COGS 118B Project: Identifying the Severity of a Heart Disease

Tiffany Streitenberger, Shania Ie, Dorine Ernst, Cory Smith, Shadman Noor, Shahab Banki

Abstract

Studies show that heart disease is the ruling cause of death for both men and women in the United States. Using the Heart Disease Dataset retrieved from the UCI Machine Learning Repository, we explored whether specific attributes (age, trestbps, chol, thalch, oldpeak) show signs that the patient has heart disease. To analyze this, we used the algorithms, K-means and Principal Component Analysis (PCA) along with factoring in pre-PCA and post-PCA.

1. Introduction and Motivation

Heart disease is one of the deadliest diseases in the United States, affecting most men, women and people of most racial and ethnic groups. Statistically, a person in the United States dies every 36 seconds from a type of heart disease, which totals up to around 655,000 Americans. ("Heart Disease Facts", 2020) With the prevalence of heart disease, we decided to conduct an analysis to see which factors contribute to the disease. This is important as identifying contributing factors could improve preventive care and cut medical cost.

We are trying to predict the severity of a heart disease through the accumulated vital organ information. We choose to cluster the data using unsupervised machine learning algorithms such as K-means and reduce dimensionality of the data using Principal Component Analysis (PCA). We will analyze the accuracy of the K-Means model before and after dimensionality reduction. We decided to use K-Means as it guarantees convergence, it adapts to new samples, is simple to implement, and generalizes the cluster shapes. We used PCA to perform feature selection in order to reduce the dimensionality of the data. This is helpful when dealing with features that have multicollinearity and projects all data points into a lower-dimensional subspace.

2. Related Work

In the Principal Component Analysis (PCA) lecture, we learned that PCA finds a linear subspace that contains most of the variance of the data. PCA is helpful when analyzing large dimensional datasets by manipulating its eigenvectors. We walked through the face reconstruction example in class where we have around 12,000 dimensions and we used PCA to decrease the dimensionality by taking 97 eigenfaces to reconstruct the face. In our project, we decrease the dimensional of our data from 5D to 2D.

In a research paper by Gajbhiye et al., "Diagnosis Of Heart Diseases Using K-means Clustering and Bell Curve Fitting", they proposed a solution to analyze patterns that might arise when identifying cardiovascular diseases. They used K-Means clustering to differentiate the percentage of healthy and sick individuals based on the vital organ information. Unlike our project, they used a total of 12 attributes and included the categorical variables as part of their K-Means analysis. This includes features such as gender, chest pain type, fasting blood sugar, etc.

In a similar paper, *"Heart Disease Prediction using Exploratory Data Analysis"*, Indrakumaria and their team focus on the prediction of heart disease using Exploratory Data Analysis (EDA). Similar to our project, they used K-means clustering to predict heart disease. Where we differed was that we used different datasets and attributes. The 8 attributes they used were age, chest pain type, blood pressure, blood glucose level, ECG in rest, heart rate and four types of chest pain. We used 5 attributes (which we will cover in the Dataset section).

3. Methods

To cluster the data, we will use 2 algorithms which are K-means and principal component analysis(PCA). K-means is a simple but effective algorithm where we will partition the data into K clusters and we will update

it using an iterative batch algorithm. PCA is the simplest method where we will find a linear subspace with the greatest projected variance, which is useful to find a reduced subspace that contains most of the variance.

3.1. Dataset

In this analysis, we are using the heart disease dataset provided by the UCI repository. This dataset contains 16 total attributes, however after filtering through we are only using the 5 numerical attributes and a target label, which identifies the diagnosis of the heart disease. The **5 selected attributes** are age (expressed in years), trestbps: resting blood pressure (measured in mm Hg during hospital admission), chol: serum cholesterol (mg/dl), thalch: measures the maximum heart rate achieved, oldpeak: measures abnormality from exercise relative to resting heart rate. The target label (num) is the stages of heart disease with the value 0 indicating no heart disease and 1-4 being the four stages of heart disease, 4 being the extreme.

