

**Department of Artificial Intelligence & Machine Learning**

**Academic Year 2021-22(Odd)**

**Mini Project Report-(20AIM39A)**

**On**

**“****Diet Recommendation System”**

By

**Name Prasanna Kotyal USN 1NH20AI079**

**Name Puneet Vernekar USN 1NH20AI082**

**Name Rahul Ravindra USN 1NH20AI084**

**Under the Guidance of**

Mr. Dinesh J

Assistant professor,

Dept. of Artificial Intelligence & Machine Learning,

New Horizon College of Engineering,

Bangalore-560103



**Department of Artificial Intelligence & Machine Learning**

CERTIFICATE

Certified that the Mini Project work entitled **“**Diet Recommendation System**”** carried out by Mr Prasanna Kotyal USN 1NH20AI079, Mr Puneet Vernekar USN 1NH20AI082, Mr Rahul Ravindra USN 1NH20AI084**.**It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Mini Project work.

**Prof. Dr. N V Uma Reddy**

Guide Head of Department

External Viva

Examiner Signature with date:

1.

2.

**ABSTRACT**

Medical Studies have revealed that consumption of healthy foods help the body to fight against diseases. Food provides our body with essential nutrients needed by the body to sustain us for our day-today activities. It is also important to note that different people have different tastes, likes and dislikes on the choice of food to eat. It is therefore necessary to develop a method to provide every individual with meals of his choice, while ensuring that the correct proportion of nutrients are present in them. We have attempted to solve this problem by developing a diet recommender system. The system is made up of two parts: the first part takes the user’s age and the second part takes in the height and weight of the user to calculate BMI and BMR. The goal of the diet recommender system is to recommend a healthy and appropriate food quantity to users. The recommendation is done using Clustering based algorithm.

**Keywords**: Diet recommender system, Clustering algorithm, Healthy food, BMI(Body Mass Index), BMR(Basal Metabolic rate).

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| Chapter | Topic | Page number |
| 1 | **INTRODUCTION** | **5** |
| 1.1 | **Introduction** | **5** |
| 1.2 | **Objectives** | **5** |
| 1.3 | **Literature survey** | **5** |
| 1.4 | **Proposed System** | **6** |
|  |  |  |
| 2 | **SYSTEM REQUIREMENTS** | **8** |
| 2.1 | **Hardware Requirements** | **8** |
| 2.2 | **Software Requirements** | **8** |
|  |  |  |
| 3 | **SYSTEM ARCHITECTURE AND WORKING** | **9** |
| 3.1 | **System Architecture** | **9** |
| 3.2 | **Flowchart** | **12** |
|  |  |  |
| 4 | **PSEUDOCODE AND RESULTS** | **18** |
| 4.1 | **Pseudocode** | **18** |
| 4.2 | **Result** | **19** |
|  |  |  |
| 5 | **CONCLUSION** | **22** |
| 5.1 | **Conclusion** | **22** |
| 5.2 | **Future Enhancements** | **23** |

**CHAPTER-1**

**1.1 Introduction**

Many factors influence an individual’s health, such as physical exercise, sleep, nutrition, heredity and pollution. Diet being one of the biggest modifiable factors in our lives, small changes to diet can have a big impact. With the exponential increase in the number of available food options, it is not possible to take them all into account anymore. The only way to consider user preferences, maximize the number of healthy compounds and minimize the unhealthy ones in food, is using Diet recommendation system. The success of Diet recommendation system is correlated with its ability to account for user preferences, maximize the number of healthy compounds and minimize the unhealthy ones in food.

**1.2 Objective**

The objectives of the project are:

1. Study and apply the needed tools, namely:
2. K-Means clustering algorithm
3. GUI using Python Tkinter.
4. Develop a diet recommender system.
5. Test the diet recommender system.
6. Document the results of the project
7. Form a report of the features of the project.

**1.3 Literature Survey (Present System)**

In this technological world, there are many existing diet apps, but as per research, there’s none app which matches to ours. Most of the diet apps input the number of calories, carbohydrates, proteins, etc. consumed, from the user, which makes the user bit inconvenient to use the app. User has to have a look over each and every amount of calorie, carbs intake and input these data in the app and then the app generates the diet. Which evolves the complexity to use the app.

****

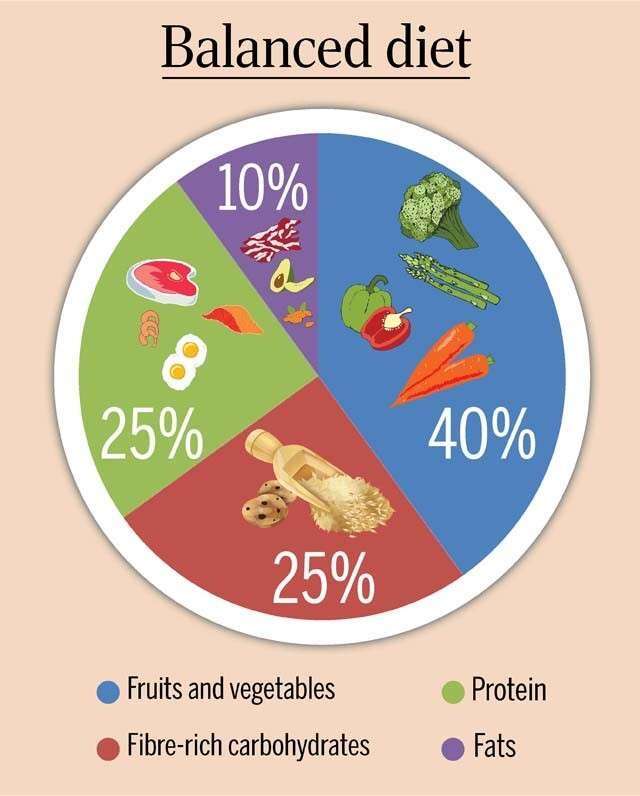
**Figure 1.1 - Balanced diet**

**1.4 Proposed system**

Our diet app mainly concerns to recommend the best diet based on the user input of Age, Height and Weight. Our project is very dynamic and has a high silhouette score to perform at max efficiency

