Lambton College | In Toronto

Assignment-2

BIG DATA

INSTRUCTOR: GRAHAM WALL

CLASS: CBD 2214

TEAM MEMBERS: LEKSHMI CHANDRAN SHEELA

PARVATHY VYSAKH

PATRICIA ADOLPH

POOJA SELBY

SUCHITRA CHANDRASHEKHARAN

Content

1. Introduction
2. Data Description
3. Exploratory Analysis
4. Data Pre-Processing
5. Model Building
6. Conclusion

Introduction

Obesity is a global phenomenon that is a public health challenge and is one of the major lifestyle diseases found nowadays. Hype in this trend is now leading to a significant increase in obesity-related diseases and deaths. It is a condition that affects people of all ages and gender which can cause long-term or immediate health risks. People having obesity when compared with people with healthy weight are always at a higher risk of getting lifestyle diseases like Blood pressure, high cholesterol, heart disease, stroke, breathing problems, etc. There are some potential factors associated with obesity like high-calorie intake, lack of physical activity, genetic problems, hereditary, anxiety, or depression. Rather than treating problems with obesity, it is always better to prevent obesity as it is very difficult to reverse once founded. So, it is very essential to identify modifiable risk factors for identifying obesity chances in people of a particular age, category, or location to mitigate the risks associated. Studies and prediction models are developed to detect future obesity patterns through which we can have effective interventions to prevent obesity and help people to have a healthier life. The main objective of our model is to analyze and explore the dataset and to find meaningful insights from the dataset. Clustering them based on

Data Description

The dataset for obesity prediction is from UCI Machine Learning Repository. The dataset contains data of individuals collected through surveys from various countries like Columbia, Peru, Mexico based on different factors like their physical conditions, eating and living styles. The dataset has 17 features with 2111 observations. The target variable NObeyesdad has seven classifications: Insufficient weight, Normal Weight, Overweight Level1, Overweight Level2, Obesity Type I, Obesity type II and Obesity type III. BMI of a person was calculated using the Height and Weight feature.

Features & Descriptions

Category Feature Name Description Variable Type

Target Variable NObesity Based on BMI Categorical

Eating Habits FAVC Frequent consumption of high caloric food Categorical

Eating Habits FCVC Frequency of consumption of vegetables Ordinal

Eating Habits NCP Number of main meals Ordinal

Eating Habits CAEC Consumption of food between meals Ordinal

Eating Habits CH20 Consumption of water daily Ordinal

Eating Habits CALC Consumption of alcohol Ordinal

Physical Conditioning SCC Calories consumption monitoring Categorical

Physical Conditioning FAF Physical activity frequency Ordinal

Physical Conditioning TUE Time using technology devices Ordinal

Physical Conditioning MTRANS Transportation used Categorical

Physical Conditioning SMOKE Yes or No Categorical

Responder Characteristics Family History with Overweight Yes or No Categorical

Responder Characteristics Gender Gender is Male or Female Categorical

Exploratory Analysis

Data pre-processing

Data from the real world is always unclean with missing values, errors and can always have noise, outliers, etc. Data pre-processing is the technique to transform real-world data into an understandable format. The data pre-processing involves checking out for lost values, seeking out categorical values, part the dataset into preparing and test set and finally do a highlight scaling to constrain the range of factors. Our dataset has been pre-processed using two methods:

Handling missing values

Missing values should be managed before actual modelling because they lead to faulty decisions. Our first step is to process the missing values. Following are some methods used for handling missing values:

- Delete rows with missing values.
- Manual filling based on similarities between samples.
- Filling missing values using a correlation between variables.
- Deleting redundant data.

The collected data are categorized among two groups—continuous and categorical. The accumulated data in this research are labelled. We have used supervised machine learning models (classification and regression) for training and testing accuracy. Several selected datasets are small, some are noisy, and the remaining contain a good volume of data to train the supervised machine learning model. Data mining was included to filter the data samples from each of the datasets and to discard samples containing outliers. Data mining involves pattern discovery, the calculation of feature association (and correlation), feature selection, classification, clustering, and outlier analysis.

