Assignment #3

Intro to Data Mining

Usman Ghani Bawany -22850

1. **Kaggle Participation**

As seen in the following pictures, the entries were submitted 70 times. Initially the values were quite low (except for one outlier) the accuracy was not very good, however as the entries progressed, the trend continued to increase as I learned new methods and applied new filters and learned from my mistakes.

This eventually led me to get my best entry at 0.84145.

Graphical user interface, text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

1. **Report**

**CRISP-DM**

* **Business understanding**: The business is from the medical domain that diagnoses the diseases based on the symptoms the patients show.
* **Data understanding:** There are a lot of columns for symptoms for each patient in binary format on which eventually the disease will be diagnosed. The data consists of 133 columns and 2384 unique rows. There are no missing values and there are no duplicate entries. Each column shows a symptom except the first which is for the rowid. A picture containing chart

  Description automatically generated
* **Data Preparation**: In this data as this is binary, there was no normalizing as well as encoding for categorical values and no missing values. So data preparation was not needed a lot. The two main data preparation done were to remove the zero variance columns and remove highly corelated columns. Also removed low variance columns but that did not work out good for the accuracy. (picture attached)

Text

Description automatically generated

* **Modelling**: There were two major models that were implemented for this assignment. As this was a clustering-based dataset, so I had applied Kmean and Agglomerative clustering. The first half entries were Kmeans and then after that the most work was done on Agglomerative clustering. For kmeans a number of tools and features were used.
  + **Elbow Curve**: The elbow curve was used for checking the best value for the numbers of clusters that were to be used. I used 2 ways to make the elbow curve, one was through calculating distortion and the second one was through intertia. (pictures attached) However it was not of much use and the best value did not indeed come through there. The graph showed no straightforward answer so not a very useful tool for this dataset.

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

* + **Silhouette Coefficient and graphs:** This was also used a lot to determine if the clusters were at the right positions or not. In addition it also was used to check the highest value I got from which cluster. However even for this tool even if I used a model with a high silhouette coefficient, the accuracy was not directly proportional to it. It helped a lot to understand the model but I could not estimate the accuracy based on the value of the silhouette coefficient.(pictures attached)

Graphical user interface, application

Description automatically generated

Chart

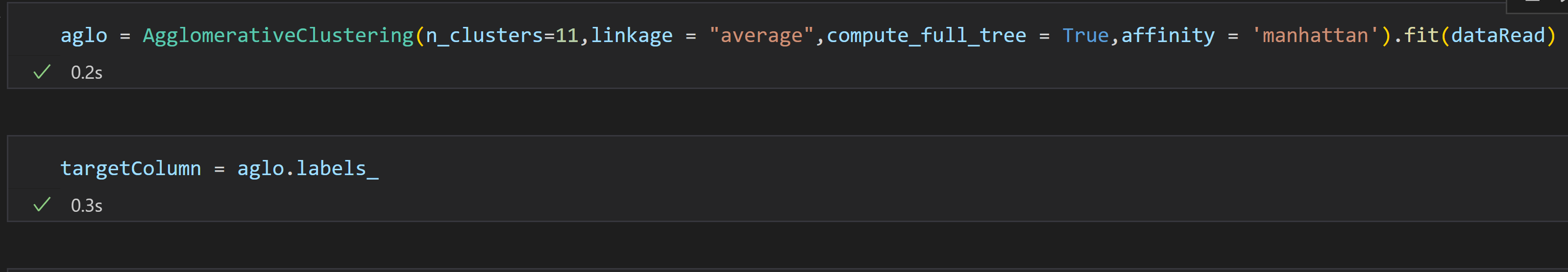
Description automatically generated

* **Agglomerative Clustering:** When I felt that I had done half my entries through kmeans and had reached a ceiling in terms of accuracy so I decide to change the model. Agglomerative is a hierarchical modeling method so its most popular tool is the dendrogram.
  + **Dendrogram**: So in order to understand the dataset and the clusters, countless dendrograms were made. Using these diagrams I got close to the best value and made me increase my knowledge base on trees.(picture attached)

A screenshot of a computer

Description automatically generated with low confidence

* **Evaluation**: The best overall score was achieved through agglomerative clustering as well as some data preprocessing. The data preprocessing done was that the highly corelated features were removed whose value was more than 0.95 and all the zero variance columns were removed.For the agglomerative model the features I used were as follows.



