

In this project we are looking at data containing 65 attributes for UTK and 56 other similar universities. These attributes are university statistics such as graduation rate, number of faculty members, and financial allocations. This data was supplied in three file formats (.xls, .xlsx, and .csv). The file I opted to use for the purposes of this project is *UTK-peers.csv*. I chose this file type to streamline the reading and organization of data through use of the *pandas* and *numpy* libraries. Looking at the data, most of the values for the attributes are of integer type. However, for attribute HBC, the value is Boolean. With the exception of two universities (Penn State and Mich. State), the values for the Med and Vet school attributes are either a '*x*' or a blank. Since there are only two values for these attributes, I feel that the values can also be represented as Boolean. Furthermore, the name attribute for the universities is a string and therefore non-numerical. After carefully looking through the data, there are multiple missing values for a few of the attributes. Many of the missing values are integers representing an amount of currency depicting specific university expenditures. As previously mentioned, Pennsylvania State Univ. and Michigan State University have the abnormal value 'pre clin' for their med school attribute.

Data Preparation

To prepare the data for analysis, I first removed all attributes with non-numerical values. Because array indexing will maintain the identity of each university variable, we can remove the attribute containing the names of the universities. The attribute for AG Research has missing values for 20 of the 55 universities. As a result, I removed the attribute because the missing data for that attribute cannot be accurately imputed. Furthermore, I removed the attributes such as the identification number, medical school, and veterinary school because they are not relevant variables in analyzing the data for these universities. Lastly, I removed the ranking attributes – Wall St. Journal Rank and 2017 US News top 65 – because the variables were either incorrect, overlapping, or missing. I cannot effectively impute the missing data for these attributes because rankings are a definite, assigned value and cannot be assumed.

After adjusting and removing attributes, we are left with 57 individual universities and the data of 51 attributes for each university. This amounts to a grand total of 2907 separate

elements of data. This adjusted set of data can be found in the included file *UTK-peers-edit.csv*. Additionally, there were 32 missing values in the given data. Many of the missing variables fell under attributes relating to various university expenditures. After looking over the available data and the accompanying deviation for each of these attributes, I found that the variances throughout these attributes were relatively low. Therefore, I decided that using the mean imputation method would be the best course of action to accurately assign missing values.

	Raw Data	Standardized Data
Avg. Mean	228,051,868	9.497×10^{-12}
Avg. Standard Deviation	202,360,990	1.009
Avg. Minimum	44,814,450	-1.485
Avg. Maximum	1.019×10^{9}	3.078

Figure 1: Average Attribute Statistics

Figure 1 illustrates the calculated statistics for each of the 51 attributes before and after data standardization. After data imputation, each attribute has a total of 57 values. Without data normalization, the average mean and deviation of the given data were too large. Many of the attributes for university expenditures contain large and exceedingly varying numbers. To rectify this issue, I opted to use z-normalization to standardize the given data.

$$z = \frac{x - \mu}{\sigma}$$

μ = Mean
σ = Standard Deviation

After normalization, the average mean and deviation are smaller and more manageable. This will allow for easier analyzing of data while minimalizing the skewering of data trends. The z-normalized data can be found within the included file *UTK-peers-zstd.csv*.

Implementation

For this project, I was given specific steps to follow in analyzing the given data. These directions were broken up into "part 1" and "part 2". Part 1 of this project is focused on using principal components analysis (PCA) for data visualization. My implementation for this section can be found in the included file *proj2.py*. This program opens *UTK-peers-zstd.csv* – a normalized set of the given data for this project. The program then reads the data into a matrix and factors it. From this, I am able to extract a list of the singular values. Using the singular values, the program then plots the corresponding scree graph of the data (figure 2).