An outlier is a data that stands significantly different from other observations in a dataset. Our dataset had outliers in all features, and it was removed using a box plot.

Outliers are the real values that have a big difference with the respect to the majority of values present in the dataset. These values can affect the performance of the model as optimization of the model is done based on the values available in the dataset.

Using the IQR_upper and IQR_lower values are obtained from quartiles and inter-quartile range where any outliers greater than IQR_upper are updated to be equal to IQR_upper and any outliers lesser than IQR_lower are updated to be equal to IQR_lower.

Here, no outliers have been removed from the numeric_features dataset to prevent the loss of data as we have only 2111 observations to the cluster. (Linear) correlation between numeric features is weak or non-existent. Thus, all features remain in the table.

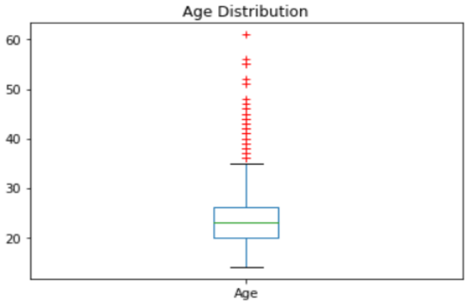


Figure 1: Outliers in the feature AGE

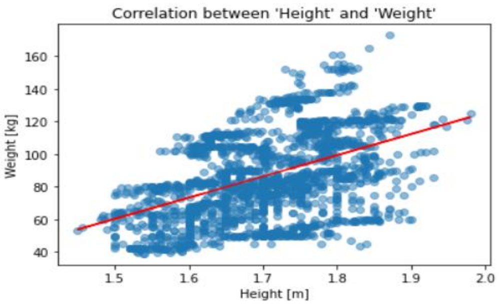


Figure 2: Correlation graph between height and weight

In the correlation graph, the red line plots each person's weight and height. The red line shows that there is a positive correlation between them, which means an increase in one variable leads to an increase in the other. In other words, taller people are more likely to weigh more.

Categorical variables are one-hot encoded with get_dummies. Labels (i.e., the column holding information if a person is overweight/obese or not) are stored in separate variables. So, we created a new feature Obese based on EMI calculation. BMI = Weight/(height^2), if BMI is greater than 30 then it is labelled as 1 (obese) else 0 (not obese). All three sets, numeric features, one-hot encoded and BMI calculated are all concatenated in a new Data Frame which now contains 32 columns. Machine learning algorithms work best with floating-point numbers so for this reason, all the values are converted into floats.

To make features in the same group obesity features are scaled with MinMaxScaler, which makes all the values between 0 and 1. Otherwise, the algorithm might misinterpret and may assign them wrong coefficients (weights). ML classification algorithms expect labels with numeric values (and no strings). For this reason, the obesity class is encoded with LabelEncoder. The latter replaces each class with an integer.

Correlation Matrix

To understand the interrelationship between various features of the dataset, we make use of the correlation matrix. The matrix represents the dependent and independent values by assigning them a score between '-1' and '+1'. Here, '-1' symbolizes negative linear correlation, '0' represents no correlation and '+1' represents positive linear correlation.

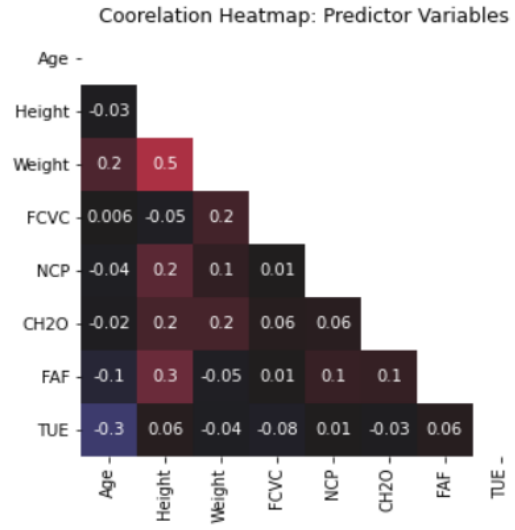


Figure 3: Correlation heatmap of predictor variables

In this matrix, the relationship can be easily identified by the colour assigned to it.

Data Segmentation

Data segmentation is the method of dividing and grouping similar data into distinct groups. Verify our dataset's fitting effect on the model, the whole dataset is segmented into a training set and test set. Then used the training set to train the model and testing set to test the efficiency of the model. Our data is split in the ratio 80:20 for training and testing respectively.

Model Building

Clustering

Clustering is one of the most common exploratory data analyses used to understand the structure of data. In clustering, data are divided into subgroups such that data points inside each subgroup are closely related which means data points will be similar. Clustering is considered an unsupervised learning method; we cannot compare the performance of the model as there is no ground truth to compare the output. Kmeans clustering is one of the clustering algorithms which is quite simple to implement.

KMeans Algorithm

It is an iterative algorithm that is used to partition the dataset into different subgroups such that no data points are common in any of the subgroups created. The algorithm works in such a way that, it makes the data points inside the subgroups more similar to each other, also it makes sure that the different subgroups are different from each other. Data points are selected in such a way that the sum of the square distance between the data points and the centroid is minimum. Kmeans work well if the cluster is spherical.

Working of Kmeans algorithm

Step1.Identify the number of clusters, K

Step2.Initialize centroid by shuffling the dataset and then randomly choose K data points without replacing them.

Step3.Keep on iterating the above step such that there is no change to centroids i.e., the clustering of data points does not change.

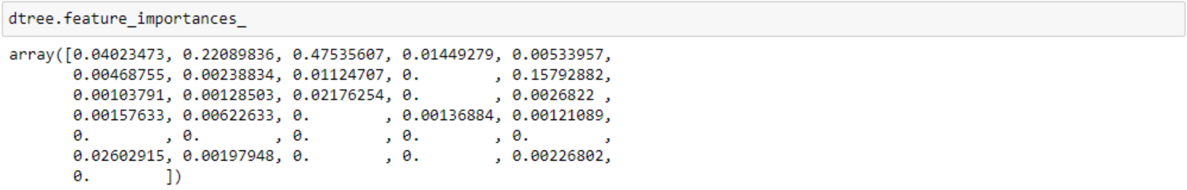
Step4.Calculate the sum of the squared distance between all the centroid and data points.

Step5.Allocate data points to the closest centroid.

Step6.Then calculate the centroid of each cluster taking into account the average of data points that are assigned to each subgroup.

Expectation-Maximization is the approach kmeans follow to solve the problem.

As there are many features in the dataset, the most important features are identified with the help of decision tree classifiers.



```
dtree.feature_importances_
```

```
array([0.04023473, 0.22089836, 0.47535607, 0.01449279, 0.00533957,
       0.00468755, 0.00238834, 0.01124707, 0.        , 0.15792882,
       0.00103791, 0.00128503, 0.02176254, 0.        , 0.0026822 ,
       0.00157633, 0.00622633, 0.        , 0.00136884, 0.00121089,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.02602915, 0.00197948, 0.        , 0.        , 0.00226802,
       0.        ])
```

Figure 4: Array of the importance of each feature wrt target feature

From the above array, it is found that the second and third columns which are the “Height” and “Weight”, are the important ones. They are responsible for 22.1 percent and 47.54 percent of the data, respectively. Also, it is clear that some features hold no relationship with the target variable, these can be removed.

