Rehan Rupawalla rmr3723
Mihir Gupta mg59824

Data Analytics Project: Prosthetic Demand

Step One

1. What is your data set about?

The dataset in use contains information about various health and accident-related proportions per state in the United States. It includes data on the proportion of accidents for the years 2018, 2019, and 2020, the proportion of current smokers, the proportion of veterans, the proportion of individuals with diabetes, and the proportion of individuals with cancer. Notably, these are all predictors for upper-limb amputations rates in the U.S. based on research.

2. Clean up your data and reduce it to no more than 2000 observations if your data set is very large.

See Jupyter Notebook

3. What is exactly your research question? What do you want to learn from data? What is your learning model ,e.g., a Classification, Clustering, etc?

The research question here surrounds "Can we group states into clusters based on their health and accident-related predictors for lower-limb amputations? If so, what characteristics define each cluster?" This is a clustering problem, and we have used the KMeans algorithm as our learning model to solve it. From the data, we want to learn which groups of states can be analyzed together and which cannot from a prosthetic demand point of view. We also want to identify any similarities that make it easier to analyze prosthetic trends in the U.S.

4. What are your current expectations about the results?

My expectation is that states with similar health and accident proportions will be grouped together. These clusters might reflect regional trends or similarities in state healthcare policies or population demographics. For example, states with higher smoking proportions might also have higher cancer proportions. In addition, my expectation is that these clusters/trends will also follow income lines.

5. How do you want to evaluate your project? How to access the correctness of your model? How well would you expect that the model will work?

We will perform KMeansClustering and PCA (Principal Component Analysis) Dimension reduction to assess our project.

We will use metrics such as the inertia and silhouette score for KMeansClustering and Explained Variance Ratios to determine the correctness of our model. I expect that the model will work to explain about 75% of the variance in the data because we are dealing with real-world data with many different factors and components, so there is a chance that there is a higher-than-expected error.

Rehan Rupawalla rmr3723
Mihir Gupta mg59824

Step Two

1. Project Implementation

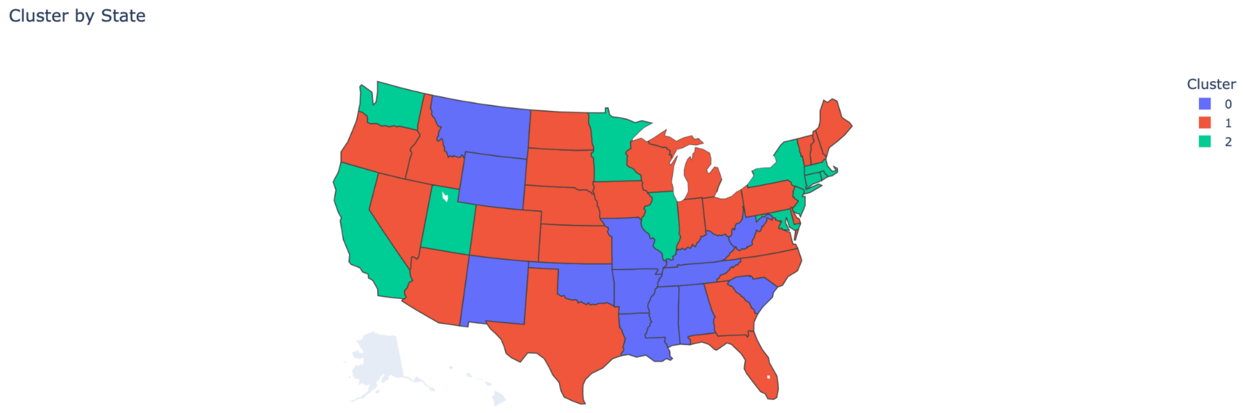
You need to implement your project in python. You are allowed to use any Machine Learning Library like scikit-learn1. You are also allowed to implement your project without using any libraries.

- You can use Jupyter Notebooks to implement your project and provide your documentations.
- Your code should be completed and be compilable without any errors. We should be able to read your documents, and be able to run your project.
- Run your implementation (on your Laptop or on Cluster if data is large) and generate the results.

See Jupyter Notebook

2. Provide the interpretations of your results.

Our results yielded 3 clusters of states that behave similarly when it comes to prosthetic demand and prosthetic supply and we plotted them on the U.S Map:



Reflecting on the inertia values computed for KMeans clusters ranging from 1 to 9, here's the data:

1 cluster: 350.0
2 clusters: 210.13837142534481
3 clusters: 163.75515443683804
4 clusters: 129.1963433576945
5 clusters: 107.39526517073608

Rehan Rupawalla rmr3723
Mihir Gupta mg59824

6 clusters: 90.79515876011035
7 clusters: 79.32467746306986
8 clusters: 72.64204250817832
9 clusters: 66.2244181842907

These values demonstrate how inertia decreases as the number of clusters increases. This is anticipated, as the more clusters we have, the closer each point is likely to be to the center of its cluster. However, lower inertia doesn't always equate to a better model as overfitting becomes a concern with an excessive number of clusters.

Moving onto the silhouette score of 0.273, it suggests to me that the clustering configuration is passably good, but with some potential overlap amongst some clusters. Given the score's range from -1 to 1, it can be seen as a measure of the compactness of the clusters. Therefore, to improve upon this, adjustments in parameters or preprocessing steps may be necessary.

Now considering the PCA results, the explained variance ratio of [0.57818803, 0.15820049] indicates that the first and second principal components account for 57.82% and 15.82% of the variance in the data, respectively. These two components collectively explain about 73.6% of the data's variance. Notably, there's a large decline in the explained variance from the first to the second component, suggesting the first component holds a substantial amount of the data's structure. Nevertheless, whether this amount of explained variance is adequate or not hinges on the specific objectives of the project. If it's vital to maintain more information, we might need to employ more components.

3. What can you do to improve your results? Apply your ideas to improve your results.

In order to optimize my results with KMeans clustering and PCA, we identified and executed several iterative steps and experiments.

For KMeans Clustering:

Determining the Optimal Number of Clusters: we noticed that the inertia values consistently decreased as the number of clusters increased, a standard expectation. However, to avoid arbitrarily increasing the number of clusters, we employed the Elbow method. This strategy involves plotting the explained variation as a function of the number of clusters, and then selecting the "elbow" of the curve as the optimal number of clusters.

Scaling the Features: Acknowledging that KMeans clustering is sensitive to the scale of data, we decided to standardize the features. Such an action ensures that all features are on the same scale, thereby improving the performance of the algorithm. (we had already done this the first time around, just to ensure the results were the most accurate from the beginning)

Rehan Rupawalla rmr3723
Mihir Gupta mg59824

Experimenting with Different Initialization Methods: Given that KMeans is sensitive to the initial placement of centroids, we leveraged the "k-means++" initialization method available in Scikit-learn. This method can often lead to enhanced results.

For PCA:

Scaling the Features: Similar to KMeans, PCA can also be affected by the scale of features. To prevent high-scale features from dominating the first principal component, we standardized the features before applying PCA.

Including More Components: To retain more information, we considered adding more components. I kept in mind the trade-off between complexity and potential overfitting while doing so.

Interpretation of the new results:

PCA:

Explained variance ratio: The explained variance ratio is now [0.57818803, 0.15820049, 0.12847184, 0.08891643]. This means the first principal component accounts for 57.82% of the variance in the data, the second for 15.82%, the third for 12.85%, and the fourth for 8.89%. In total, these four components explain about 89.5% of the variance in the data. The addition of the third and fourth components in my PCA model has clearly increased the amount of variance captured compared to the previous two-component model.

KMeans:

Inertia: The inertia is now 111.57, which is lower than the previous inertia with fewer clusters (163.51). This suggests that the clusters are tighter, meaning that the data points within each cluster are closer to their respective centroids.

Silhouette Score: The silhouette score has increased from 0.273 to 0.341. This indicates that the clusters are more defined than before and there's less overlap between them. While it's still not close to 1, this improvement in the silhouette score suggests that the changes we made to the parameters and preprocessing steps have improved the quality of the clusters.

4. Provide any references that you have used

Data Sources:
https://www.cdc.gov/nchs/nhis/ad292tb1.htm

https://static.nhtsa.gov/nhtsa/downloads/FARS/FileList.pdf

https://u.osu.edu/wheelbarrows/upper-limb-injury-statistics/

Rehan Rupawalla rmr3723
Mihir Gupta mg59824

https://www.va.gov/vetdata/veteran_population.asp

https://www.lung.org/research/trends-in-lung-disease/tobacco-trends-brief/data-tables/ad-cig-smoking-state

https://gis.cdc.gov/grasp/diabetes/diabetesatlas-surveillance.html#

Step Three

1. Create Recorded Presentation Video of your Project

- Create a document or a presentation (You can create a presentation using your Jupyter notebook, or create a powerpoint presentation, or other formats) to describe your project and results
- Describe your code.
- Describe the results of your project in a professional way.
- You may want to visualization diagrams and describe the results based on some diagrams - but having diagrams is not a MUST have to get the full credit.
- Describe the model and results of your project in a way that every person in this field can read, enjoy and understand.

See Video

2. Code Outline

Code to setup DF
Demand Graph
Demand Graph + Cancer Scores
Supply Graph
Opportunity Graph
KMeans Cluster States
PCA Clustering
Scores of PCA and KMeans Models
Heatmap of States
Map of State and Cluster
Correlations