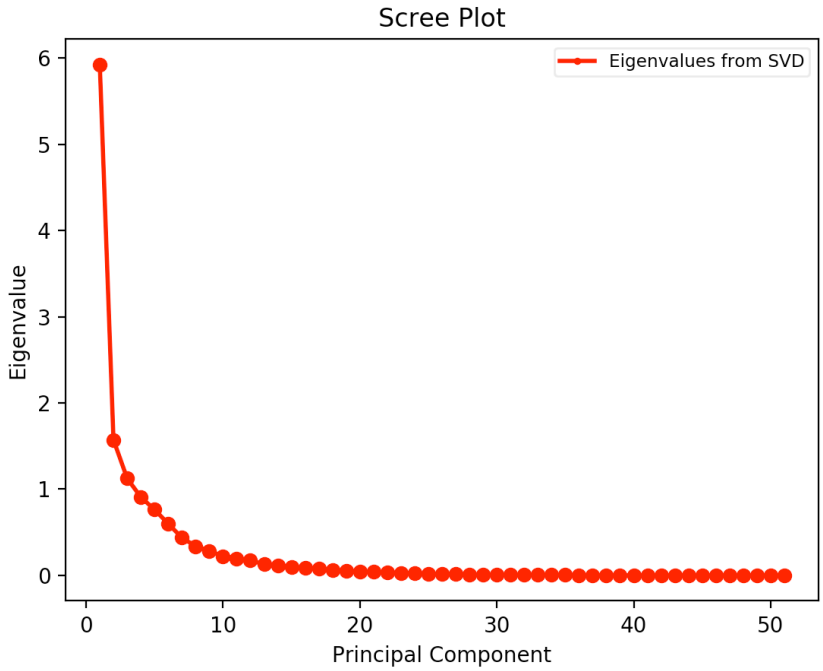

Figure 2: Scree Plot of Data

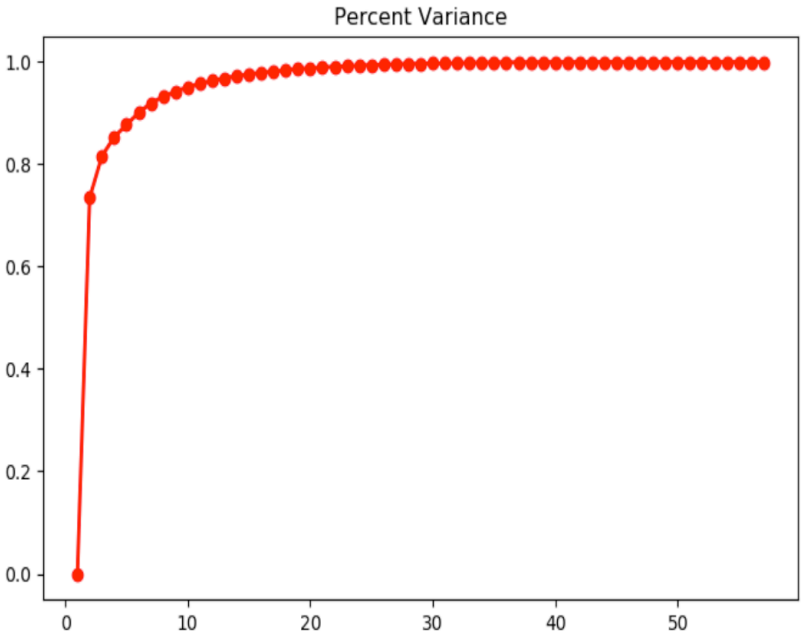

Figure 3: Graph of Percentage of Variance (PoV)

Additionally, using the extracted singular values, the program then plots the percentage of variance covered by the first *k* singular values vs. *k*. Looking at the produced charts above, the

best choice of *k* would be 3. Next, the program effectively reduces the matrix to the first *k* PCs. The program achieves this by taking the first k columns of the matrix V returned from the SVD calculation. Figure 4 below is a scatter plot of the first two PCs. The points have been annotated with numbers to map to each university in the given data.

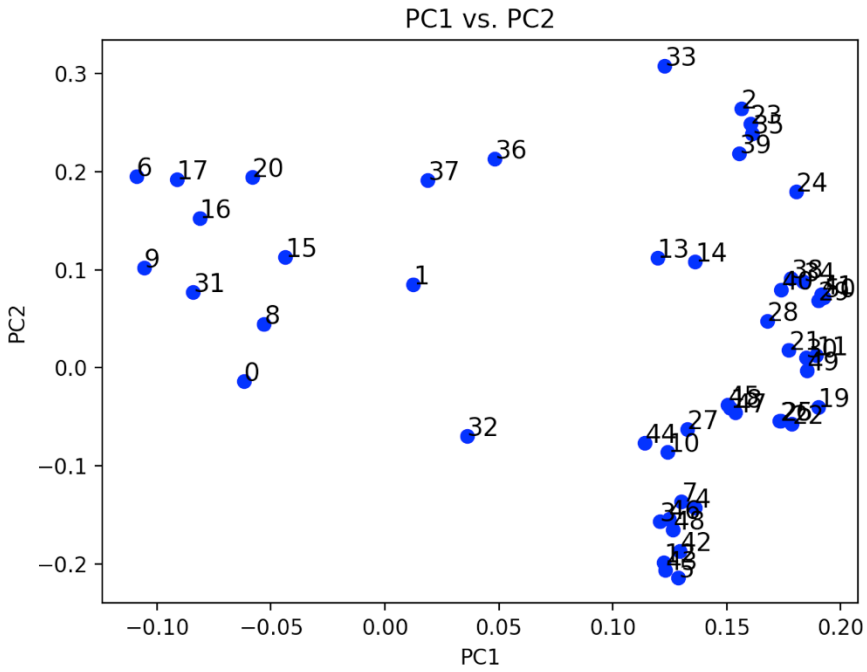

Figure 4: Scatter plot of PC1 vs. PC2

For the second part of the project, we are tasked with implementing a k-means clustering program and apply It to the original given data. The file *k-cluster.py* contains my implementation of the k-means clustering method. Using the original data matrix and 3 as our value for *k*, the program required only 5 iterations to reach absolute convergence. This was probably due to the small size of data we are analyzing. The table below (figure 5) reports the figures of merit for the clustering. To calculate these, I used the distance formula to find the distances between points within the clusters. Through experimentation, I found the most affective value for k to be 3. The 3-cluster graph is shown below (figure 6). Other universities in the same cluster as UTK include Michigan State University, Auburn University, University of Kentucky, and Clemson University.

	Values
Min. Intercluster Distance	0.97009 (green)
Max. Intracluster Distance	19.6752 (red)
Dunn Index	0.0493

Figure 5: k-Means Clustering Stats

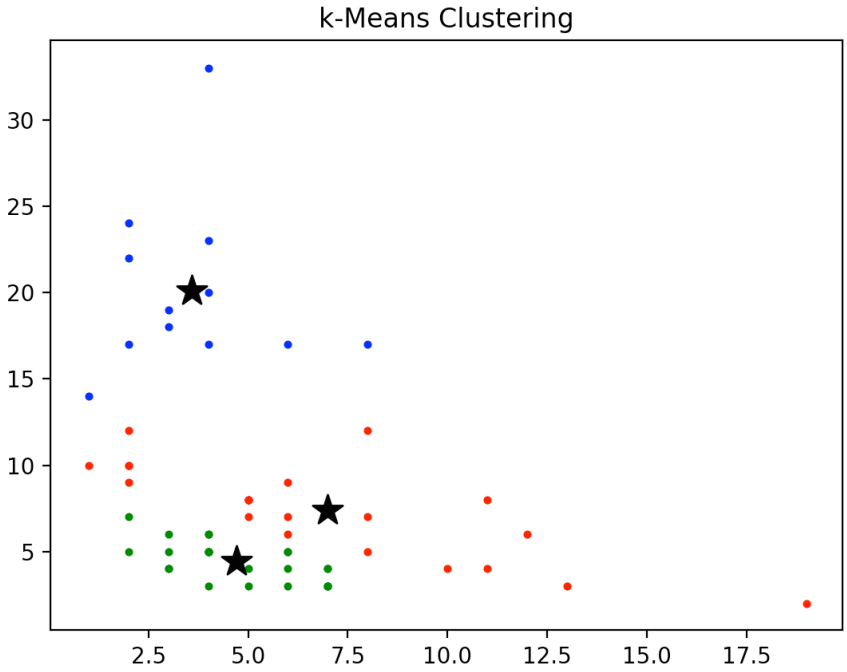

Figure 6: *k*-Means Clustering Scatter Plot

Finally, in the second part of the project 2 prompt, we are asked to run our implementation of the *k*-means clustering algorithm twice more – First, on the data matrix that reflects the number of PCs I opted to use in part 1. Secondly, we are to run the same algorithm using the first two PCs like in part 1. On the first run using the first 3 PCs, I find that using 5 clusters adequately covers the entirety of the data. Furthermore, when comparing the figures of merit for both runs, I find that this second run with 5 clusters has a higher Dunn Index. Universities sharing a cluster with UTK include University of South Carolina and Colorado State. On the final run, the data being cluster is that of the first two PCs. Once again, I extracted these two PCs by slicing the first two columns from the return 'V' matrix. On this final run, I found that 4 clusters cover the data most effectively. Once again, this run's overall Dunn index is much higher than that of the first run. This most likely is due to the less affective spread of the 3 clusters on the first run. Some of the universities in the same cluster of UT are Rutgers University, Clemson University, University of Illinois, and University of California – Santa Barbara.

Often times, machine learning and data science requires the use of exceedingly large amounts of data. Using principal components is an extremely useful way to help visualize data. In addition to making the data easier to understand and analyze, PCA's help the user to find the axes that account for the most variance within the data. This information provided through the use of PCA can be used to find any relations within the population of data.