**Table 1**

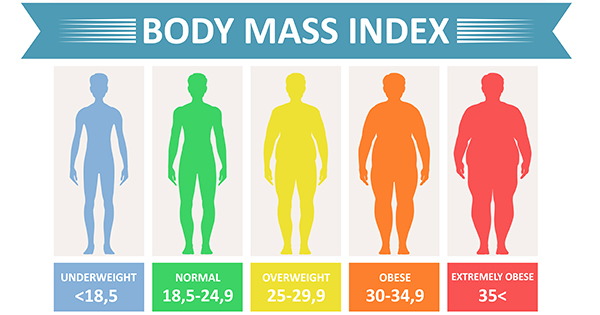
|  |  |  |
| --- | --- | --- |
| **QUANTIFICATION OF FOOD -PER-DAY IN KCAL** | | |
| BREAKFAST PROPORTIONS | LUNCH PROPORTIONS | DINNER PROPORTIONS |
| 435 KCAL | 580 KCAL | 435 KCAL |



**Figure 1.2 – Diet Pie chart**

**Table 2**

|  |  |  |  |
| --- | --- | --- | --- |
| **GRAM EQUIVALENCE OF FOOD NUTRIENT** | | | |
| MACRO FOOD NUTRIENT | CARBOHYDRATE | PROTEIN | FAT |
| BREAKFAST | 59.8 gm | 27.2 gm | 9.7 gm |
| LUNCH | 79.8 gm | 36.3 gm | 12.9 gm |
| DINNER | 59.8 gm | 27.2 gm | 9.7 gm |



**Figure 1.3 – BMI chart**

**CHAPTER-2**

**2.1 HARDWARE REQUIREMENTS:**

* 4GB Ram
* 256 GB Rom
* Intel Pentium or above processor
* 100mb Gpu

**2.2 SOFTWARE REQUIREMENTS:**

* Python 3.0 or above
* Modules- sklearn, tkinter, pandas and numpy
* OS – Windows/Mac/Linux/Ubuntu
* Web Browser – Chrome/Firefox/Microsoft Edge (Latest versions preferable)

**CHAPTER-3**

**3.1 System Architecture:**

The project architecture consists of the following steps:

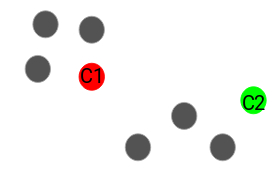
**Step 1:** Choose the number of clusters *k*

The first step in k-means is to pick the number of clusters, k.

**

**Figure 3.1 – Pre clustering data points**

**Step 2:** Select k random points from the data as centroids

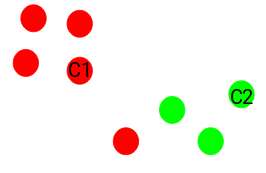
Next, we randomly select the centroid for each cluster. Let’s say we want to have 2 clusters, so k is equal to 2 here. We then randomly select the centroid:

**Figure 3.2 – Centroid of the clustered datapoints**

Here, the red and green circles represent the centroid for these clusters.

### Step 3: Assign all the points to the closest cluster centroid

Once we have initialized the centroids, we assign each point to the closest cluster centroid:

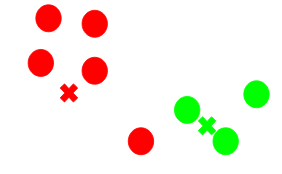
****

**Figure 3.3 – Datapoints assignment to the respective centroid**

Here you can see that the points which are closer to the red point are assigned to the red cluster whereas the points which are closer to the green point are assigned to the green cluster.

### Step 4: Recompute the centroids of newly formed clusters

Now, once we have assigned all of the points to either cluster, the next step is to compute the centroids of newly formed clusters:

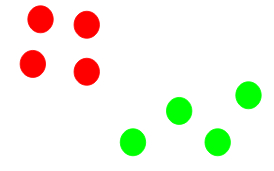


**Figure 3.4 – Recomputation of centroids**

Here, the red and green crosses are the new centroids.

### Step 5: Repeat steps 3 and 4

We then repeat steps 3 and 4:



**Figure 3.5 – Recursive procedure of the previous steps**

The step of computing the centroid and assigning all the points to the cluster based on their distance from the centroid is a single iteration.

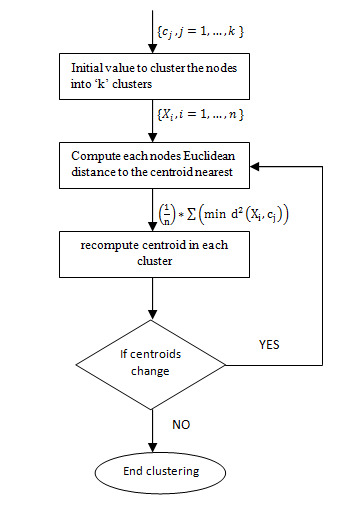
**Note:**

### Stopping Criteria for K-Means Clustering

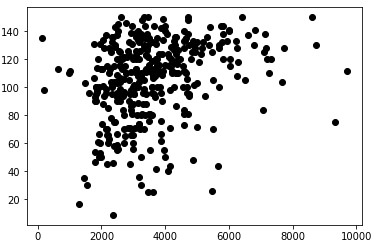
### There are essentially three stopping criteria that can be adopted to stop the K-means algorithm:

1. Centroids of newly formed clusters do not change
2. Points remain in the same cluster
3. Maximum number of iterations are reached

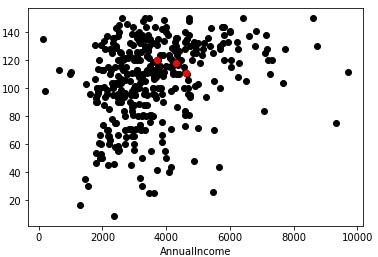
**3.2 FLOWCHART:**

****

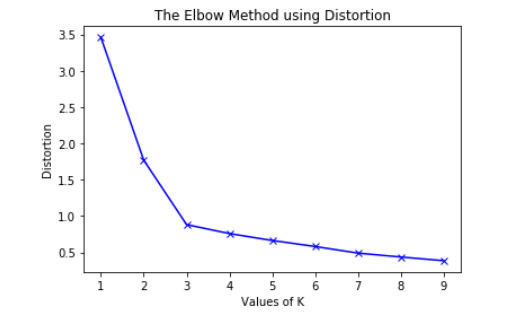
* **Initial condition of clustering the data:**

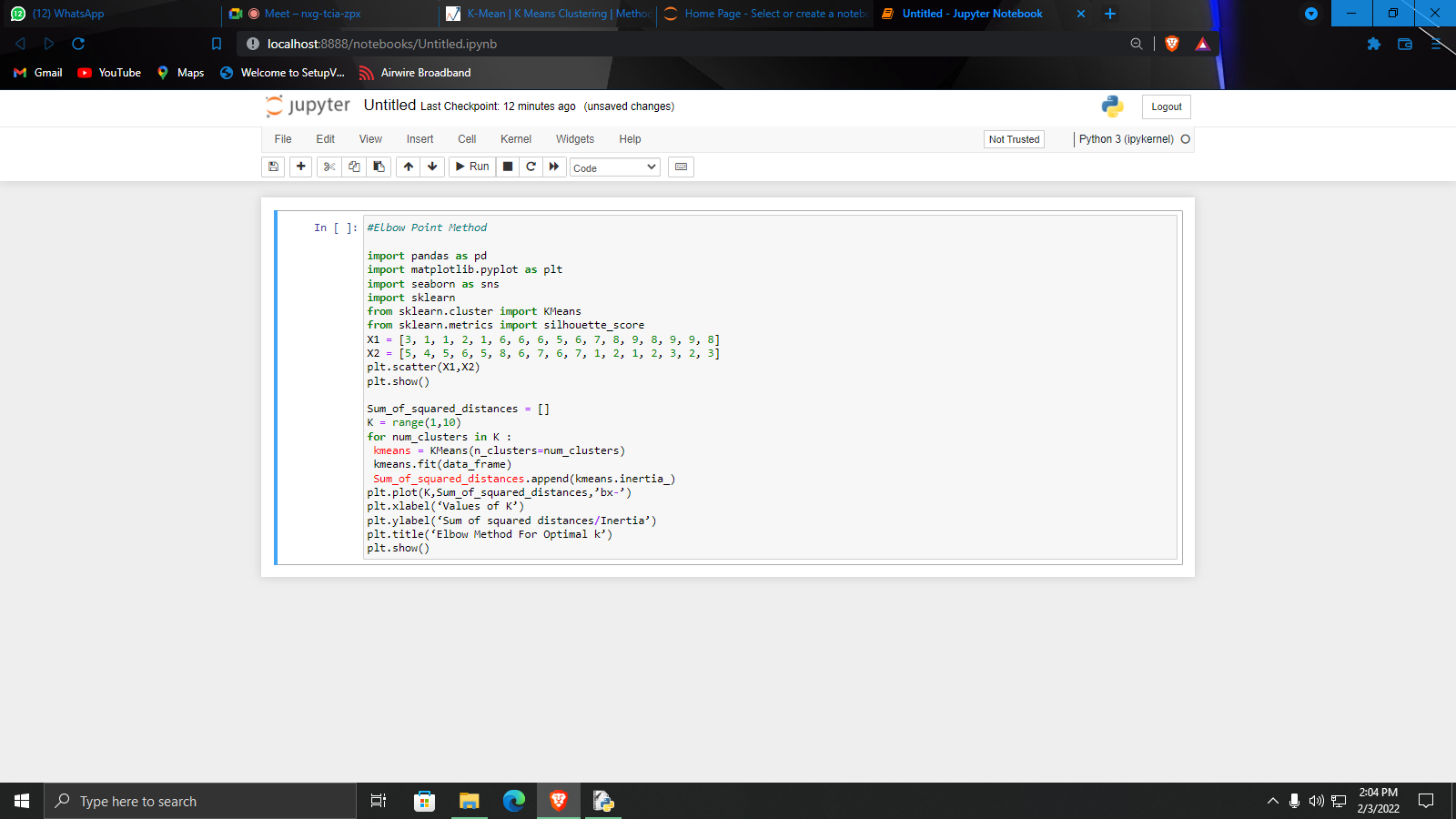
****

* **Choosing the optimal number of clusters:**

**`**

**Methods to choose best number of clusters: -**

1. **Elbow point method:**

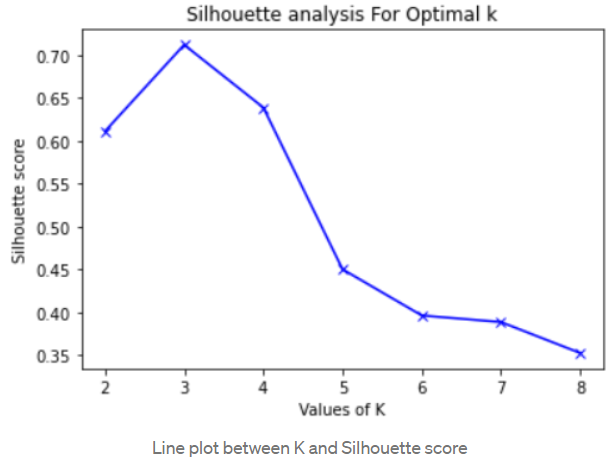


**Figure 3.6 – Plot of elbow point**

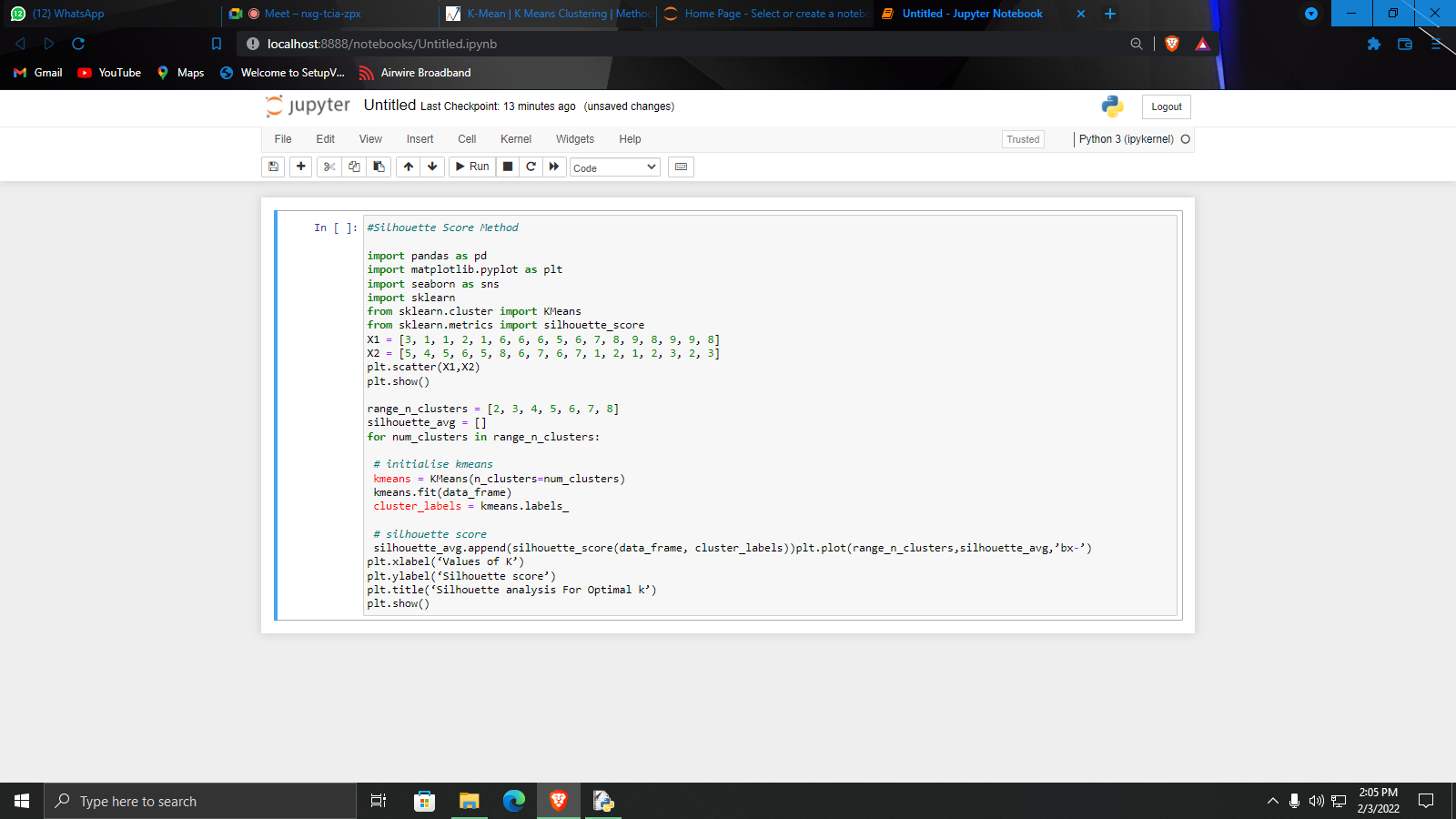
**Figure 3.7 – Code for elbow point**

* The elbow method runs k-means clustering on the dataset for a range of values of k (say 1 to 10).
* Perform K-means clustering with all these different values of K. For each of the K values, we calculate average distances to the centroid across all data points.
* Plot these points and find the point where the average distance from the centroid falls suddenly (“Elbow”).

**2) Silhouette analysis**

****

**Figure 3.8 – Plot of Silhouette analysis**



**Figure 3.9 – Code for Silhouette analysis**

The silhouette coefficient is a measure of how similar a data point is within-cluster (cohesion) compared to other clusters (separation).

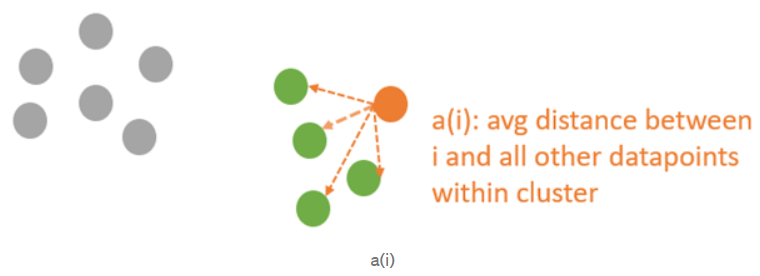
Select a range of values of k (say 1 to 10).

Plot Silhouette coefﬁcient for each value of K.

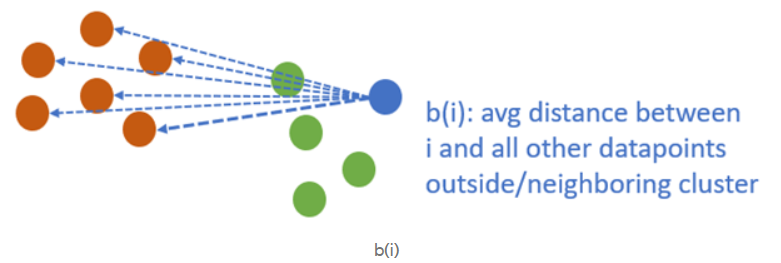
The equation for calculating the silhouette coefﬁcient for a particular data point:

silhouette score

S(i) is the silhouette coefficient of the data point i.

**fig 3.10 -** is the average distance between i and all the other data points in the cluster to which i belongs.

**Figure 3.10**

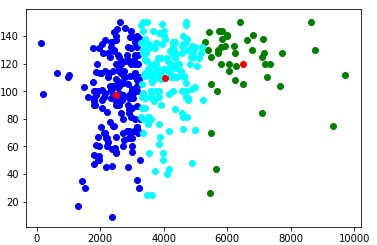
**fig 3.11 -** is the average distance from i to all clusters to which i does not belong.

**Figure 3.11**

We will then calculate the average silhouette for every k.

Then plot the graph between average silhouette and K.

* **Post cluster formation diagram:**

****

**CHAPTER-4**

**4.1 Pseudocode:**

1) At first user is asked to input his/her age, weight and height using a popup window made by the GUI module Tkinter.

2) After inputting the details, the values of the dataset is stored in a variable(data) and all the names of the food items from the dataset are stored in another variable(fooditems) in the form of numpy array.

3) Then the food items are bifurcated i.e., all the food items which are to be included in training of the model are included in the variable(foodid) in the form of numpy array.

4) Then all the data of those food items is stored in a variable (foodiddata).

5) Then the BMI is calculated based on the users input and based on the BMI user is given a tag as underweight or overweight, etc.

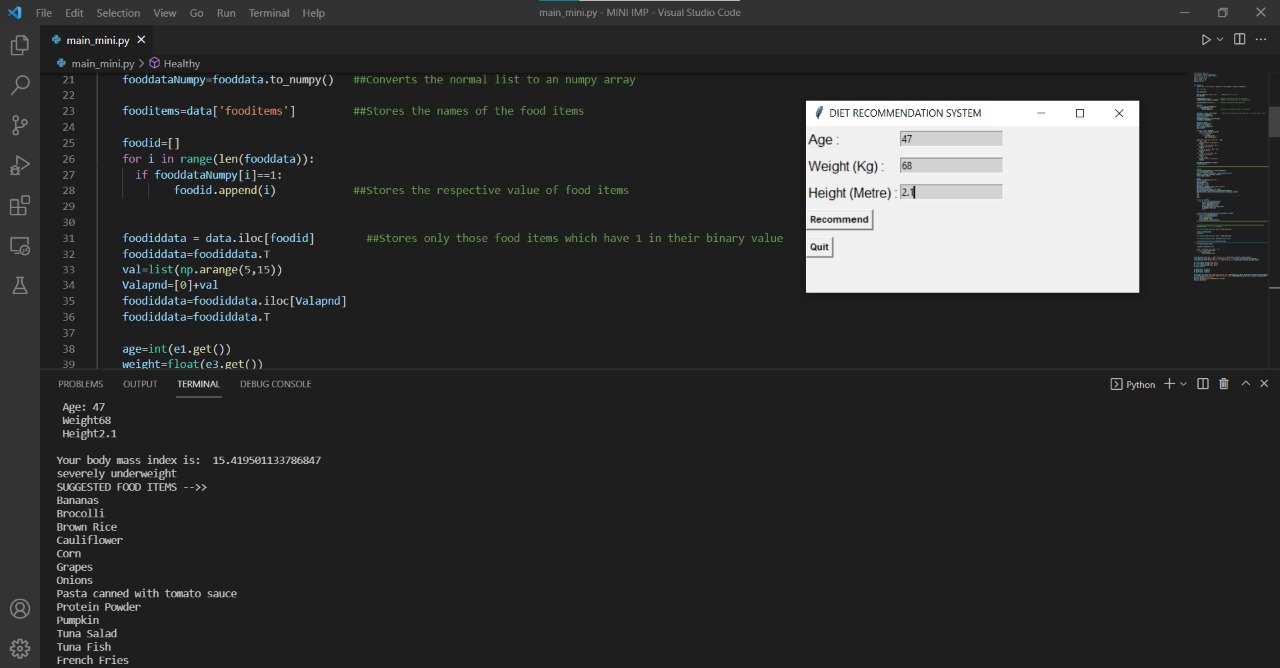
6) Now the data about the food items stored in the variable is being extracted and made into different clusters (using Kmeans clustering) based on their nutrients value.

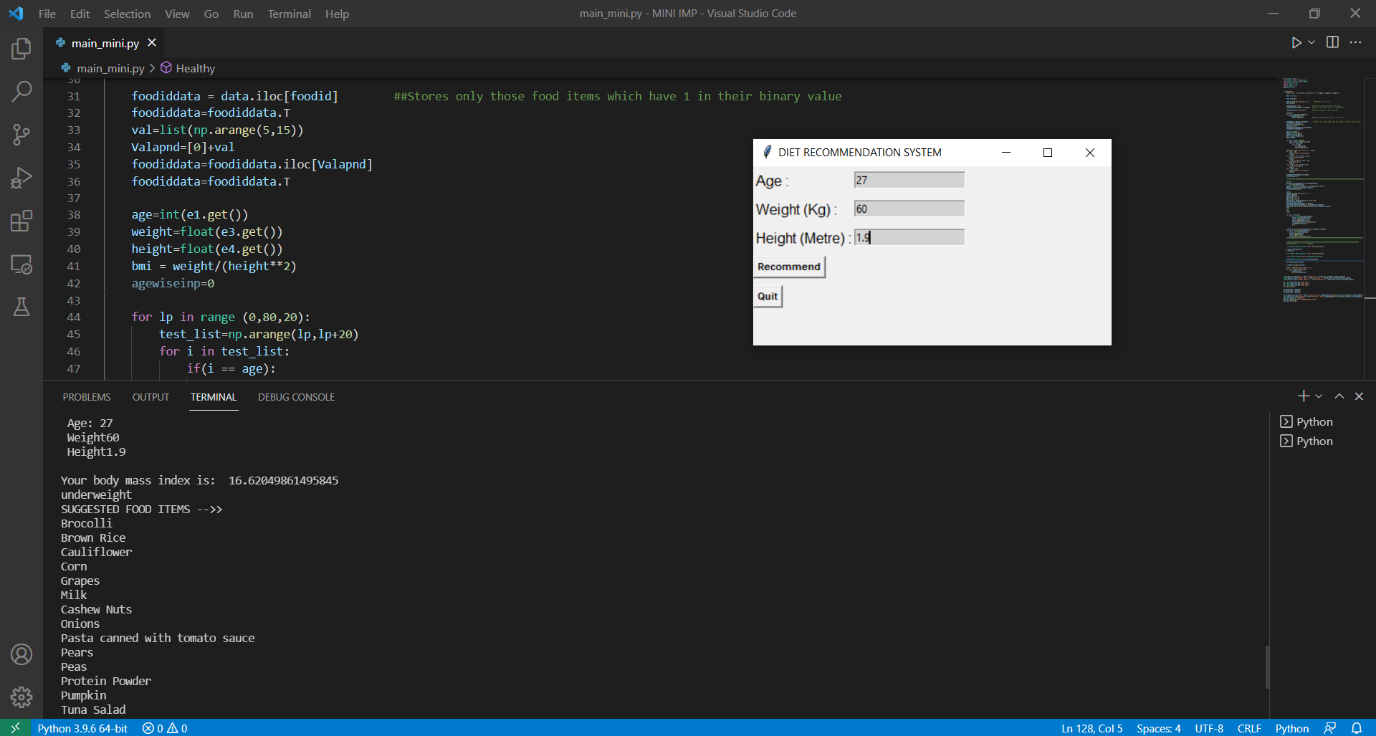
7) Then this model is trained using Sklearn and an appropriate cluster is chosen by the ML model in concurrence with the user input.

8) The system finally recommends several food items using the ML algorithm(Kmeans) and the information of the user data.

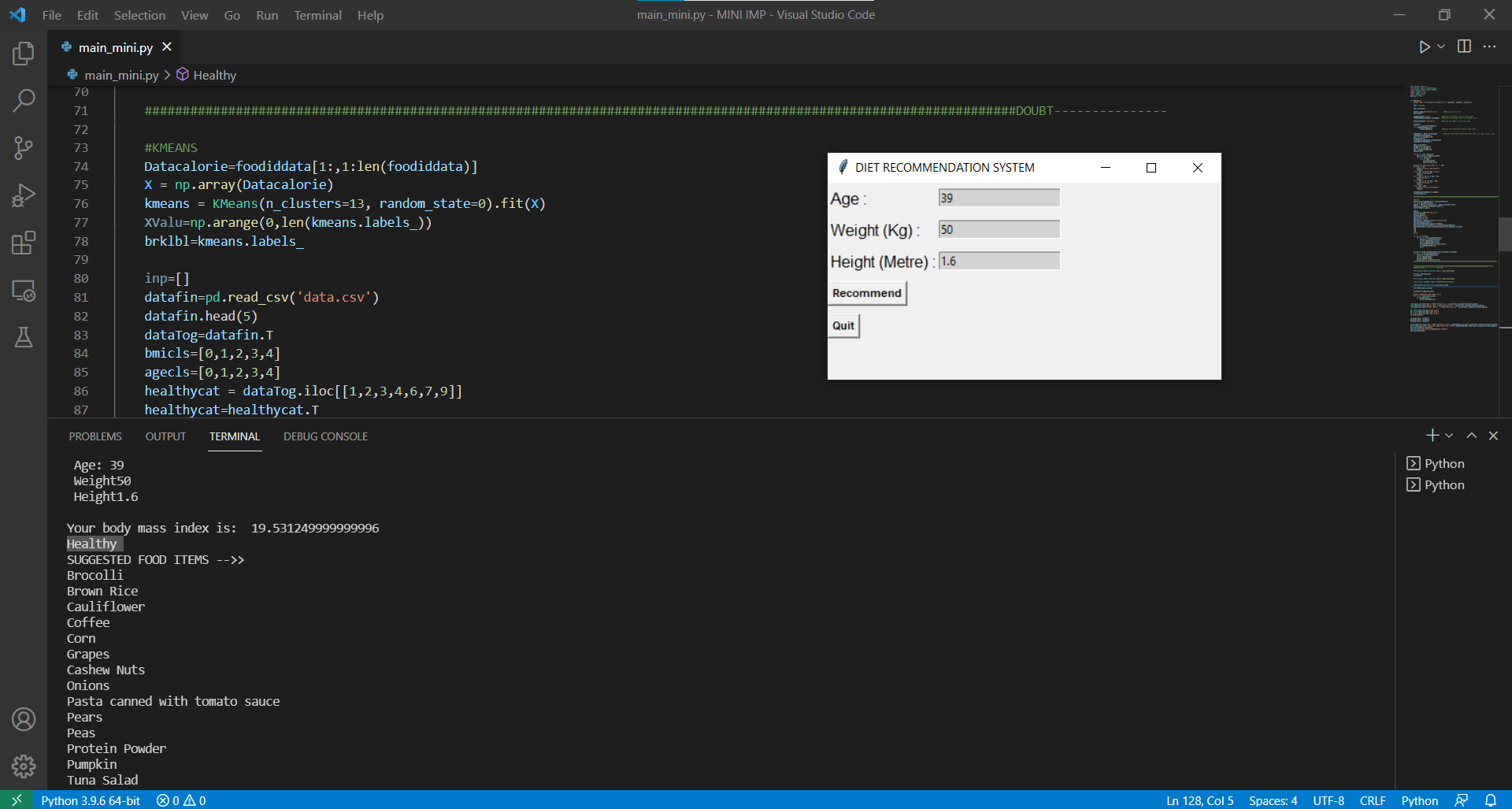
**4.2 Results:**

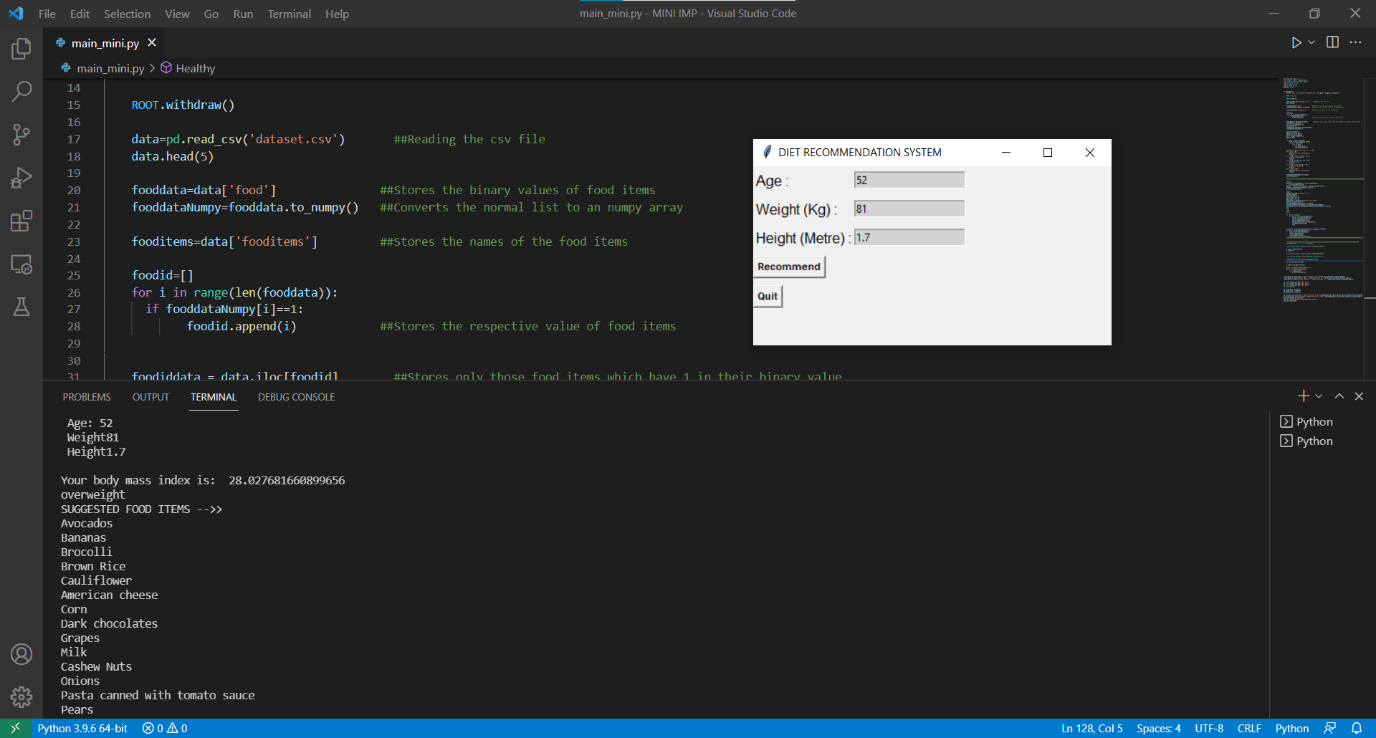
**Case 1: Severely underweight**

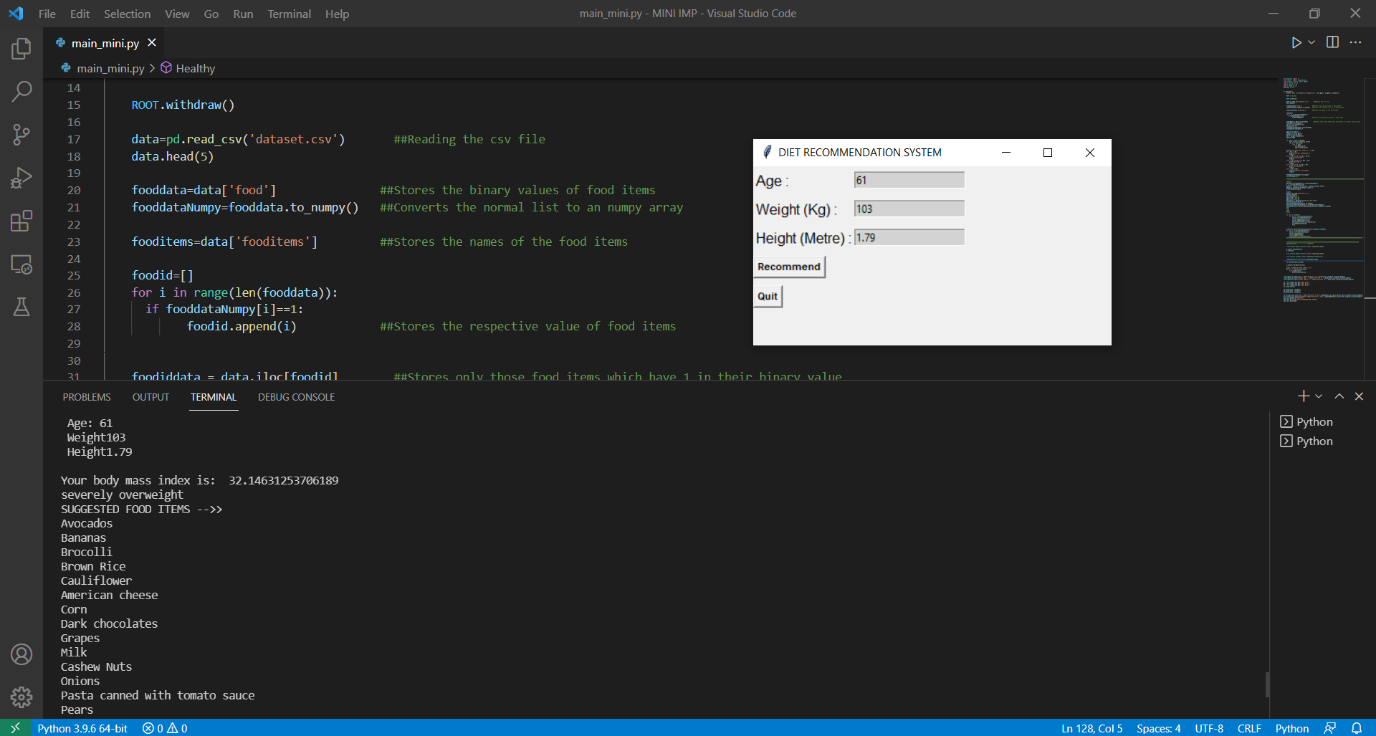
****

**Case 2: Underweight**

**Case 3: Healthy**

****

**Case 4: Overweight**

**Case 5: Severely overweight**

**Chapter 5**

**5.1 Conclusion**

Good nutrition and a balanced diet is fundamental for current and future health, as well as for development and learning. The benefits of developing healthy dietary and lifestyle patterns from an early age onwards can positively impact on people’s nutrition and health throughout their adult lives, and enhance the productivity of individuals and nations.

Due to heavy information overload triggered by the Internet, extracting accurate information about diet becomes increasingly difficult. In this context, diet recommendation system becomes an effective tool to extract useful information and deliver it in an efficient way.

In conclusion diet recommendation system is explored as an effective tool in order to support users for changing their eating behavior and reaching the goal of healthier food choices

**5.2 Future Enhancements**

1. Dataset can be enhanced and feature like disease can be included
2. Food category like veg/non-veg and seasonal based recommendation can be performed
3. Web application can be developed

**References:**

[1] T. N. Trang Tran, M. Atas, A. Felfernig, and M. Stettinger, ``An overview

of recommender systems in the healthy food domain,'' *J. Intell. Inf. Syst.*,

vol. 50, no. 3, pp. 501\_526, Jun. 2018.

[2] C. Chensi, L. Feng, T. Hai, S. Deshou, S. Wenjie, L. Weizhong,

Z. Yiming, B. Xiaochen, and X. Zhi, ``Deep learning and its applications

in biomedicine,'' *Genomics Proteomics Bioinf.*, vol. 16, no. 1, pp. 17\_32,

2018.

[3] C. Iwendi*et al.*: Realizing an Efficient IoMT-Assisted Patient Diet Recommendation System Through Machine Learning Model

[4] P. Maiyaporn, P. Phathrajarin, and P. Suphakant, ``Food recommendation

system using clustering analysis for diabetic patients,'' in *Proc. Int. Conf.*

*Inf. Sci. Appl.*, vol. 6, no. 2, 2010, pp. 5\_14.

[5] Published in: 2019 IEEE 14th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)

[6]Jong-Hun Kim; Jung-Hyun Lee; Jee-Song Park; Young-Ho Lee; Kee-Wook Rim Published in: 2009 Fourth International Conference on Computer Sciences and Convergence Information Technology.

[7] Raciel Yera Toledo; Ahmad A. Alzahrani; Luis Martínez: Published in: IEEE Access ( Volume: 7) A Food Recommender System Considering Nutritional Information and User Preferences.