3.2. K-means (Pre-PCA)

K-means is commonly used to find clusters that are not specifically labelled. We will be using K-means to cluster the stages of heart disease. First we will assign our five numerical attributes (age, trestbps, chol, thalch, oldpeak) to X and our target label (num) to y. In our case, we chose K = 5 to correspond to each numerical attribute.

The K-Means Clustering Algorithm works in the following way:

1. Start with random initialization of 5 cluster centers
2. Calculate the squared distance of each point to each cluster mean
3. Assign the nearest mean to each datapoint in a responsibility matrix, setting each index to 1 if point n is closest to cluster k, and 0 otherwise
4. Recalculate cluster centers from mean of data points based on different cluster assignments from columns of responsibility matrix
5. Repeat steps 1 to 4 until cluster centers have converged below some chosen threshold

We defined the functions *calcSqDistances, determineRnk,* and *recalcMus* to implement steps 2,3, and 4 listed above, with functions *determineRnk* and *recalcMus* provided in Homework 2 solutions. The results of running the K-Means Clustering Algorithm is usually best shown through a visualization, but since we had 5 dimensions, it was difficult to provide a visual. Instead, once we ran the K-Means Clustering Algorithm on X, we created a new matrix (count) to tell us which cluster was associated to which target label (num).
For count, the rows are for the different possible values of num (0-4) and columns are associated to the 5 final cluster centers determined by K-means. We initialized our matrix count to be 5x5 and filled with zeroes. Then we used a nested for loop to run through each value in our final responsibility matrix, as mentioned above, and if the index equaled 1, we would add one to the associated count index. From our final count matrix, we were able to somewhat determine how accurate K-means was based on distribution of numbers.

Clusters (k=5)

num

```
0 [151. 119.  18.  19.  64.]
1 [ 42.  45.  55.  42.  60.]
2 [  5.  15.  37.  24.  18.]
3 [  5.   4.  32.  28.  20.]
4 [  0.   2.   6.  14.   2.]
```

3.3. Principal Component Analysis (PCA)

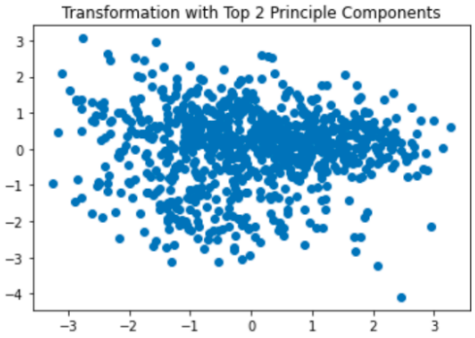
Principal Component Analysis is done by finding the directions of the greatest project variance on zero mean data, which are found by ordering the eigenvectors from the covariance matrix of the mean subtracted data.
Dimensionality reduction through principal component analysis is done by first finding the covariance matrix of the mean subtracted data. Since our dataset has only 5 quantitative data and many more data points there is no need to perform the transpose trick as our covariance matrix is only 5 by 5. Take Z to be our mean subtracted data, by calculating the mean of all data points and subtracting it from each one to get the difference. The covariance matrix is found by calculating:

$$\frac{1}{n}(ZZ^T) \tag{1}$$

Next, np.linalg.eig(covmatrix) can be used to find the eigenvalues and vectors of this covariance matrix(covmatrix). We next will need to sort the eigenvectors by size of corresponding eigenvalue in which we use the function provided in Homework 4 to assist us. With these sorted eigenvectors we now

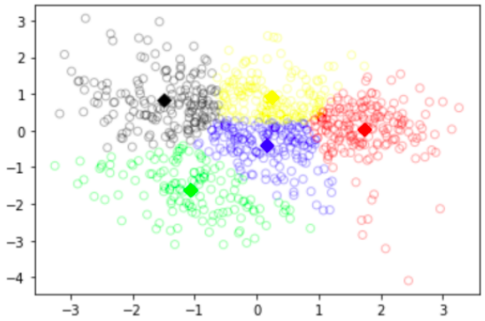
have the principal components in order of importance, the top eigenvector corresponding to the axis of most variance. We can now multiply the sorted eigenvectors by the mean subtracted data to obtain the transformed data. Any number of eigenvectors can be used to dimensionally reduce and transform the data using either all or some of the important principal components, determined by largest variance/largest eigenvalues.