At this implementation I scored my best accuracy of 0.84145

* **Deployment**: at this stage not applicable but InshaAllah for future projects will be doing.

Which algorithm worked best for the given dataset and why?

* The agglomerative clustering performed the best. I tried all sorts of feature selection as well as data preprocessing with both the models but the agglomerative turned out the best in giving the max accuracy score. Using tools such as dendrogram also helped in fine tuning the features

What is the optimal number of clusters in the data as per your findings and why?

* As per my findings and test and trial, the best cluster number was 11. The numbers 12 and 6 were also giving high score but 11 was the best. I believe it was due to the differences in data segregation and how the model divided the data into clusters.

What were the overall challenges that you faced while improving the score, and so on?

* I tried a lot of things and the score had got stuck and I wasn’t getting any progress so was disheartened initially but eventually with a lot of effort and using methods and tools as well as tweaking features ultimately got me a decent score.

Some features used and played with?

* KMeans, KMeans++, Linkage, Compute full tree, random state, number of clusters, algorithm= elkan, correlation, covariance, zero variance, dropping columns, Agglomerative, Affinity, manually changing csv etc

**Summary Of Entries**

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Data preprocessing | Model Details | Score |
| 1 | none | Kmeans tried with 5 clusters | 0.55 |
| 2 | none | Kmeans tried with 5 clusters | 0.55 |
| 3 | none | Kmeans 6 cluster and random state 0 | 0.8009 |
| 4 | none | Kmeans tried with 8 clusters | 0.35 |
| 5 | none | Kmeans tried with 9 clusters | 0.36 |
| 6 | none | Kmeans tried with 7 clusters | 0.45 |
| 7 | none | Kmeans tried with 11 clusters | 0.55 |
| 8 | none | Kmeans tried with 5 clusters | 0.55 |
| 9 | none | Kmeans 6 cluster and random state 0 | 0.8009 |
| 10 | none | Kmeans 7 cluster and random state 0 | 0.553 |
| 11 | none | Kmeans 7 cluster and random state 0 | 0.553 |
| 12 | Deleted all zero columns | Kmeans 6 cluster and random state 0 | 0.8009 |
| 13 | Deleted all zero columns | Kmeans 6 cluster and random state 7 | 0.5716 |
| 14 | Deleted all zero columns | Kmeans 6 cluster and random state 20 | 0.49 |
| 15 | Deleted all zero columns | Kmeans 13 cluster and random state 0 | 0.55 |
| 16 | Deleted all zero columns | Kmeans 13 cluster and random state 7 | 0.55 |
| 17 | Deleted all zero columns | Kmeans 6 cluster and random state 0 | 0.8009 |
| 18 | Deleted all zero columns | Kmeans 13 cluster and random state 7 | 0.5545 |
| 19 | Deleted all zero columns | Kmeans 13 cluster and random state 70 | 0.523 |
| 20 | none | Kmeans tried with 7 clusters | 0.4576 |
| 21 | none | Kmeans tried with 40 clusters | 0.31641 |
| 22 | Deleted all zero columns | Kmeans 18 cluster and random state 0 | 0.385 |
| 23 | Deleted all zero columns | Kmeans 16 cluster and random state 0 | 0.530 |
| 24 | Deleted all zero columns | Kmeans 40 cluster and random state 0 | 0.316 |
| 25 | Deleted all zero columns | Kmeans 6 cluster and random state 7 | 0.494 |
| 26 | Deleted all zero columns | Kmeans 16 cluster and random state 7 | 0.519 |
| 27 | Deleted all zero columns | Kmeans 12 cluster and random state 7 | 0.589 |
| 28 | Deleted all zero columns | Kmeans 14 cluster and random state 7 | 0.507 |
| 29 | Deleted all zero columns | Kmeans 18 cluster and random state 7 | 0.439 |
| 30 | Deleted all zero columns | Kmeans 46 cluster and random state 7 | 0.303 |
| 31 | Deleted all zero columns | Kmeans 46 cluster and random state 70 | 0.306 |
| 32 | Deleted all zero columns | Kmeans 12 cluster and random state 70 | 0.517 |
| 33 | Deleted all zero columns | Kmeans 12 cluster and random state 0 | 0.516 |