So the steps we performed are first labels are stored in separate variables. Then created a new feature 'Obese' based on BMI Calculation

BMI = Weight/(height^2); if BMI > 30 then 1 (obese) else 0 (not obese)

Then, Features and labels are separated and stored in different variables. Machine Learning algorithms work best with floating-point numbers. Obesity features are then scaled with MinMaxScaler() which makes all values between 0 and 1.ML classification algorithms expect labels with numeric values (and no strings). So, the encoder is instantiated. Then, it "overviews" the data. Here, Cross validation was used to address the problem of the dataset is small. Splitting function (train_test_split) shuffles the data and reserves 20% for testing.

Instantiating requires the number of clusters to form, as well as the number of centroids to generate. The number of clusters is known: 2, based on BMI."K-means++" is the chosen method for initialization - it smartly selects initial cluster centers to speed up convergence.

From the Scatter plot, 2 centroids are seen as red. Data points in black.

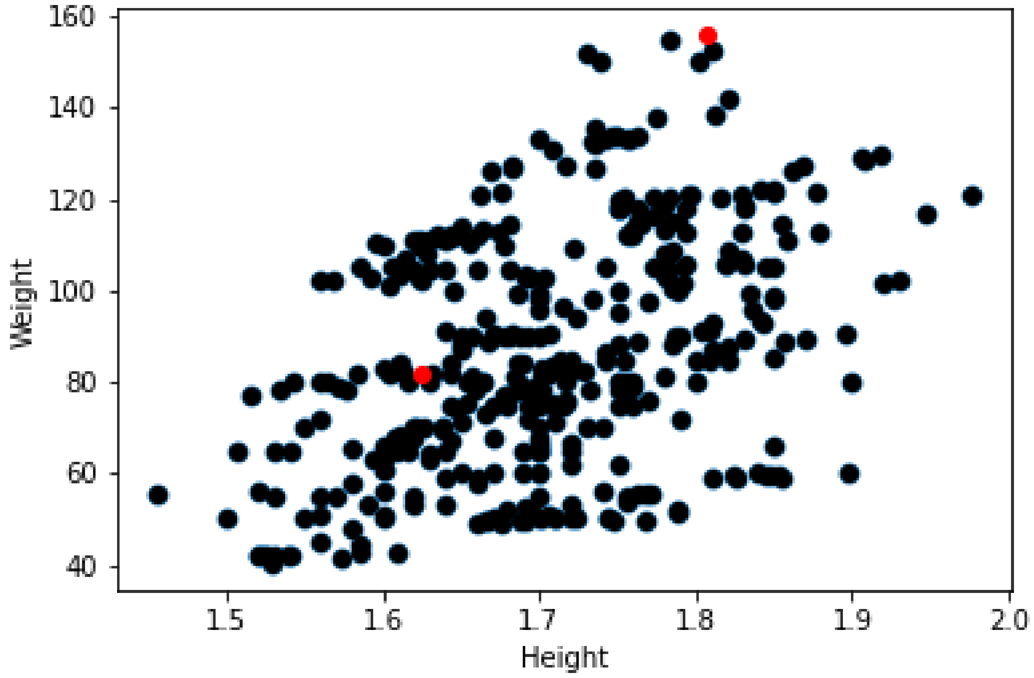


Figure 5: Scatter plot of the data points with centroid

Only the most important features (i.e., those holding the most valuable information) are taken to represent the visualization.

Clusters (formed by "Height" and "Weight" features) in the testing data according to their real labels and predicted clusters are plotted below.

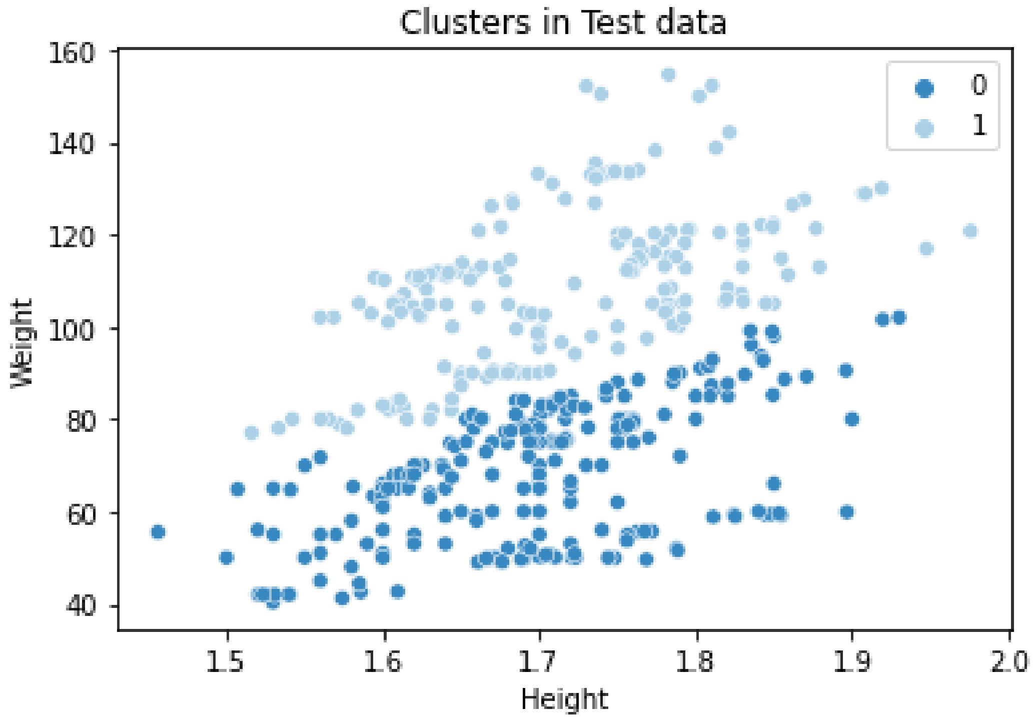


Figure 6 : Cluster in testing data

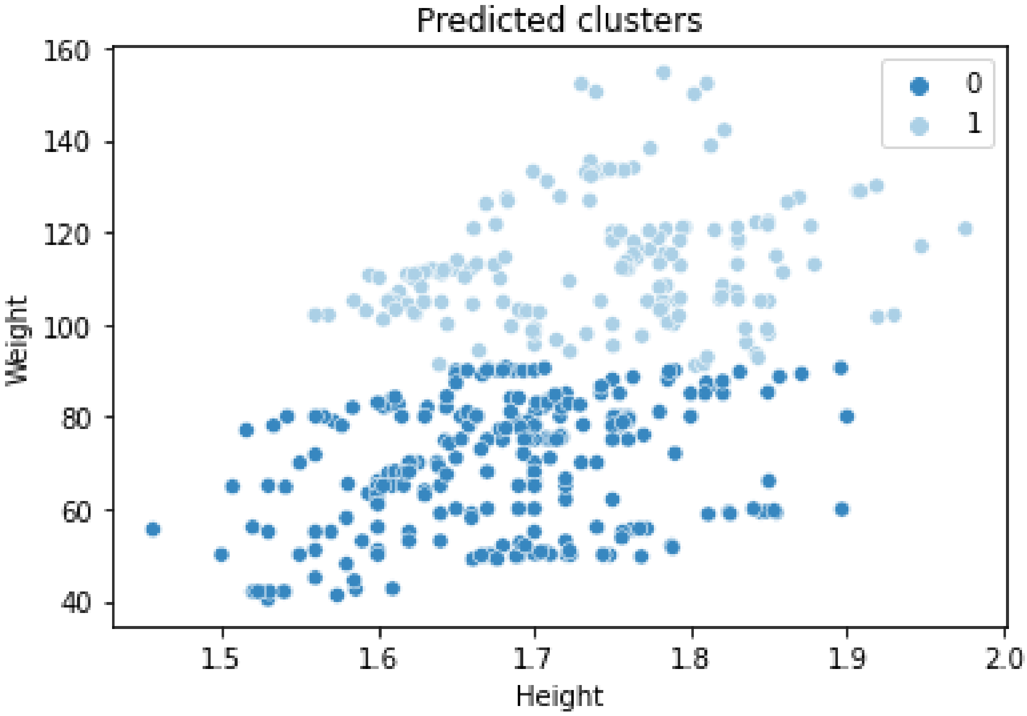


Figure 6: Predicted clusters

Result of Implementation

K-means algorithms consist of 2 major steps: Expectation and Maximization. During the Expectation step, each data point of the Obese feature is assigned to the nearest cluster and then the average of all the data points for each cluster is computed and set as a new centroid during the Maximization step. The number of clusters for the target class which is Obese or not is randomly initialized as 2 for obvious reasons.

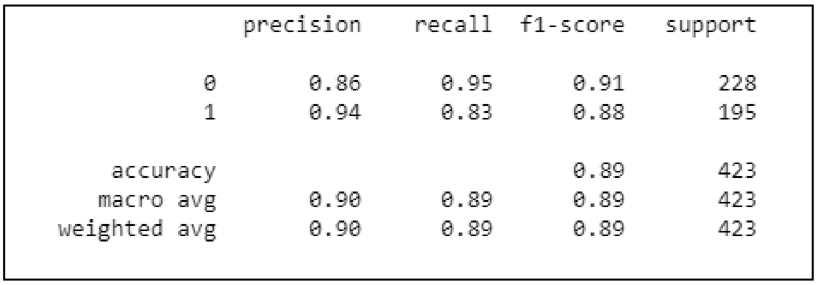
It is inevitable to understand the quality of assigning data points to clusters and hence one of the fundamental steps of any unsupervised machine learning algorithm is to find the optimal number of clusters into which the data points are clustered into and for this, in our implementation, we have used 2 as the values of K which means we have randomly chosen 2 clusters for the initial iteration of modelling. The evaluation of results of the K-Means models compares the actual labels and predicted labels and it was seen that based on the BMI calculation using “Weight” and “Height” features, the model was able to rightly assign the correct labels to all the data points with BMI computed using the following equation:

BMI = Weight/(height^2)   if BMI > 30 then 1 (obese) else 0 (not obese)

We have the following performance evaluation matrices to compute the accuracy and an optimal number of clusters:

Performance evaluation of model:

1. Classification report: This is a summary report for the performance of the modelling algorithm to identify the precision, recall, f1-score and accuracy. The table below shows the classification report based on the confusion matrix.

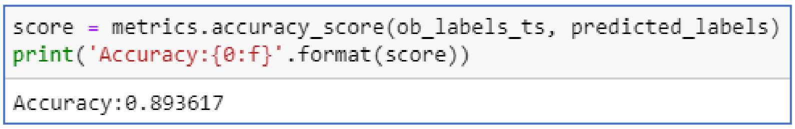


```
              precision    recall  f1-score   support

           0       0.86      0.95      0.91       228
           1       0.94      0.83      0.88       195

    accuracy                           0.89       423
   macro avg       0.90      0.89      0.89       423
weighted avg       0.90      0.89      0.89       423
```

2. Accuracy score: Accuracy is a metric used for evaluating model performance and the fraction of predictions of right predictions our model made. It is computed as below:

To infer our model predicted that out of the 423 data points, 217 points were rightly labelled which shows an accuracy of 89%.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$



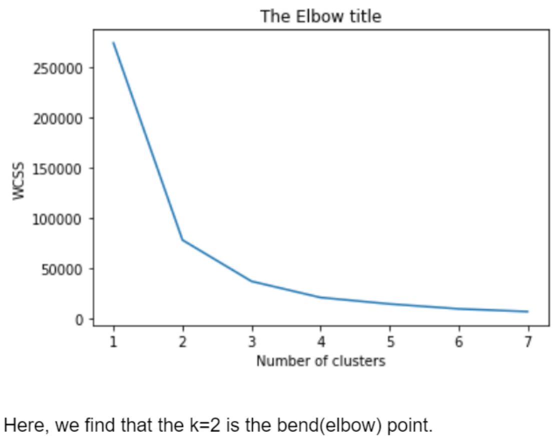
```
score = metrics.accuracy_score(ob_labels_ts, predicted_labels)
print('Accuracy:{0:f}'.format(score))
```

```
Accuracy:0.893617
```

Finding or evaluating optimal clusters

The cluster assignment quality checking is done by calculating the sum of the squared error (SSE) after the entire iteration stops when no further convergence or change in cluster data points are seen. SSE is calculated as the sum of the square of Euclidean distances between each data point and the nearest centroid. K-means aim to minimize the measure of error that SSE determines.

1. Elbow method: This is one of the most commonly used methods for determining the optimal value of K. Here, K-means is iterated along increasing K value to record and plot the SSE of each iteration. The graph of the SSE function shows that as the value of clusters increases the error (SSE) decreases as the greater the number of centroids, the lesser will be the distance between each point and closet centroid. The point where SSE bends in the graph below is called the elbow point. And hence the value of optimal clusters is 2

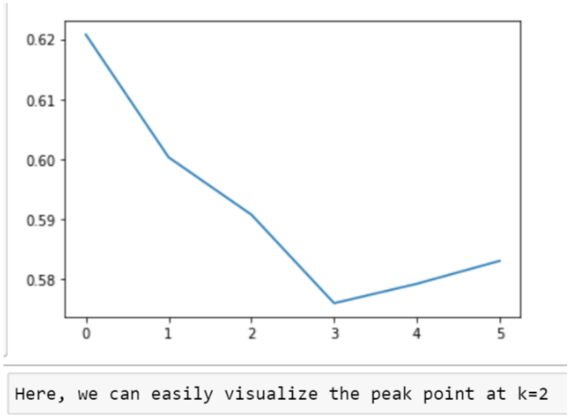


Here, we find that the k=2 is the bend(elbow) point.

Figure 7: Graph of Elbow method with k = 2

2. Silhouette coefficient is a measure that determines the quality of clusters by computing the cluster cohesion and separation. Cluster cohesion identifies how closeness among the

data points in the same cluster whereas separation finds how far the data points are from those in other clusters. A higher value of Silhouette coefficients indicates that the samples are closer to the assigned cluster than other clusters. Using this method the optimal number of clusters find finding the peak point from the graph plotted using the SSE function. And the graph shows a peak at k=2.



Here, we can easily visualize the peak point at k=2

Figure 8: Graph of Silhouette method with k = 2

Conclusion

K-Means clustering is an unsupervised machine learning algorithm that group data into different clusters based on the similarity between each data point. The aim of entire implementation was to group the given observations as obese and not obese, given their eating habits, physical activity and gene factors that determined their obesity predisposition. In order to have an increased accuracy in the model performance using K-Means, we computed the BMI for all the records and determined the quality of clusters using 2 of the most prevalent methods such as Elbow method and Silhouette method to find the optimal number of clusters.

Supervised learning algorithms have ground truth for evaluating model performance unlike unsupervised learning clustering methods like K-Means. A solid evaluation metric is not possible since K-Means requires k as input and there is no learning phase and hence the right label for clusters is also not know beforehand. Domain knowledge and intuition can help to certain extend but this is not the case always. Hence in case of clustering methods we evaluate the quality of clusters by the metrics discussed above and found out that the average silhouette score for cluster size 2 is nearly 60% and is considered as good score for grouping obese and non-obese categories.

References

Machine learning decision tree classification algorithm - javatpoint. www.javatpoint.com. (n.d.). https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm.

12.9 dealing with missing values and Outliers ... (n.d.). https://otexts.com/fpp2/missing-outliers.html.

Singh, B., & Tawfik, H. (2020, May 23). *Machine learning approach for the early prediction of the risk of overweight and obesity in young people*. Computational Science – ICCS 2020: 20th International Conference, Amsterdam, The Netherlands, June 3–5, 2020, Proceedings, Part IV. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7303691/.