Performing PCA, however, did not lead to data that could be more easily separated into clusters. The figure below shows the data after dimensionally transformed with the top two principal components. It can be seen that even along the dimensions of most variance we still cannot make separations between the clusters for the 5 stages of heart disease.



3.4. K-means (Post-PCA)

After taking our top 2 principal components to be our new features for the data set we wanted to try to run K-means using this new reduced dimensional data set. It had only 2 features but the features where the lines of most variance. So we wanted to try to obtain a similar accuracy as regular K-means with a quicker runtime of using 2-dimensional as opposed to 5-dimensional data for the K-means. Then, by tweaking the *calcSqDistances* function to account for only 2 dimensions for our transformed data and using the function *plotCurrent* provided in Homework 2, we are able to rerun K-Means algorithm and plot the top 2 principal components of X. The final iteration of K-means is shown below. As we can see there were never clearly defined clusters to begin with so K-means might be defining clusters arbitrarily.



4. Results

We discovered that K-means could not accurately predict the stage of heart disease of a patient from the 5 quantitative variables we obtained from the data set. The clusters seemed to struggle to distinguish the different stages from each other, with some clusters containing multiple stages, and others containing barely any. After performing a transformation with Principal Component Analysis we were able to reduce the dimensionality of the data from 5 to 2, however these two dimensions of most variance still did not allow for accurate classification of the 5 stages.

Perhaps in predicting heart disease the categorical variables provide more information. As K-means does not perform well with one-hot encoded data we had to exclude the majority of the data's features. As a result we could have been removing important information for making these classifications. Perhaps another algorithm could have been a better model for making these distinctions. The graph of our normalized data with just the 5 quantitative variables did not have obvious clusters just by sight which lead to K-means also struggling to find such groups. Without clear separation of these clusters K-means will find clusters almost arbitrarily as it is heavily dependent on its random initialization.

5. Discussion

After applying the K-Means algorithm to the dataset, we learnt that K-Means did not do a very good job in clustering our dataset accurately. When calculating the accuracy scores of the pre-PCA K-Means applied data and post-PCA K-Means applied data, we found out that the accuracy for the pre-PCA K-Means applied data did very poorly when it came to clustering the data correctly, with only an accuracy of around 58%. However, after applying PCA and then re-applying K-Means clustering to the data, the accuracy score rose a bit higher, resulting in a score of around 77% but still not very accurate. In terms of the run-time, we found that the post-PCA data ran faster than the raw data during the application

of K-Means. Although different iterations of the above produced slightly different values and results, overall, we felt that the K-Means algorithm wasn't the best fit for this particular dataset that we chose, and better clustering algorithms could have been used instead to cluster our Heart Disease dataset.

Through our implementations, we learned that depending on how the data is spread in the dataset, a different clustering method would have been a better fit. With the use of graphs and visualization, we could have spent more time to make sure if K-Means would really have been the best fit for our dataset. More time could have also prompted us to opt for a different, more suitable dataset or used different clustering algorithms instead. With more time, we could have tested other algorithms to see if we would have gotten better results with this dataset.

Additionally, when we visualize the datasets fit the first time it looks like a blop. It is hard to distinguish with the human eye, so it might be the reason our algorithm failed to work. Comparing the accuracy of different algorithms could have given us a better understanding of which kind of clustering algorithms work best with what type of datasets. We believe that because of the exclusion of many of the one-hot-encoded features of the dataset, since K-Means doesn't work well with them, we lost important categorical variables which could have been a better fit in predicting heart disease. If everything had worked perfectly, we would have liked to apply even more clustering algorithms to the dataset and see how the data could be clustered with them, through comparison. Which is why we would suggest experimenting with other clustering algorithms to see how they perform on our dataset.