| 34 | Deleted all zero columns | Kmeans++ 8 cluster and random state 0 with many features | 0.6011 |
| 35 | Deleted all zero columns | Kmeans++ 12 cluster and random state 0 with many features | 0.516 |
| 36 | Deleted all zero columns | Kmeans++ 6 cluster and random state 0 with many features | 0.3855 |
| 37 | Deleted all zero columns | Kmeans++ 12 cluster and random state 0 with many features algorithm = elkan | 0.512 |
| 38 | Deleted all zero columns | Kmeans++ 20 cluster and random state 0 with many features algorithm = elkan | 0.426 |
| 39 | Deleted all zero columns | Kmeans++ 8 cluster and random state 0 with many features algorithm = elkan | 0.626 |
| 40 | Deleted all zero columns | Kmeans++ 8 cluster and random state 0 with many features algorithm = elkan | 0.626 |
| 41 | Deleted all zero columns | Kmeans 6 cluster and random state 0 | 0.385 |
| 42 | Deleted all zero columns | Kmeans 6 cluster and random state 0 also dropped columns with high correlation above 0.95 | 0.36 |
| 43 | Deleted all zero columns | Agglomerative 6 cluster and linkage ward | 0.40 |
| 44 | Deleted all zero columns | Agglomerative 12 cluster and linkage ward | 0.57 |
| 45 | Deleted all zero columns | Agglomerative 12 cluster and linkage average | 0.70 |
| 46 | Deleted all zero columns | Agglomerative 6 cluster and linkage average | 0.57 |
| 47 | Deleted all zero columns | Agglomerative 12 cluster and linkage complete | 0.40 |
| 48 | Deleted all zero columns | Agglomerative 12 cluster and linkage single | 0.47 |
| 49 | Deleted all zero columns | Agglomerative 10 cluster and linkage average | 0.57 |
| 50 | Deleted all zero columns | Agglomerative 10 cluster and linkage average | 0.57 |
| 51 | Deleted all zero columns | Agglomerative 23 cluster and linkage average | 0.39 |
| 52 | Same file | Made changes to excel sheet | 0.81496 |
| 53 | Same file | Made changes to excel sheet | 0.8148 |
| 54 | Deleted all zero columns | Agglomerative 14 cluster and linkage average | 0.66 |
| 55 | Deleted all zero columns | Agglomerative 11 cluster and linkage average | 0.63 |
| 56 | Deleted all zero columns | Agglomerative 12 cluster and linkage average full tree | 0.704 |
| 57 | Deleted all zero columns | Agglomerative 12 cluster and linkage average affinity cosine | 0.787 |
| 59 | Deleted all zero columns | Agglomerative 12 cluster and linkage average affinity manhattan | 0.63 |
| 60 | Deleted all zero columns | Agglomerative 12 cluster and linkage average affinity l2 | 0.704 |
| 61 | Deleted all zero columns | Agglomerative 15 cluster and linkage average removed high correlation columns > 0.95 affinity cosine | 0.758 |
| 62 | Deleted all zero columns | Agglomerative 15 cluster and linkage average removed high correlation columns > 0.95 | 0.62 |
| 63 | Deleted all zero columns | Agglomerative 15 cluster and linkage average removed low covariance columns | 0.453 |
| 64 | Deleted all zero columns | Agglomerative 12 cluster and linkage average removed low covariance columns | 0.395 |
| 65 | Deleted all zero columns | Agglomerative 12 cluster and linkage average removed low covariance columns affinity cosine | 0.34 |
| 66 | Deleted all zero columns | Agglomerative 12 cluster and linkage average affinity l2 | 0.704 |
| 67 | Deleted all zero columns | Agglomerative 11 cluster and linkage average removed high correlation columns > 0.95 affinity manhattan | 0.84145 |
| 68 | Deleted all zero columns | Agglomerative 11 cluster and linkage average removed high correlation columns > 0.95 affinity cosine | 0.75 |
| 69 | Deleted all zero columns | Agglomerative 11 cluster and linkage average removed high correlation columns > 0.97 affinity manhattan | 0.84145 |
| 70 | Deleted all zero columns | Agglomerative 15 cluster and linkage average removed high correlation columns > 0.95 affinity manhattan | 0.62 |