The two extensions that we would propose for our project would be trying the Gaussian Mixture Models algorithm and also seeing how both Gaussian Mixture Models and K-Means work with the categorical variables that we had removed from the dataset. With the way the data is spread out, we think that Gaussian Mixture Models could potentially model the data more efficiently than K-Means. And since we excluded some one-hot-encoded categorical variables and lost some data, we could possibly apply Gaussian Mixture Models without removing those variables to see how the algorithm performs as well.

Since we cannot use K-Means to cluster based on categorical features, another method would be to use K-Modes. K-Means calculates the euclidean distance with respect to the cluster centroid, hence in the case of categorical features where the data is discrete and does not have any natural origin, the computation would not be meaningful. K-Modes, on the other hand, uses a simple matching dissimilarity measure and modes as the cluster centroid. In each iteration, the centroid is updated through finding categorical values with the most frequency. This method guarantees that the cluster converges to a local minimal result and minimizes the sum of the distances relative to the centroids.

6. Contribution

- **Shadman Noor** - Tasked with and completed the 'Comparative Data Analysis' and 'Conclusion' sections of the code, and 'Discussion' section of the write-up and presentation.
- **Shania Ie** - Tasked with Data Cleaning (which included getting rid of null values, selecting the appropriate variables, and standardization of the data), 'Related Works', and 'Introduction', part of the 'Methods' and 'Discussion' section of the write-up and presentation.
- **Dorine Ernst** - Tasked with 'Overview' 'Introduction' and 'Conclusion', and 'Related Works; and part of 'Methods' section of the write up and presentation
- **Cory Smith** - Tasked with 'Methods': implementing K-Means Clustering Algorithm pre and post PCA, helped clean and organize code, final paper editing
- **Shahab Banki** - Tasked with 'Methods': performing Principal component analysis on the data and helping to analyze our results and determine what went wrong.
- **Tiffany Streitenberger** - Tasked with 'Data Cleaning' assistance, 'Related Works', group communication, and final production of research paper into LaTeX format, video compilation.

7. Code

`https://github.com/shaniaie/COGS_118B_PROJECT`

8. References

- Centers for Disease Control and Prevention. (2020, September 8). *Heart Disease Facts. Centers for Disease Control and Prevention.* `https://www.cdc.gov/heartdisease/facts.htm`.
- Gajbhiye, S., Natasha (2016). *Diagnosis Of Heart Diseases Using K-means Clustering and*

Bell Curve Fitting. IRJET. https://www.irjet.net/archives/V3/i6/IRJET-V3I6388.pdf

- Google. (n.d.). *k-Means Advantages and Disadvantages — Clustering in Machine Learning.* Google. https://developers.google.com/machine-learning/clustering/algorithm/advantages-disadvantages.
- Indrakumaria, R., Poongodi, T., Jena, S. (2020). *Heart Disease Prediction using Exploratory Data Analysis..* ScienceDirect. https://towardsdatascience.com/inferential-statistics-series-t-test-using-numpy-2718f8f9bf2f
- Janosi, A., Steinbrunn, W., Pfisterer, M., Detrano, R. (1988, July 1). UCI Machine Learning Repository: Heart Disease Data Set. UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/Heart+Disease/
- Kumar, S. (2021, May 22). *Clustering algorithm for data with mixed categorical and Numerical FEATURES.* https://towardsdatascience.com/clustering-algorithm-for-data-with-mixed-categorical-and-numerical-features-d4e3a48066a0.
- Loukas, S. (2020, December 26). *PCA Clearly Explained -When, Why, How To Use It and Feature Importance: A Guide in Python.* Medium. https://pub.towardsai.net/pca-clearly-explained-when-why-how-to-use-it-and-feature-importance-a-guide-in-python-56b3da72d9d1.
- Saini, B. (2021, February 12). *The Most Common Clustering Algorithm for Data Science and Their Code.* Medium. https://medium.com/swlh/the-most-common-clustering-algorithm-for-data-science-and-their-code-39dd4224a480.