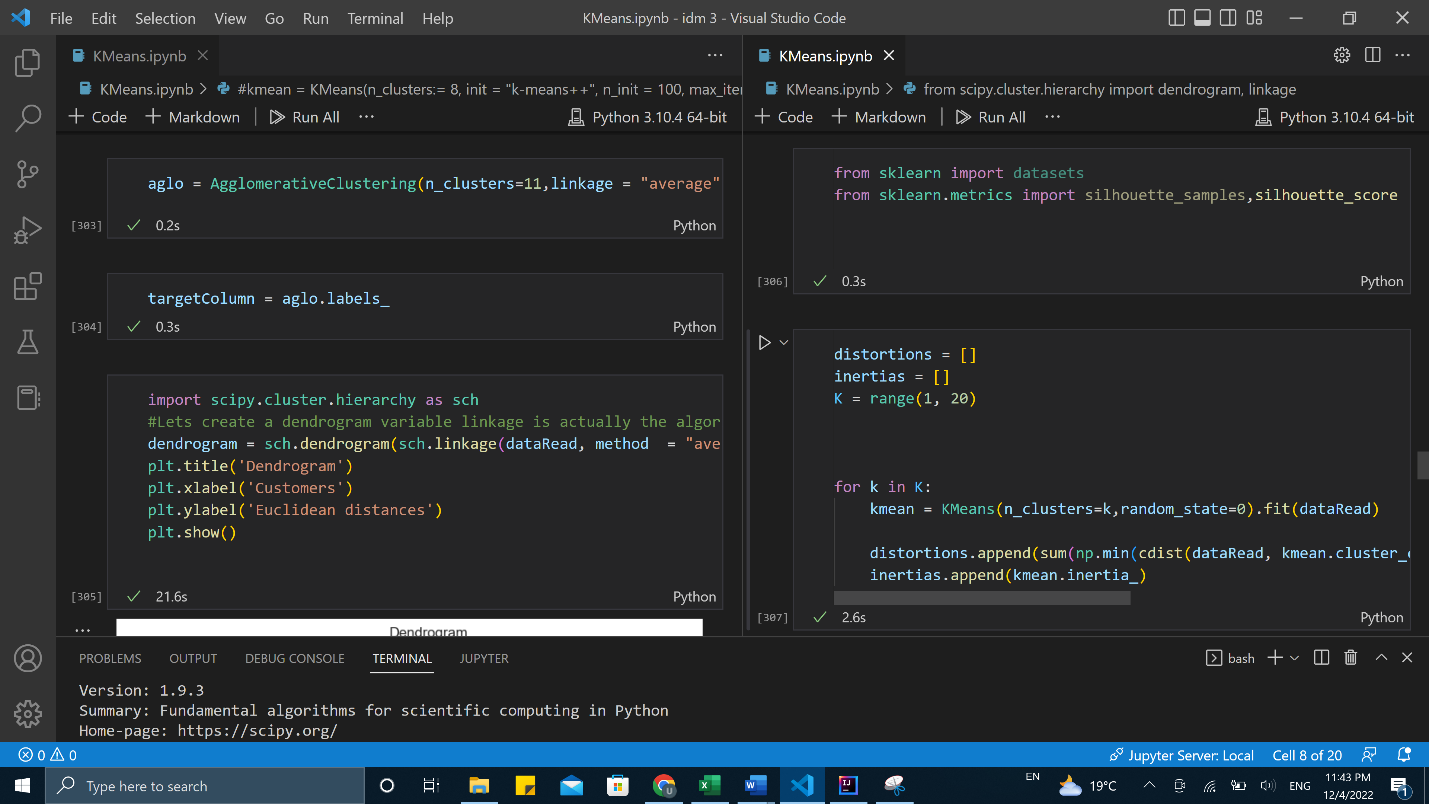
**Code Snapshots**

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence



A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence