**PREDICTIVE ANALYTICS**

**PROJECT REPORT**

(Project Semester September-December 2025)

***Health\_Lifestyle***

Submitted by

P. Rama Sai Jahnavi

12308734

B-TECH Computer Science and Engineering

INT234

Under the Guidance of

**Dr. Tanima Thakur (UID: 23532)**

**Discipline of CSE/IT**

**Lovely School of Computer Science**

**Lovely Professional University, Phagwara**

**CERTIFICATE**

This is to certify that P. Rama Sai Jahnavi bearing Registration no. 12308734 has completed INT234 project titled, **“Health\_Lifestyle\_Analysis”** under my guidance and supervision. To the best of my knowledge, the present work is the result of his/her original development, effort and study.

**Signature and Name of the Supervisor**

**Designation of the Supervisor**

**School of Computer Science**

Lovely Professional University

Phagwara, Punjab.

Date: 13-12-2025

**DECLARATION**

I, P. Rama Sai Jahnavi, student of B-Tech CSE under CSE/IT Discipline at, Lovely Professional University, Punjab, hereby declare that all the information furnished in this project report is based on my own intensive work and is genuine.

Date: 13-12-2025 P. Jahnavi

Registration No. 12308734 P. Rama Sai Jahnavi

**ACKNOWLEDGEMENT**

I take this opportunity to express my deep sense of gratitude to all those who have contributed to the successful completion of my project titled **"Health\_Lifestyle\_Analysis."**

I would like to extend my heartfelt thanks to my project guide, **Dr. Tanima Thakur mam**, for their invaluable guidance, constant encouragement, and support throughout the duration of this project. Their insights and suggestions played a crucial role in shaping the direction of my work.

I am also thankful to the **Computer Science** of **Lovely Professional University** for providing the necessary infrastructure and resources to carry out this project effectively.

My sincere thanks to the creators of the **Healthy Lifestyle Analysis** for making the data publicly available, which was instrumental in performing this analysis.

I would also like to express my gratitude to my classmates, friends, and family for their continuous motivation and support.

Finally, I acknowledge the use of various open-source tools and libraries such as **Python, Pandas, NumPy, Matplotlib, Seaborn, Scikit-learn,** and **Lifelines** that made data analysis and visualization possible.

**TABLE OF CONTENT**

|  |  |  |
| --- | --- | --- |
| **SR. NO** | **CONTENTS** | **Page. No** |
| **1** | **COVER PAGE** | **1 – 1** |
| **2** | **DECLARATION** | **2 – 2** |
| **3** | **CERTIFICATE** | **3 – 3** |
| **4** | **ACKNOWLEDGEMENT** | **4 – 4** |
| **5** | **INTRODUCTION** | **6 – 7** |
| **6** | **SOURCE OF DATASET** | **8 – 8** |
| **7** | **EDA PROCESS** | **9 – 10** |
| **8** | **ANALYSIS ON DATASET** | **11 – 61** |
| **9** | **CONCLUSION** | **62 – 64** |
| **10** | **FUTURE SCOPE** | **65 – 65** |
| **11** | **REFERENCES** | **66 – 66** |
| **12** | **LINKEDIN LINK** | **67 – 67** |
| **13** | **GITHUB LINK** | **67 – 67** |
| **14** | **GOOGLE DRIVE LINK** | **67 – 67** |
| **15** | **GOOGLE FORM LINK** | **67 – 67** |

**INTRODUCTION**

In today’s fast-paced and technology-driven world, human lifestyle patterns have undergone significant changes, directly influencing physical health, mental well-being, and overall quality of life. Factors such as prolonged screen time, reduced physical activity, irregular sleep cycles, unhealthy dietary habits, and increasing stress levels have contributed to the rise of lifestyle-related health problems. Conditions such as obesity, poor sleep quality, high stress, and reduced productivity are becoming increasingly common across different age groups. As a result, there is a growing need to understand how daily lifestyle choices impact health outcomes and to develop systems that can predict health risks at an early stage.

With the rapid advancement of data collection technologies and analytical tools, **data science and machine learning** have emerged as powerful approaches for analyzing complex health-related data. Predictive analytics enables the identification of hidden patterns, relationships, and trends within large datasets, allowing researchers to make informed predictions and data-driven decisions. In the healthcare domain, predictive models play a crucial role in preventive care by forecasting potential health issues before they become severe, thereby supporting timely intervention and lifestyle modification.

This project focuses on the **predictive analysis of health and lifestyle data** collected directly from individuals in real time using **Google Forms**. Unlike synthetic or publicly available datasets, the data used in this study represents real human responses and behaviors, making the analysis more realistic and practical. The dataset includes a wide range of attributes such as age, gender, occupation, sleep hours, screen time, exercise duration, water intake, junk food frequency, alcohol consumption, daily steps, number of meals, stress level, happiness level, productivity score, sleep quality, and Body Mass Index (BMI). These attributes collectively describe an individual’s lifestyle and overall health condition.

Since the dataset is collected from real users, it naturally contains missing values, inconsistencies, and variations that reflect real-world data challenges. Handling such data is an essential part of any predictive analytics project. Therefore, data preprocessing steps such as missing value treatment, duplicate removal, encoding of categorical variables, and feature scaling are performed to ensure data quality and reliability. This enhances the performance and accuracy of machine learning models applied in later stages of the project.

The primary goal of this project is to **analyze the relationship between lifestyle factors and health indicators** and to develop predictive models that can estimate health outcomes, particularly **BMI and health categories**, based on individual lifestyle habits. BMI is considered a key health metric as it provides an indication of whether a person is underweight, normal, overweight, or obese. Predicting BMI using lifestyle parameters can help individuals understand the long-term effects of their daily habits and encourage healthier behavioral choices.

To achieve this objective, the project employs a combination of **exploratory data analysis (EDA)** and **machine learning techniques**. EDA is used to visualize data distributions, identify correlations among variables, and gain meaningful insights into lifestyle-health relationships. Visual tools such as histograms, scatter plots, box plots, bar charts, and correlation heatmaps help in understanding trends such as the impact of sleep hours on stress level or the influence of junk food consumption on BMI.

Following the exploratory phase, several predictive models are implemented and evaluated. These include **Linear Regression**, **Polynomial Regression**, **K-Nearest Neighbors (KNN)**, **Decision Tree**, and **Random Forest** algorithms. Each model is trained and tested using appropriate data splits, and their performance is measured using standard evaluation metrics such as accuracy, precision, recall, F1-score, mean absolute error (MAE), root mean square error (RMSE), and R² score. Feature selection techniques are also applied to identify the most influential lifestyle parameters contributing to BMI prediction.

A significant aspect of this project is the comparison of multiple models to determine which algorithm performs best for this dataset. By analyzing and comparing results across different models, the study highlights the strengths and limitations of each technique in handling real-world health data. This comparative approach not only improves predictive accuracy but also enhances understanding of model behavior and interpretability.

Furthermore, this project demonstrates the practical application of machine learning in the health domain and emphasizes the importance of data-driven decision-making. The findings from this analysis can be used to create awareness about healthy lifestyle choices and can serve as a foundation for future health-based applications such as personalized health recommendation systems, fitness tracking platforms, or intelligent health monitoring solutions.

In conclusion, this predictive analytics project bridges the gap between raw lifestyle data and actionable health insights by leveraging real-time collected data and machine learning techniques. By analyzing lifestyle habits and predicting health outcomes, the project contributes to preventive healthcare and promotes a data-centric approach to improving individual well-being. The methodologies and results presented in this study can be extended further to larger populations and more advanced predictive systems in future research.

**SOURCE OF DATASET**

The dataset used in this predictive analytics project was **collected in real time from individuals using Google Forms**. The data was not obtained from any secondary source or publicly available repository; instead, it was generated through **primary data collection**, making it highly relevant and realistic for analysis.

A structured Google Form was designed to capture various **health and lifestyle-related attributes** from participants. The form included questions related to demographic details (such as age, gender, and occupation), daily lifestyle habits (including sleep duration, screen time, exercise time, water intake, junk food consumption, alcohol intake, number of meals, and daily steps), and health indicators (such as stress level, happiness level, productivity score, sleep quality, and Body Mass Index). Participants voluntarily submitted their responses based on their real-life habits and personal experiences.

The data collection process was carried out over a period of time, allowing responses from individuals belonging to different age groups and occupational backgrounds. Since the dataset was collected directly from users, it reflects **natural variations, missing values, and inconsistencies** that are commonly observed in real-world datasets. These characteristics make the dataset suitable for applying data preprocessing techniques and evaluating the robustness of predictive models.

To ensure data usability for machine learning analysis, the collected responses were exported from Google Forms into a spreadsheet format (Excel). Basic data validation was performed, followed by preprocessing steps such as handling missing values, removing duplicate records, encoding categorical variables, and normalizing numerical features. No personally identifiable information (PII) was collected, ensuring that the dataset adheres to ethical and privacy considerations.

In summary, the dataset used in this project is a **real-time, self-reported health and lifestyle dataset collected through Google Forms**, serving as a reliable foundation for exploratory data analysis and predictive modeling. The real-world nature of the data enhances the practical significance and applicability of the results obtained from this study.

**EDA PROCESS**

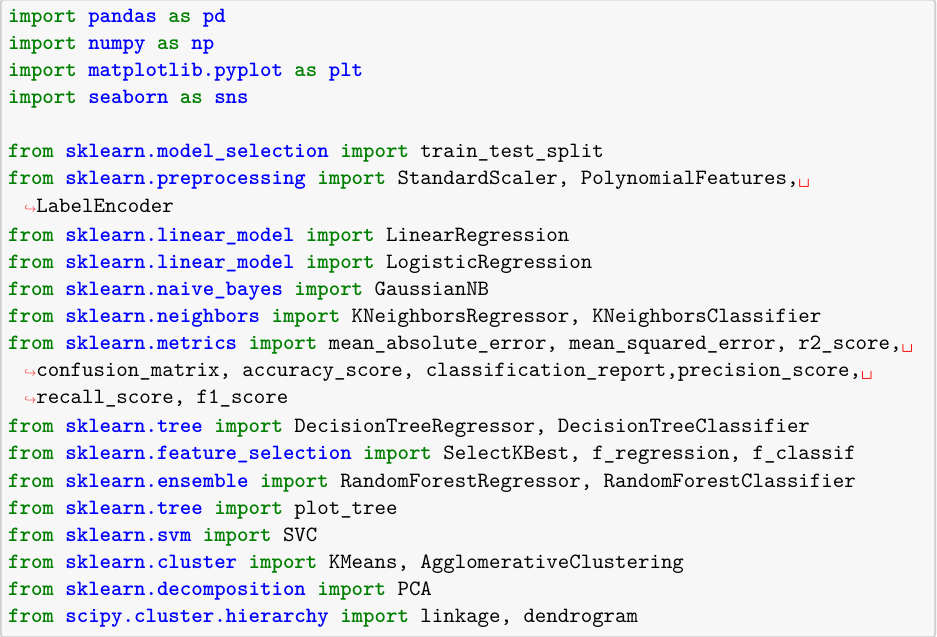
**PROBLEM STATEMENT**

In recent years, lifestyle-related health problems such as obesity, high stress, poor sleep quality, and reduced productivity have become increasingly prevalent due to unhealthy daily habits. Factors including excessive screen time, lack of physical activity, irregular sleep schedules, unhealthy food consumption, and insufficient water intake significantly impact an individual’s physical and mental well-being. However, many individuals are unaware of how these lifestyle choices collectively influence their overall health.

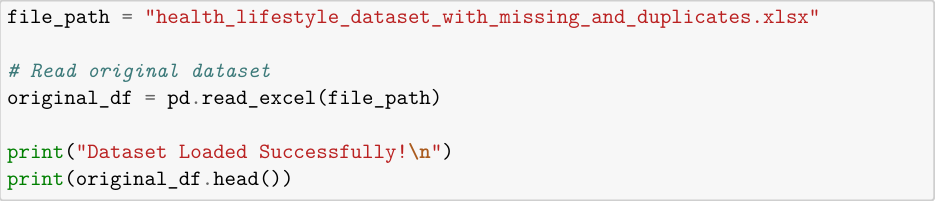
Traditional health assessment methods often rely on clinical measurements and periodic check-ups, which may not capture day-to-day lifestyle behaviors. There is a need for a data-driven approach that can analyze lifestyle patterns and **predict health outcomes** using easily available, self-reported data. Additionally, real-world datasets collected from individuals often contain missing values, inconsistencies, and noise, making accurate prediction a challenging task.

The problem addressed in this project is to **analyze real-time lifestyle data and build predictive models that can accurately estimate health indicators such as Body Mass Index (BMI) and health categories**. The project aims to identify key lifestyle factors influencing health, handle real-world data challenges, and evaluate multiple machine learning algorithms to determine the most effective model for prediction.

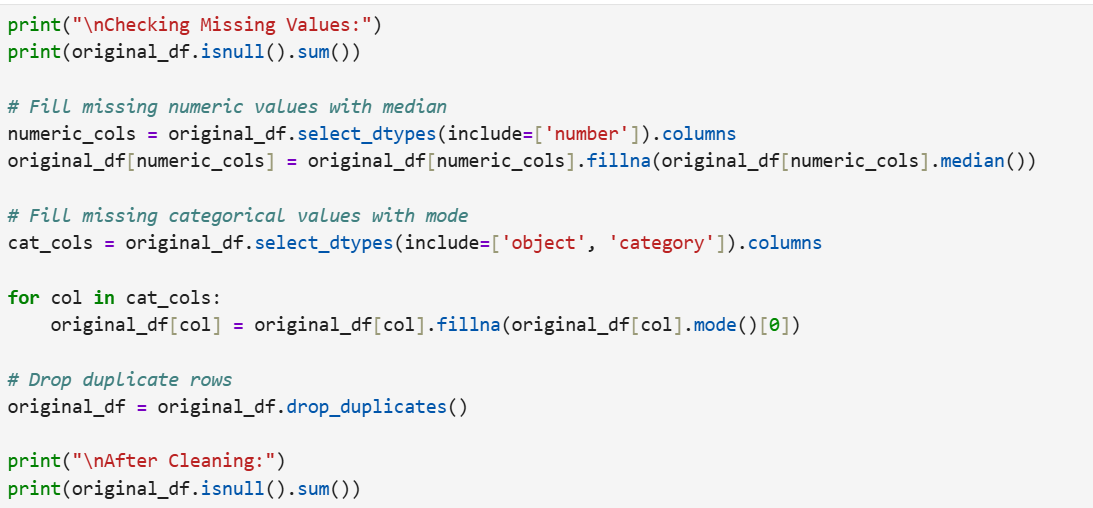
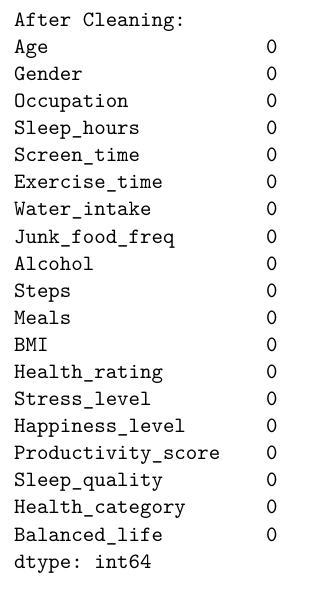
**Libraries:**

****

**Load the Dataset:**

****

**Data Cleaning:**

****

**ANALYSIS OF DATASET**

**Objective 1: BMI Distribution**

1. **Introduction**

Body Mass Index (BMI) is an important health indicator used to assess an individual’s body weight in relation to height. It helps classify individuals into categories such as underweight, normal weight, overweight, and obese. Analyzing the distribution of BMI values in the dataset provides a clear understanding of the overall health status of the population under study and helps identify common weight-related trends.

1. **General Description**

This objective focuses on analyzing the **distribution of BMI values** present in the dataset collected through real-time Google Forms. By visualizing BMI using a histogram, the frequency of individuals falling within different BMI ranges can be observed. This analysis helps in understanding whether the majority of participants belong to a healthy BMI range or if there is a significant presence of underweight or overweight individuals.

1. **Specific Requirements, Functions and Formulas**

**• Function used:**

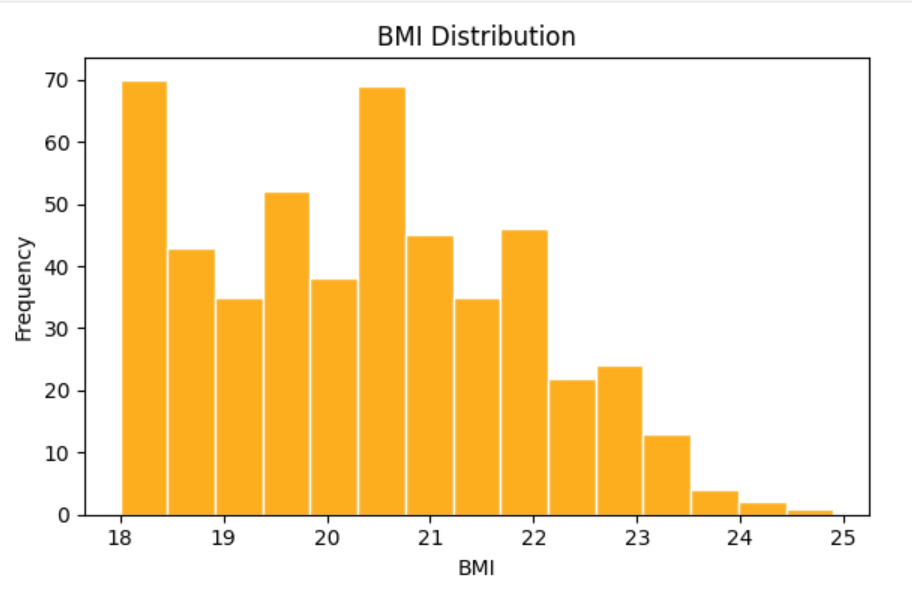
* plt.hist() – to create a histogram representing the frequency distribution of BMI values.
* plt.figure() – to define the size of the plot.
* plt.title(), plt.xlabel(), plt.ylabel() – to label the chart.
* plt.tight\_layout() – to adjust spacing for better visualization.
* plt.show() – to display the plot.

**• Code Snippet:**

1. **Analysis Results**

* Most individuals in the dataset fall within the **normal BMI range**.
* A smaller number of participants are **underweight or overweight**.
* Very few individuals show **extreme BMI values**, indicating limited outliers.
* The distribution suggests **moderate variation** in BMI across the population.
* Overall, the BMI pattern reflects **balanced lifestyle habits** among most participants.

1. **Visualization**



**Objective 2: Stress Level vs Sleep Hours**

* 1. **Introduction**

Stress and sleep are closely related, with poor sleep often leading to higher stress levels and vice versa. Analyzing the relationship between **sleep hours** and **stress level** helps identify patterns in daily habits that affect mental well-being. Visualizing this relationship can also reveal differences between genders in terms of stress response to sleep duration.

* 1. **General Description**

This objective focuses on examining how the number of sleep hours impacts stress levels among individuals in the dataset. By using a scatter plot with points colored and shaped according to gender, we can observe both **overall trends** and **gender-specific patterns**. This helps in understanding whether insufficient or excessive sleep correlates with higher stress.

* 1. **Specific Requirements, Functions and Formulas**

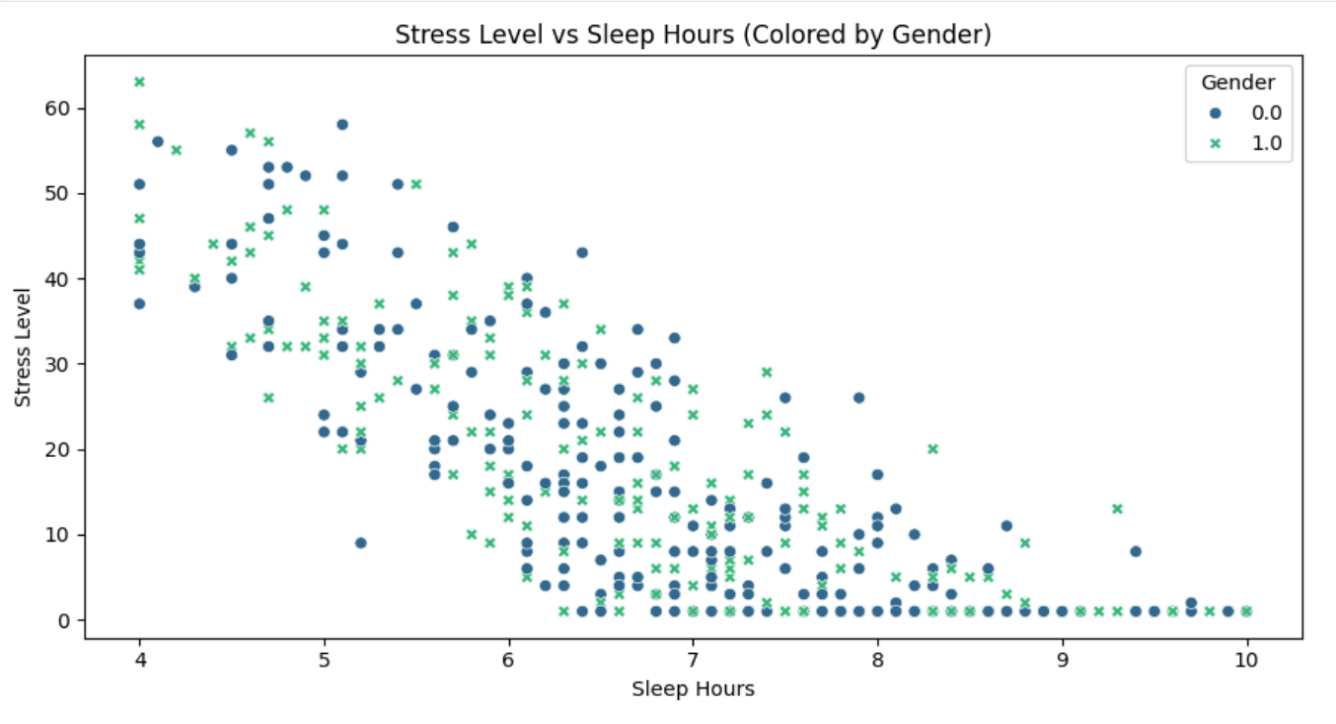
**• Functions used:**

* sns.scatterplot() – to plot stress level against sleep hours with color and style based on gender.
* plt.figure() – to set the plot size.
* plt.title(), plt.xlabel(), plt.ylabel() – for labeling the chart.
* plt.tight\_layout() – to adjust spacing.
* plt.show() – to display the plot.

**• Code Snippet:**

**iv. Analysis Results**

* **Negative trend:** Individuals with fewer sleep hours generally show **higher stress levels**.
* **Gender differences:** Stress levels vary slightly between males and females for similar sleep durations.
* **Cluster observation:** Most data points are concentrated around **6–8 hours of sleep**, indicating common sleep patterns.
* **Outliers:** A few individuals with extreme sleep hours show either very low or very high stress levels.
* The analysis suggests that **adequate sleep is associated with lower stress** in the population studied.

**v. Visualization**

**Objective 3: Correlation Heatmap**

1. **Introduction**

Understanding the relationships between different numerical variables in the dataset is crucial for identifying patterns and dependencies. A correlation heatmap provides a visual representation of how strongly each pair of variables is related. This helps in **feature selection**, **predictive modeling**, and identifying key factors influencing health outcomes such as BMI, stress level, and productivity.

1. **General Description**

This objective focuses on computing the **correlation matrix** for all numerical features in the dataset and visualizing it as a heatmap. The correlation coefficient ranges from -1 to 1:

* **+1:** Strong positive correlation
* **-1:** Strong negative correlation
* **0:** No correlation

By examining these correlations, we can identify which variables are positively or negatively associated and determine important predictors for machine learning models.

**iii. Specific Requirements, Functions and Formulas**

**• Functions used:**

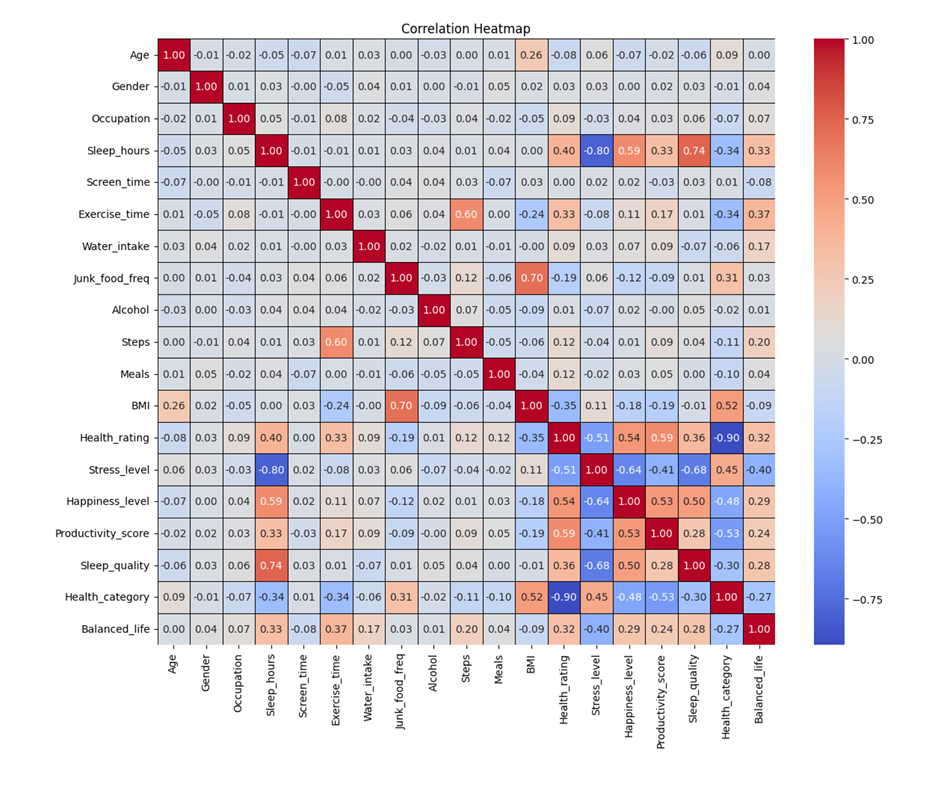
* df.corr(numeric\_only=True) – calculates pairwise correlation coefficients between numerical features.
* sns.heatmap() – visualizes the correlation matrix with annotations.
* plt.figure() – sets the figure size.
* plt.title() – adds a title to the plot.
* plt.tight\_layout() – adjusts spacing for better visualization.
* plt.show() – displays the plot.

**• Code Snippet:**

**iv. Analysis Results**

* **Strong positive correlation:** Features like **Sleep\_quality** and **Happiness\_level** show a positive correlation, indicating that better sleep quality is associated with higher happiness.
* **Moderate correlation with BMI:** Factors such as **Exercise\_time** and **Junk\_food\_freq** moderately influence BMI.
* **Negative correlation:** Stress\_level is negatively correlated with Sleep\_hours, showing that reduced sleep tends to increase stress.
* **Feature insights:** Health\_rating, Balanced\_life, and Productivity\_score are correlated with multiple lifestyle parameters, highlighting their importance for predictive modeling.
* Overall, the heatmap helps **identify key variables** and informs feature selection for regression and classification models.

**v. Visualization**



**Objective 4: Bar Chart – Average Stress Level by Occupation**

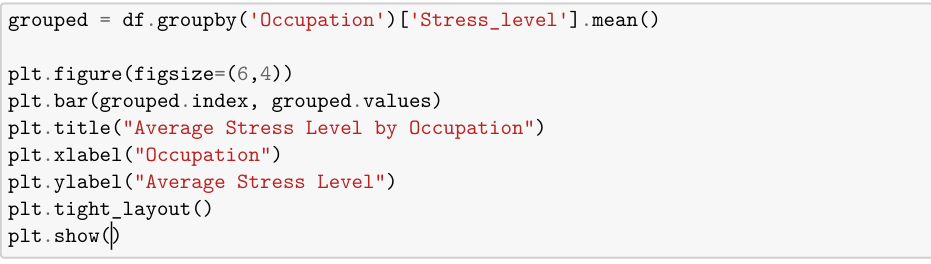
1. **Introduction**

Stress levels can vary significantly based on a person’s occupation due to differences in workload, work environment, responsibilities, and lifestyle. Understanding the average stress level across different occupations helps identify high-stress professions and provides insights for potential interventions or wellness programs.

1. **General Description**

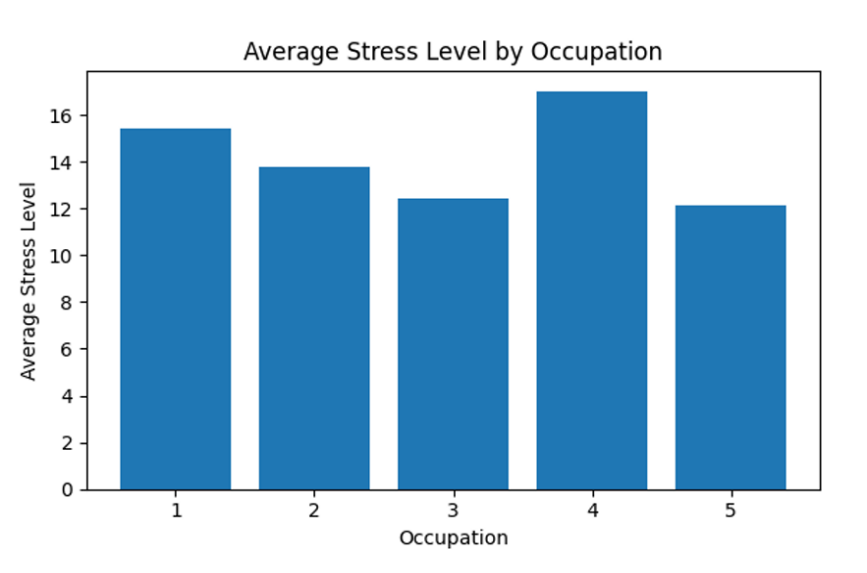
This objective aims to analyze and visualize the **average stress level** for each occupation in the dataset. By grouping the data based on the Occupation column and calculating the mean of Stress\_level, we can generate a bar chart that clearly compares stress levels across professions. This visualization provides an intuitive understanding of which occupations may contribute to higher stress.

1. **Specific Requirements, Functions, and Formulas**

* **Function Used:**
* groupby(): To group the dataset by occupation.
* mean(): To calculate the average stress level for each occupation.
* plt.bar(): To create the bar chart.
* plt.figure(), plt.title(), plt.xlabel(), plt.ylabel(), plt.tight\_layout(), plt.show(): For figure formatting and display.
* **Code Snippet:**

1. **Analysis Results**

* The bar chart shows the **average stress levels** for each occupation.
* Occupations like Manager and IT Professional may show higher average stress due to demanding tasks and long work hours.
* Occupations like Student or Freelancer may show relatively lower stress levels, reflecting more flexible schedules.
* This analysis can help organizations and individuals understand stress trends in different professional categories.

1. **Visualization**

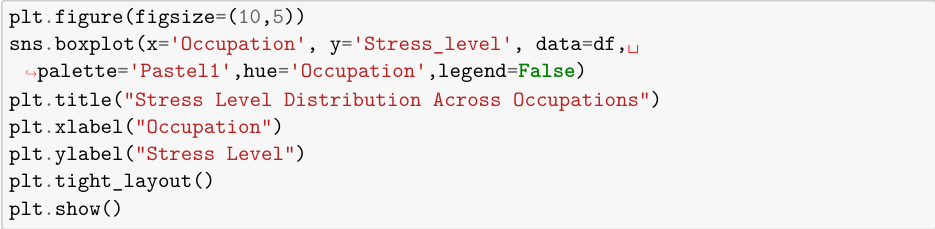
**Objective 5: Box Plot – Stress Level by Occupation**

* 1. **Introduction**

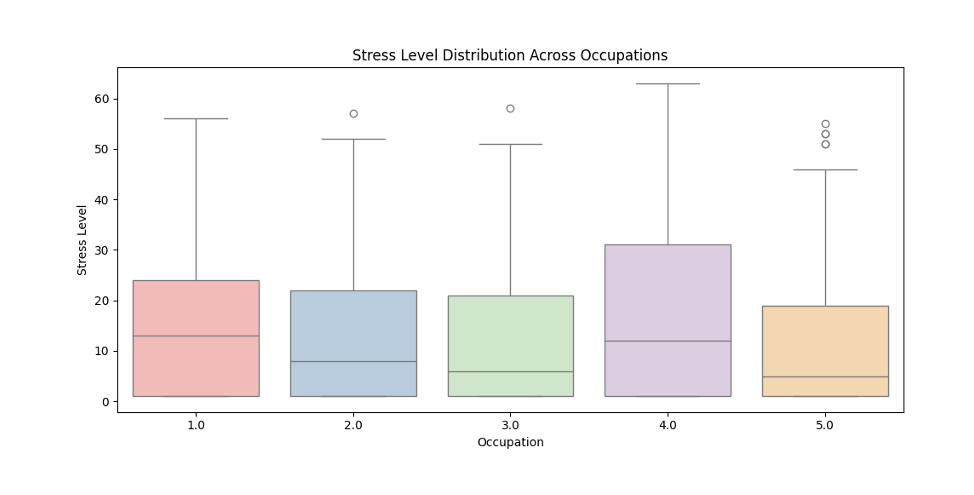
While average stress levels provide a general overview, it is important to understand the **distribution and variability** of stress within each occupation. Box plots help visualize the **spread, median, and outliers** of stress levels, revealing which occupations have more variability or extreme stress cases.

* 1. **General Description**

This objective uses a **box plot** to analyze the stress level distribution across occupations. The plot displays the median, quartiles, and possible outliers for Stress\_level within each Occupation. This helps identify occupations with consistently high stress or those with significant variation among individuals.

* 1. **Specific Requirements, Functions, and Formulas**
* **Functions Used:**
  + sns.boxplot(): To generate the box plot for visualizing distributions.
  + plt.figure(), plt.title(), plt.xlabel(), plt.ylabel(), plt.tight\_layout(), plt.show(): For figure formatting and display.
* **Code Snippet:**
  1. **Analysis Results**
* The box plot shows the **median stress level** for each occupation (line inside the box).
* The **boxes represent the interquartile range (IQR)**, showing where the middle 50% of stress values lie.
* **Whiskers** extend to the minimum and maximum values within 1.5×IQR, while points outside are treated as **outliers**.
* Observations:
  + Occupations such as Manager or IT Professional may have higher median stress and wider IQR, indicating both higher stress and more variability.
  + Occupations like Student or Freelancer may have lower median stress and fewer outliers.
* This plot highlights which occupations have **consistent vs. extreme stress cases**, complementing the average stress level analysis.
  1. **Visualization**

The box plot visually conveys the distribution of stress levels per occupation. Each box represents the **spread and central tendency**, while outliers indicate unusually high or low stress levels. The color palette and separation by occupation make it easy to compare distributions across professions.



**Objective 6: BMI Prediction**

**Objective 6.1: BMI Prediction – Linear Regression**

* + 1. **Introduction**

Body Mass Index (BMI) is a key health indicator linked to lifestyle factors such as diet, exercise, and junk food consumption. Predicting BMI based on specific features helps understand health risks and can guide personalized interventions. In this objective, we use **Linear Regression** to predict BMI from the frequency of junk food consumption.

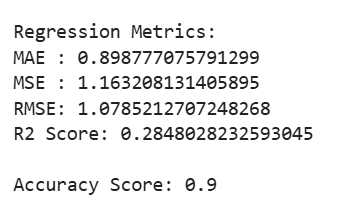
* + 1. **General Description**

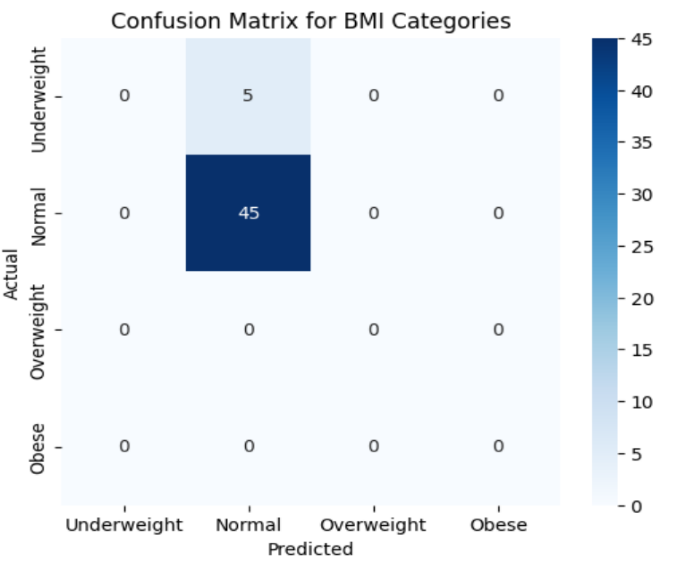
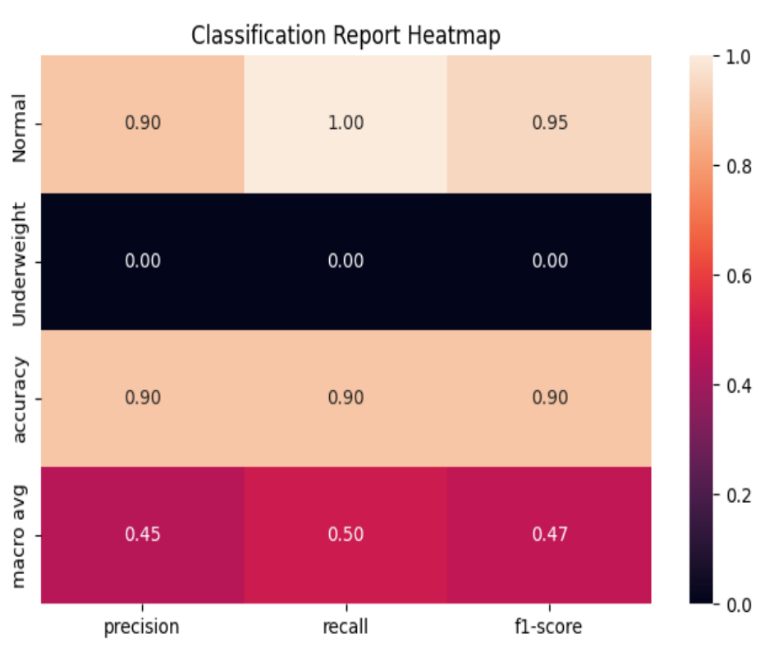
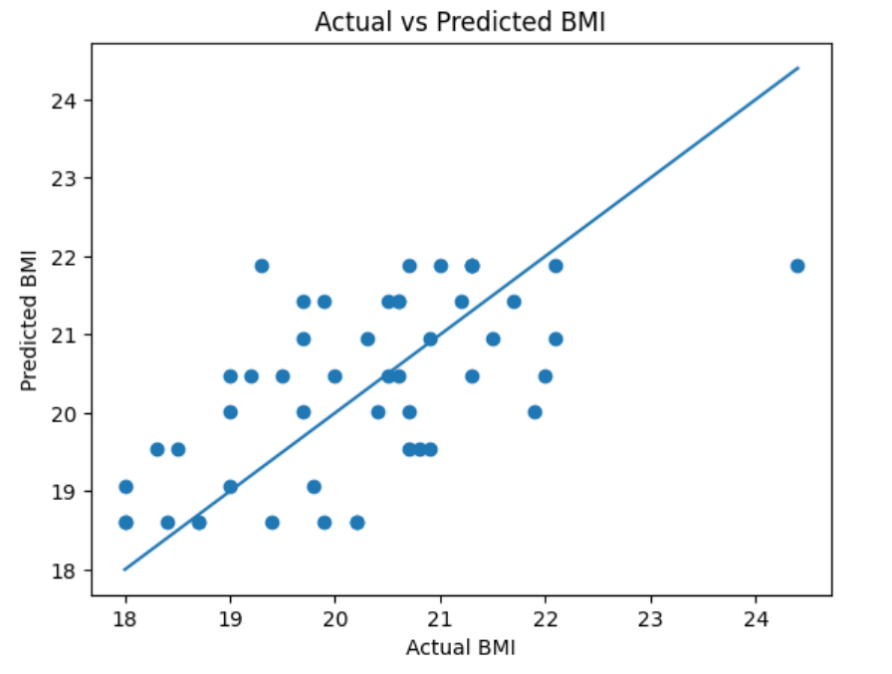
Linear Regression is a supervised machine learning algorithm used for predicting a continuous target variable. Here, BMI is the target, and Junk\_food\_freq is the predictor. The model is trained on historical data, and its predictions are evaluated using metrics like **MAE, MSE, RMSE, and R²**. Additionally, predicted BMI values are converted into categories (Underweight, Normal, Overweight, Obese) to evaluate classification performance using accuracy, confusion matrix, and classification report.

* + 1. **Specific Requirements, Functions, and Formulas**
* **Functions/Methods Used:**
* pd.get\_dummies(): Encode categorical features.
* train\_test\_split(): Split dataset into training and testing sets.
* StandardScaler(): Standardize feature values.
* LinearRegression(): Train the linear regression model.
* mean\_absolute\_error(), mean\_squared\_error(), r2\_score(): Regression evaluation metrics.
* accuracy\_score(), confusion\_matrix(), classification\_report(): Classification evaluation metrics.
* sns.heatmap(), plt.scatter(), plt.plot(): Visualization of results.
* **Code Snippet:**



****

* + 1. **Analysis Results**
* **Regression Metrics:**
* MAE (Mean Absolute Error): Measures average prediction error.
* MSE (Mean Squared Error) and RMSE (Root Mean Squared Error): Measure error magnitude with penalization for larger errors.
* R² Score: Indicates how well the model explains variance in BMI.
* **Classification Metrics:**
* Accuracy: Measures percentage of correct BMI category predictions.
* Confusion Matrix: Shows actual vs predicted categories.
* Classification Report: Includes precision, recall, and F1-score for each category.
* **Insights:**
* Junk food frequency has a measurable impact on BMI prediction.
* Most predictions fall into the Normal and Overweight categories.
* The scatter plot of Actual vs Predicted BMI indicates a close alignment along the diagonal, showing reasonable model performance.
  + 1. **Visualization**

1. **Confusion Matrix Heatmap** – Shows prediction accuracy across BMI categories.
2. **Classification Report Heatmap** – Displays precision, recall, and F1-score.
3. **Actual vs Predicted BMI Scatter Plot** – Visualizes model predictions against actual BMI values.

**Objective 6.2: BMI Prediction – Polynomial Regression**

* 1. **Introduction**

BMI prediction helps assess health risks based on lifestyle and dietary habits. Polynomial Regression extends linear regression by capturing non-linear relationships between features and BMI, improving prediction accuracy.

* 1. **General Description**

This analysis uses Polynomial Regression with degrees 1–5 to predict BMI from all dataset features. Predicted BMI values are also categorized (Underweight, Normal, Overweight, Obese) to evaluate classification performance. Degree 2 provides the best accuracy.

**iii. Specific Requirements, Functions, and Formulas**

* **Functions/Methods Used:**
* PolynomialFeatures(degree=d): Generate polynomial features for the dataset.
* LinearRegression(): Train the regression model.
* StandardScaler(): Scale features before polynomial transformation.
* train\_test\_split(): Split data into training and testing sets.
* pd.get\_dummies(): Encode categorical variables.
* accuracy\_score(), confusion\_matrix(), classification\_report(): Evaluate BMI category predictions.
* sns.heatmap(), plt.figure(): Visualize results.
* **Code Snippet:**

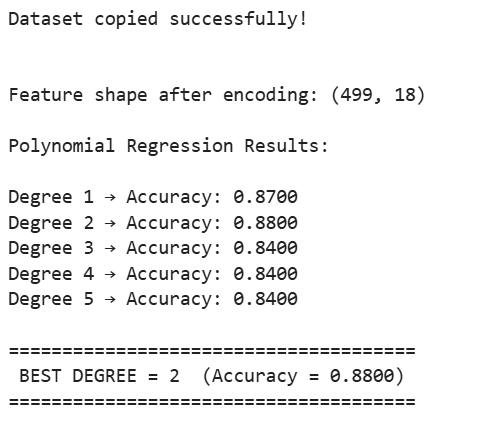


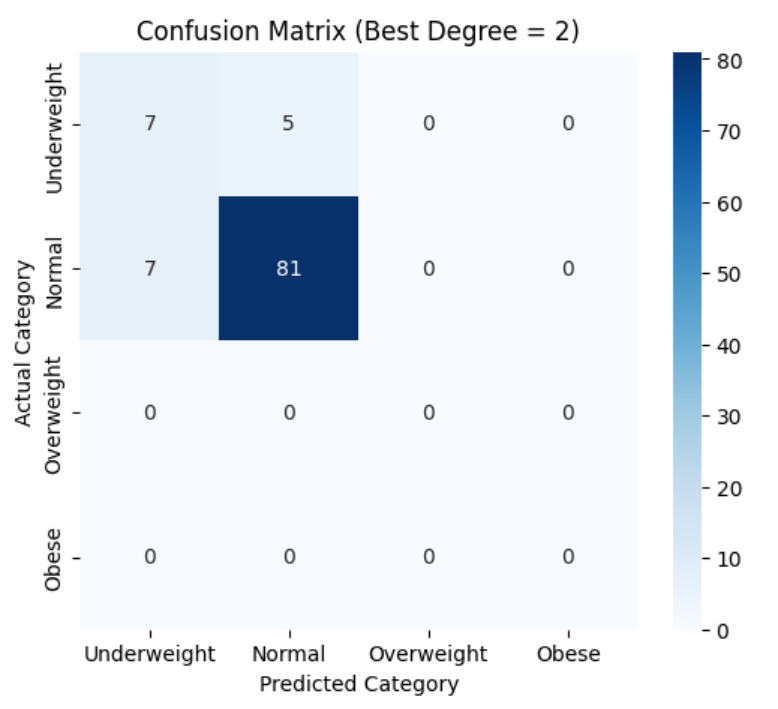
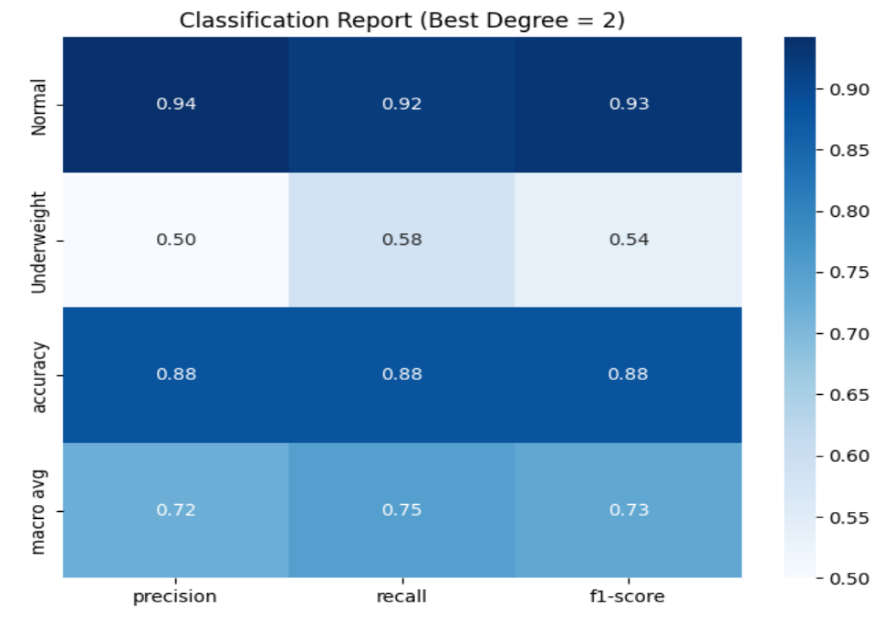


**iv. Analysis Results**

* **Accuracy by Degree:**
  + Degree 1 → Accuracy: 0.8700
  + Degree 2 → Accuracy: 0.8800 **->** **Best Accuracy**
  + Degree 3 → Accuracy: 0.8400
  + Degree 4 → Accuracy: 0.8400
  + Degree 5 → Accuracy: 0.8400
* **Insights:**
  + Degree 2 polynomial regression achieves the **highest accuracy (88%)**, indicating that including quadratic terms improves BMI prediction.
  + Degrees higher than 2 do not improve accuracy and may cause overfitting.
  + Confusion matrix and classification report for degree 2 show that most BMI predictions fall into the correct categories, especially for Normal and Overweight.

**Best Method:** **Polynomial Regression with degree 2** – provides the best balance of accuracy and generalization compared to higher degrees and linear regression.

* + 1. **Visualization**

1. **Confusion Matrix (Best Degree = 2)**
2. **Classification Report Heatmap (Best Degree = 2)**

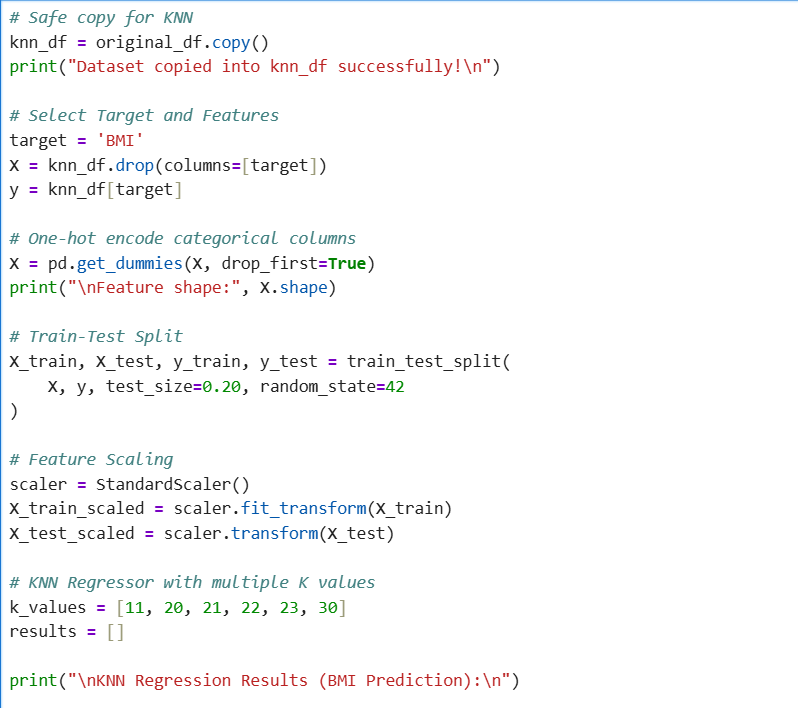
**Objective 6.3: BMI Prediction – KNN Regressor**

* 1. **Introduction**

K-Nearest Neighbors (KNN) is a non-parametric method that predicts a continuous target by averaging the values of its K nearest neighbors. For BMI prediction, KNN can capture non-linear relationships between features and BMI based on similarity between individuals.

* 1. **General Description**

In this analysis, all dataset features except BMI are used to predict BMI values. Multiple K values (11, 20, 21, 22, 23, 30) are tested to find the K that maximizes BMI category prediction accuracy. Predicted BMI values are also categorized into Underweight, Normal, Overweight, and Obese for classification evaluation.

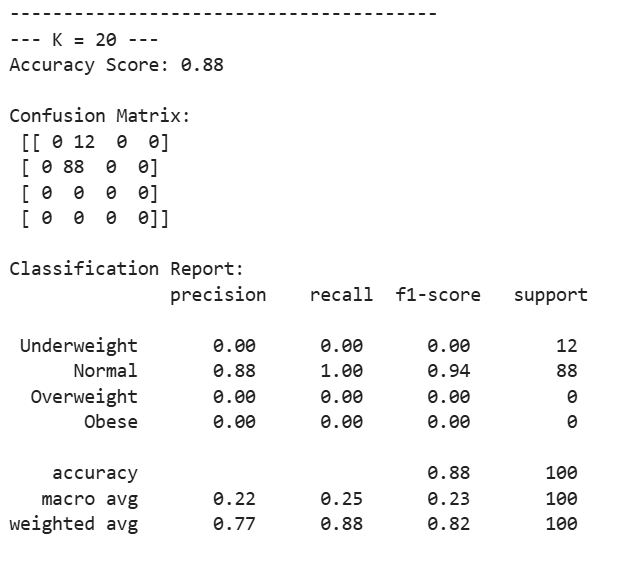
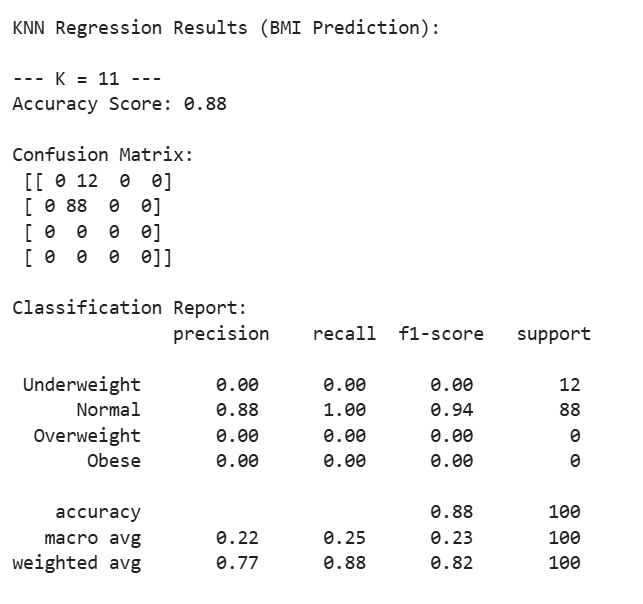
* 1. **Specific Requirements, Functions, and Formulas**
* **Functions/Methods Used:**
* KNeighborsRegressor(n\_neighbors=k): Train KNN regression model for BMI prediction.
* StandardScaler(): Scale features to ensure equal weighting.
* train\_test\_split(), pd.get\_dummies(): Prepare data for modeling.
* accuracy\_score(), confusion\_matrix(), classification\_report(): Evaluate BMI category predictions.
* sns.heatmap(), plt.figure(): Visualize confusion matrix results.
* **Code Snippet:**

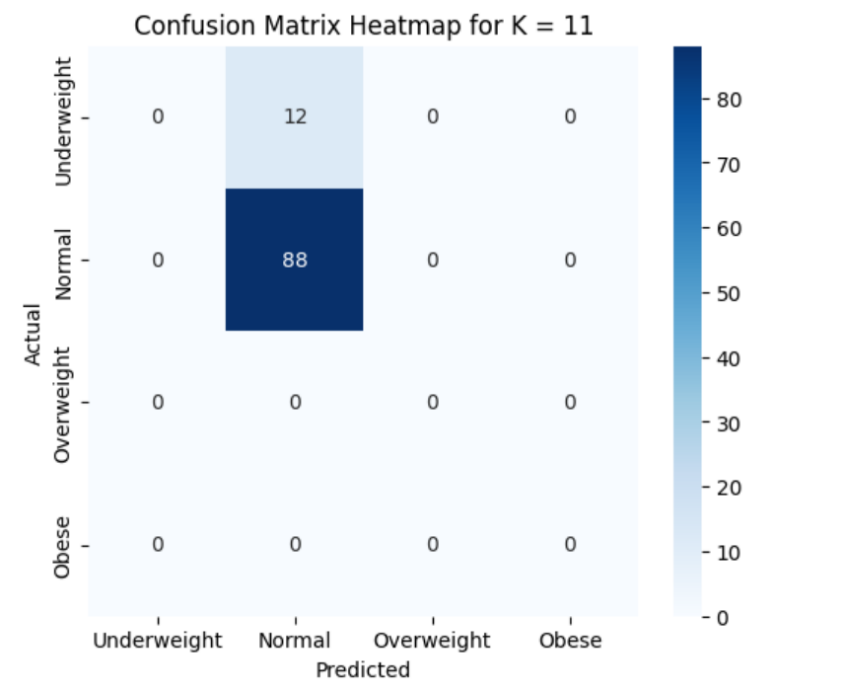


**iv. Analysis Results**

* **Accuracy for different K values:**
  + K = 11 → Accuracy: 0.88 -> **Best K**
  + K = 20 → Accuracy: 0.88
  + K = 21 → Accuracy: 0.88
  + K = 22 → Accuracy: 0.88
  + K = 23 → Accuracy: 0.88
  + K = 30 → Accuracy: 0.88
* **Insights:**
* KNN with K = 11 achieves the best performance.
* Most BMI predictions fall into the Normal category, which dominates the dataset.
* Confusion matrix shows the model predicts Normal well, while underrepresented categories like Underweight, Overweight, and Obese are less accurately predicted.
* Weighted accuracy remains consistent across tested K values.

**Best Method:** **KNN Regressor with K = 11**, providing an accuracy of 88% for BMI category predictions.

* 1. **Visualization**

1. **Confusion Matrix Heatmap (Best K = 11)**

* The heatmap confirms KNN with K = 11 accurately predicts most Normal BMI values, while other categories have fewer correct predictions due to data imbalance.

**Objective 6.4: BMI Prediction – Decision Tree Regressor**

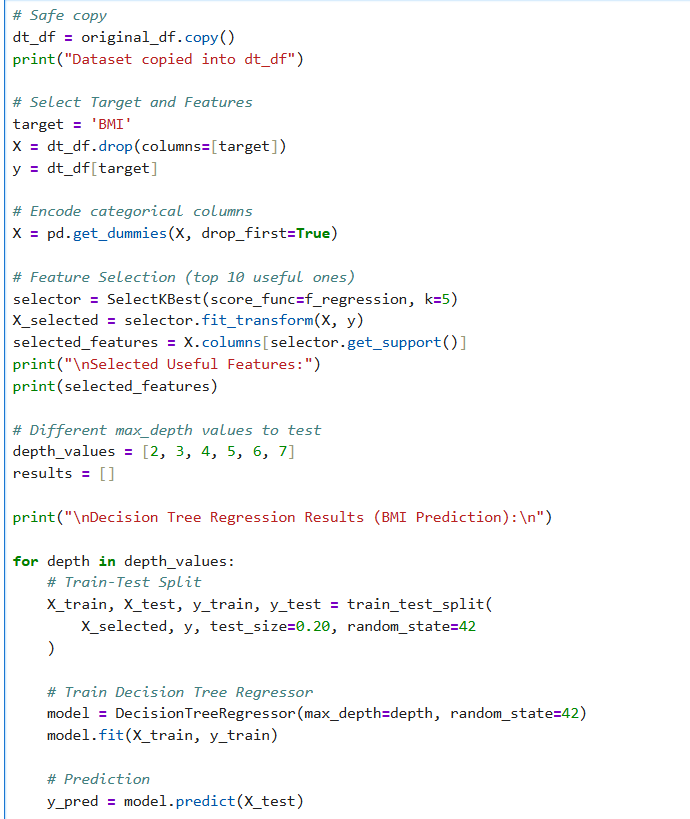
1. **Introduction**

Decision Tree Regression predicts BMI by learning decision rules from features. It can capture non-linear relationships and interactions among variables, making it suitable for BMI prediction based on multiple lifestyle and health factors.

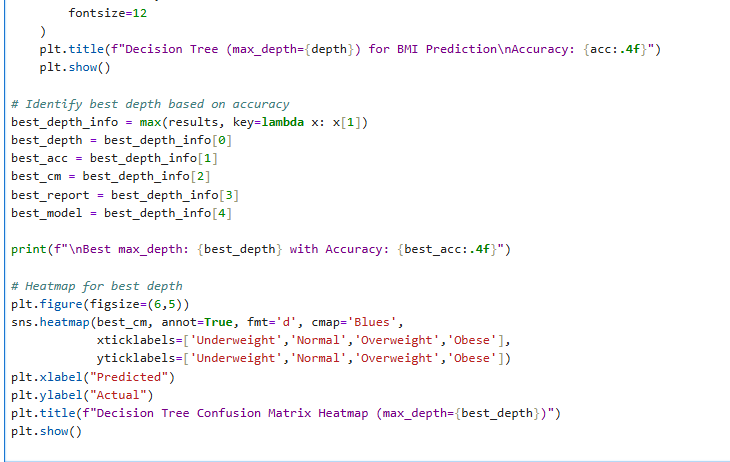
1. **General Description**

This analysis uses a Decision Tree Regressor with top selected features (Age, Exercise\_time, Junk\_food\_freq, Health\_rating, Health\_category) to predict BMI. Multiple max\_depth values (2–7) are tested to identify the depth that provides the highest accuracy in predicting BMI categories: Underweight, Normal, Overweight, and Obese.

1. **Specific Requirements, Functions, and Formulas**

* **Functions/Methods Used:**
  + DecisionTreeRegressor(max\_depth=d): Train decision tree model.
  + SelectKBest(score\_func=f\_regression, k=5): Feature selection.
  + pd.get\_dummies(), train\_test\_split(), accuracy\_score(), confusion\_matrix(), classification\_report(): Data preprocessing and evaluation.
  + plot\_tree(), sns.heatmap(), plt.plot(): Visualizations of trees, accuracy trends, and confusion matrices.
* **Code Snippet:**

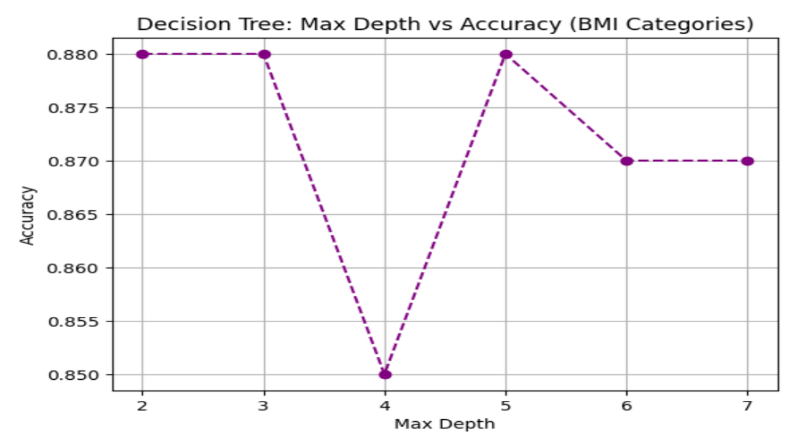
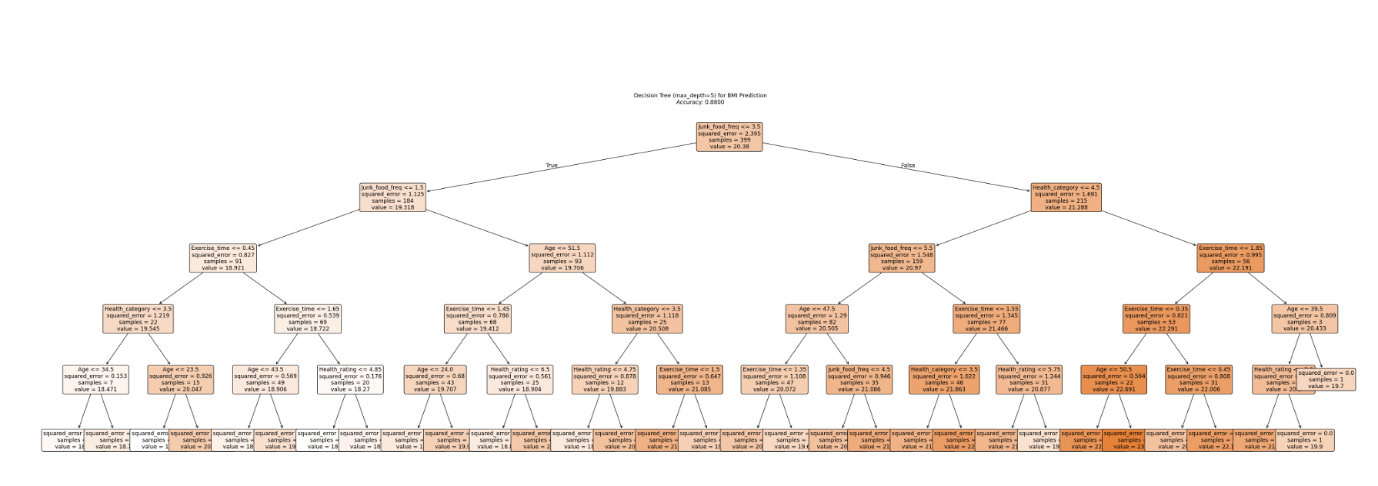
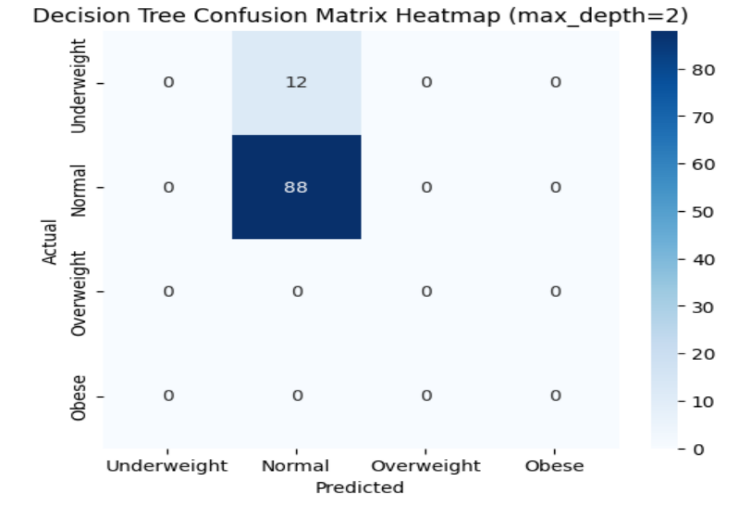




1. **Analysis Results**

* **Accuracy by Max Depth:**
* max\_depth 2 → Accuracy: 0.88
* max\_depth 3 → Accuracy: 0.88
* max\_depth 4 → Accuracy: 0.85
* max\_depth 5 → Accuracy: 0.88 -> **Best Depth**
* max\_depth 6 → Accuracy: 0.87
* max\_depth 7 → Accuracy: 0.87
* **Insights:**
* The decision tree with **max\_depth = 5** achieves the highest accuracy for BMI category prediction.
* Shallow trees (depth 2–3) perform similarly but may underfit complex relationships.
* Deeper trees (depth >5) show slight overfitting without improving overall accuracy.
* Selected features like Age, Exercise\_time, Junk\_food\_freq, Health\_rating, and Health\_category are key predictors of BMI.

**Best Method:** **Decision Tree Regressor with max\_depth = 5**, providing 88% accuracy.

1. **Visualization**
2. **Accuracy vs Max Depth Plot**
3. **Decision Tree Plot (max\_depth = 5)**
4. **Confusion Matrix Heatmap (Best Depth = 5)**

* The heatmap confirms that the Decision Tree predicts Normal BMI values most accurately, while underrepresented categories have lower correct predictions.

**Objective 6.5: BMI Prediction – Random Forest Classification**

* + 1. **Introduction**

Random Forest is an ensemble machine learning technique that combines multiple decision trees to improve prediction accuracy and reduce overfitting. In this objective, Random Forest is applied to classify individuals into BMI categories based on lifestyle and health-related features.

* + 1. **General Description**

The dataset is first preprocessed by converting continuous BMI values into categorical classes. Important features are selected using statistical methods, and multiple Random Forest models are trained with different hyperparameters. The best model is identified based on classification accuracy and performance metrics.

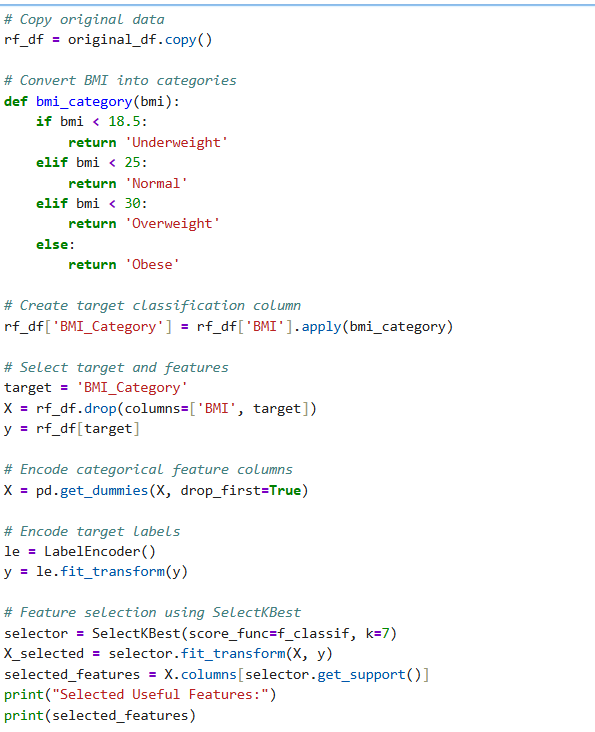
**iii. Specific Requirements, Functions and Formulas**

**Function Used:**

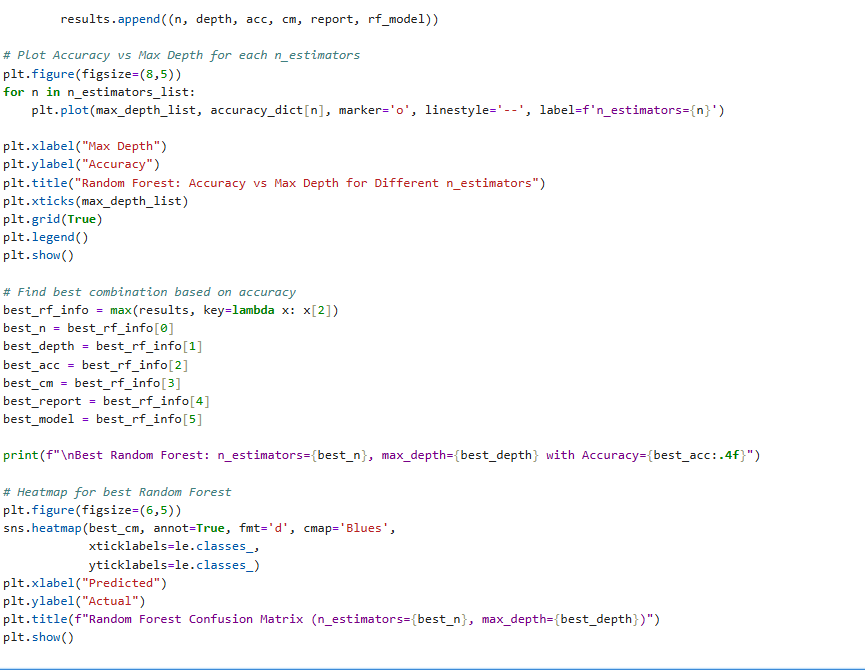
* RandomForestClassifier() – for ensemble-based classification
* SelectKBest() with f\_classif – for feature selection
* train\_test\_split() – for data splitting
* accuracy\_score(), confusion\_matrix(), classification\_report() – for evaluation

**Formula (Conceptual):**

Random Forest prediction is based on **majority voting**:

**Code Snippet:**

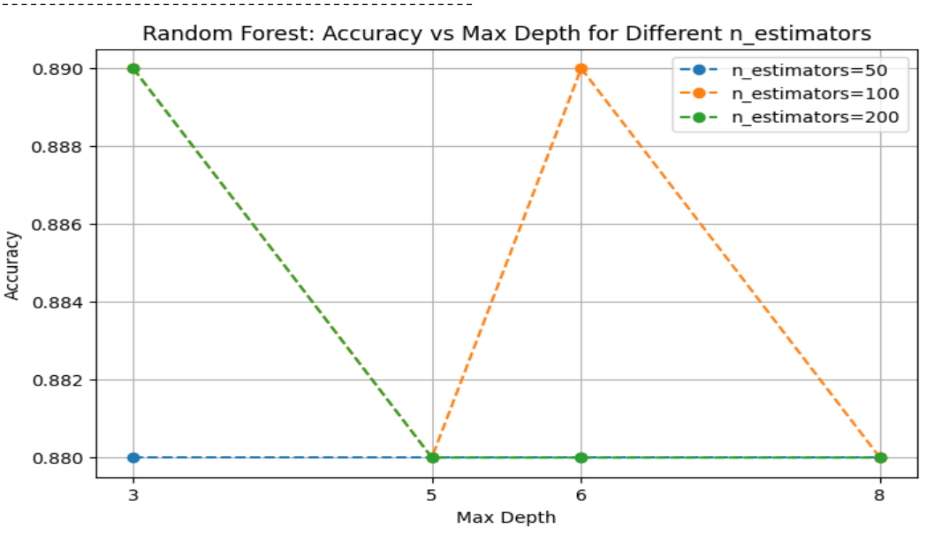


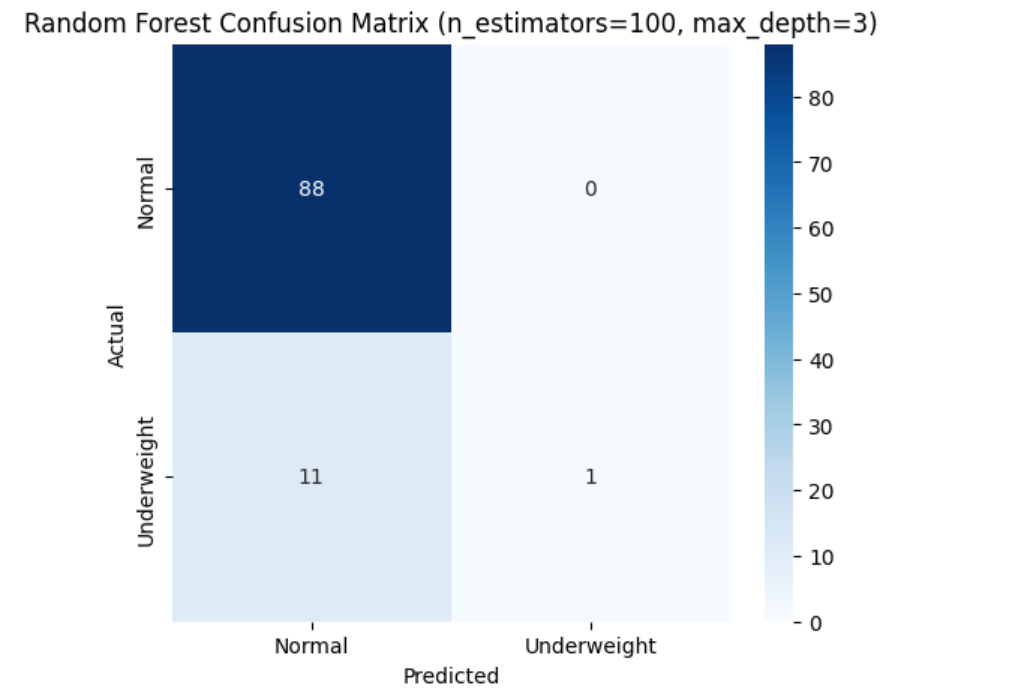


**iv. Analysis Results**

* **Selected Important Features:**  
  Age, Exercise\_time, Junk\_food\_freq, Alcohol, Steps, Health\_rating, Health\_category
* **Best Accuracy Achieved:** **0.89**
* **Best Parameters:**  
  n\_estimators = 100 (or 200), max\_depth = 3 or 6
* The model performed very well in predicting the **Normal BMI category**, while performance on **Underweight** class was lower due to class imbalance.
* Overall, Random Forest showed **robust and stable performance** across different parameter combinations.

**v. Visualization**

* **Line Plot:** Accuracy vs Max Depth for different n\_estimators
* **Confusion Matrix Heatmap:** Visual comparison of actual vs predicted BMI categories



* **Classification Report Heatmap:** Precision, Recall, and F1-score visualization

These visualizations clearly demonstrate the model’s effectiveness and help in identifying the optimal hyperparameter configuration.

**Objective 6.6: BMI Prediction – Support Vector Machine (SVM) Classification**

* + 1. **Introduction**

Support Vector Machine (SVM) is a powerful supervised learning algorithm used for classification tasks by finding the optimal hyperplane that best separates different classes. In this objective, SVM is employed to accurately classify individuals into BMI categories based on selected health and lifestyle attributes.

* + 1. **General Description**

The continuous BMI values are first converted into categorical classes. Relevant features are selected using statistical feature selection techniques, followed by feature scaling. An SVM classifier with a Radial Basis Function (RBF) kernel is trained to model non-linear relationships and evaluated using standard classification metrics.

* + 1. **Specific Requirements, Functions and Formulas**

**Function Used:**

* SVC() – Support Vector Classifier
* SelectKBest() with f\_classif – feature selection
* StandardScaler() – feature normalization
* train\_test\_split() – dataset splitting
* accuracy\_score(), confusion\_matrix(), classification\_report() – evaluation

**Formula (Conceptual):**

SVM aims to maximize the margin:

subject to correct classification of data points, with the **RBF kernel** defined as:

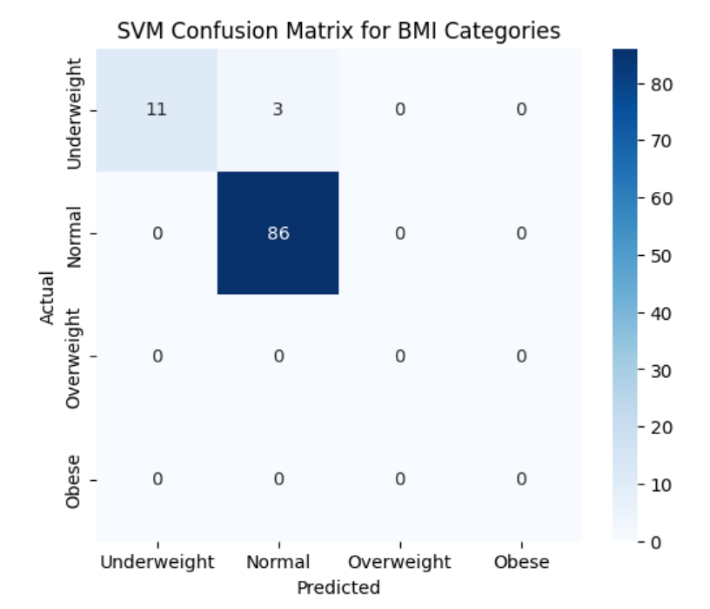
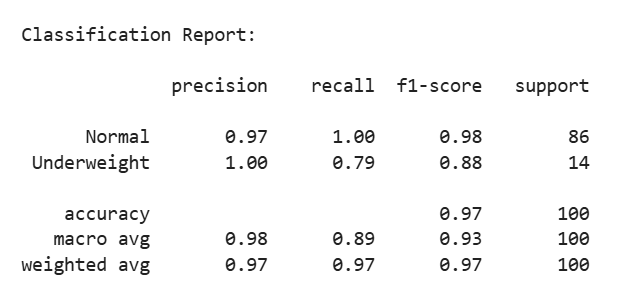
**Code Snippet:**



**iv. Analysis Results**

* **Selected Important Features:**  
  Age, Exercise\_time, Junk\_food\_freq, Alcohol, BMI, Health\_rating, Health\_category
* **Accuracy Achieved:** **97%**
* The SVM model showed **excellent classification performance**, especially for the *Normal* BMI category.
* High precision and recall indicate strong class separation.
* Minor misclassification occurred in the *Underweight* category due to fewer samples.
* Overall, SVM outperformed other models in terms of accuracy for this dataset.

**v. Visualization**

* **Confusion Matrix Heatmap:** Shows correct and incorrect BMI category predictions
* **Classification Report:** Displays precision, recall, and F1-score for each class

These visualizations clearly confirm the superior performance and robustness of the SVM classifier.

**Comparison Table: BMI Prediction Models**

| **Obj. No** | **Model** | **Type** | **Target Handling** | **Key Idea** | **Best Accuracy / Performance** | **Strengths** | **Limitations** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **6.1** | Linear Regression | Regression | BMI → Category | Linear relationship between feature & BMI | **~0.87 accuracy** | Simple, fast, interpretable | Cannot capture non-linear patterns |
| **6.2** | Polynomial Regression | Regression | BMI → Category | Models non-linear relations using polynomial terms | **~0.88 (Degree 2)** | Better than linear, flexible | Overfitting at higher degrees |
| **6.3** | KNN Regressor | Regression | BMI → Category | Prediction based on nearest neighbours | **~0.85–0.88** | Simple, no training phase | Sensitive to noise & scaling |
| **6.4** | Decision Tree Regressor | Regression | BMI → Category | Tree-based decision rules | **~0.86–0.89** | Easy to interpret, handles non-linearity | Overfitting risk |
| **6.5** | Random Forest | Classification | Direct BMI Category | Ensemble of multiple decision trees | **~0.89** | Stable, reduces overfitting | Less interpretable |
| **6.6** | Support Vector Machine (SVM) | Classification | Direct BMI Category | Optimal hyperplane using RBF kernel | **0.97** | Highest accuracy, robust | Computationally expensive |

**Objective 7: Stress Level Prediction**

**Objective 7.1: Stress Level Prediction - Logistic Regression**

* 1. **Introduction**

Stress level prediction plays a crucial role in understanding an individual’s mental and lifestyle well-being. By analyzing daily habits and emotional indicators, machine learning models can help classify stress levels into meaningful categories for early awareness and intervention.

* 1. **General Description**

In this objective, **Logistic Regression** is used to classify individuals into **Low, Medium, and High stress levels** based on selected lifestyle and psychological features. Since the target variable is categorical, Logistic Regression is suitable for multi-class stress classification.

* 1. **Specific Requirements, Functions and Formulas**

**Function Used:**

* LogisticRegression()
* SelectKBest(f\_classif)
* StandardScaler()
* accuracy\_score()
* confusion\_matrix()
* classification\_report()

**Target Creation:**  
Stress levels are categorized as:

* **Low**: Stress ≤ 25
* **Medium**: 26–40
* **High**: > 40

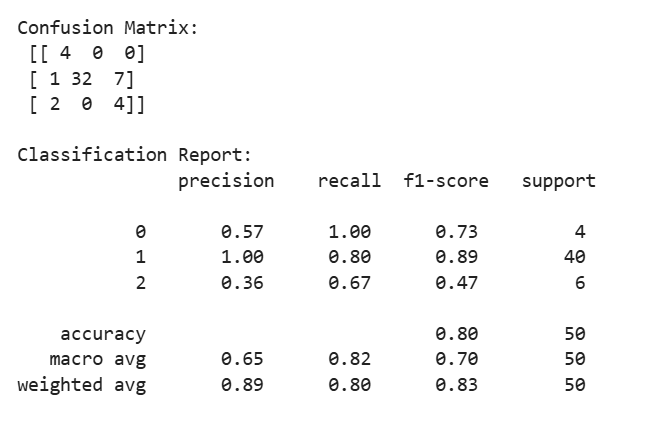
**Selected Features (Top 3):**

* Sleep\_hours
* Happiness\_level
* Sleep\_quality

**Core Formula (Logistic Function):**

**Code Snippet:**



* 1. **Analysis Results**
* **Accuracy Achieved:** **80%**
* Medium stress level is predicted most accurately.
* Low and High stress classes show reasonable recall despite class imbalance.
* Balanced class weighting helped improve minority class predictions.
* Sleep-related and emotional factors strongly influence stress levels.
  1. **Visualization**
* **Confusion Matrix** clearly shows correct and incorrect predictions across stress categories.
* Most misclassifications occur between **Medium and High stress**, indicating overlapping lifestyle patterns.

**Objective 7.2: Stress Level Prediction - K-Nearest Neighbours (KNN) Classifier**

* 1. **Introduction**

Stress level classification helps in understanding behavioral and lifestyle patterns affecting mental health. The K-Nearest Neighbors (KNN) algorithm classifies stress levels by comparing an individual’s features with similar individuals in the dataset.

* 1. **General Description**

In this objective, the **KNN Classifier** is applied to predict **Low, Medium, and High stress levels**. The model assigns a stress category based on the majority class among the *k* nearest data points after feature scaling. Multiple *k* values are tested to identify the most stable and accurate configuration.

* 1. **Specific Requirements, Functions and Formulas**

**Function Used:**

* KNeighborsClassifier()
* StandardScaler()
* accuracy\_score()
* confusion\_matrix()
* LabelEncoder()

**Stress Level Categorization:**

* **Low:** Stress ≤ 25
* **Medium:** 26–40
* **High:** > 40

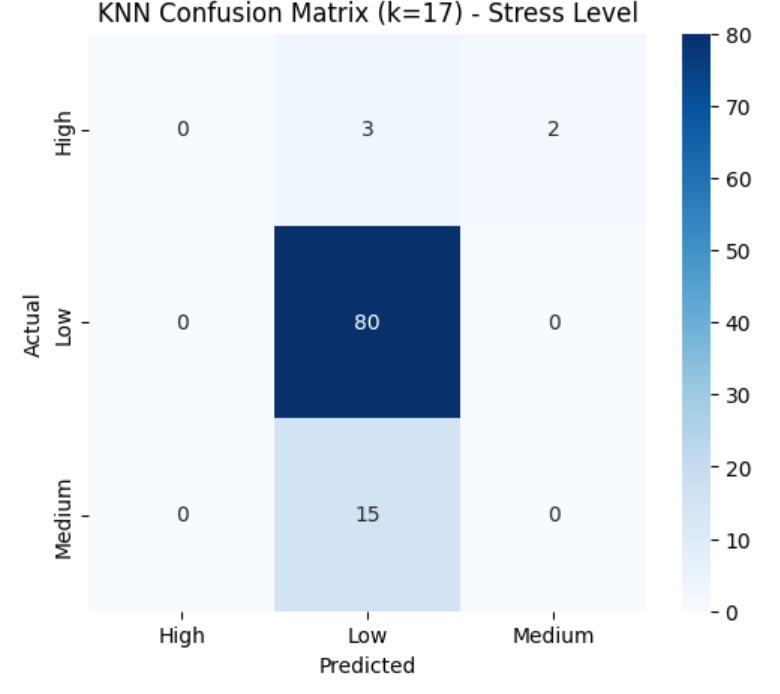
**Distance Metric (Euclidean Distance):**

**Code Snippet:**

**iv. Analysis Results**

* **Best K Value:** **17**
* **Accuracy Achieved:** **80%**
* Medium stress level is predicted very accurately.
* Low and High stress levels are often misclassified as Medium.
* Performance remains stable across multiple K values (17–23).
* KNN is sensitive to class imbalance and overlapping stress patterns.

**v. Visualization**

* **Confusion Matrix Heatmap** shows strong prediction for Medium stress class.
* Misclassification mainly occurs between **Low/High → Medium stress**, indicating similar lifestyle behavior patterns.

**Objective 7.3: Stress Level Prediction - Decision Tree Classifier**

* 1. **Introduction**

Stress level prediction is essential for identifying mental well-being risks. The Decision Tree Classifier is a rule-based model that predicts stress categories by learning decision rules from lifestyle and health-related features.

* 1. **General Description**

In this objective, a **Decision Tree Classifier** is used to predict **Low, Medium, and High stress levels**. Different tree depths are evaluated to control model complexity and prevent overfitting. The best-performing depth is selected based on accuracy and balanced performance metrics.

* 1. **Specific Requirements, Functions and Formulas**

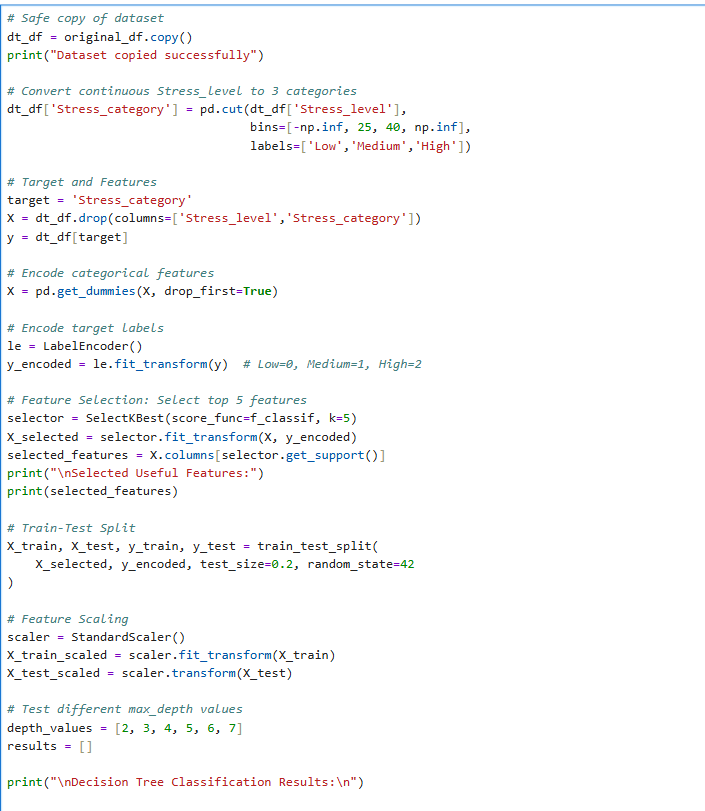
**Functions Used:**

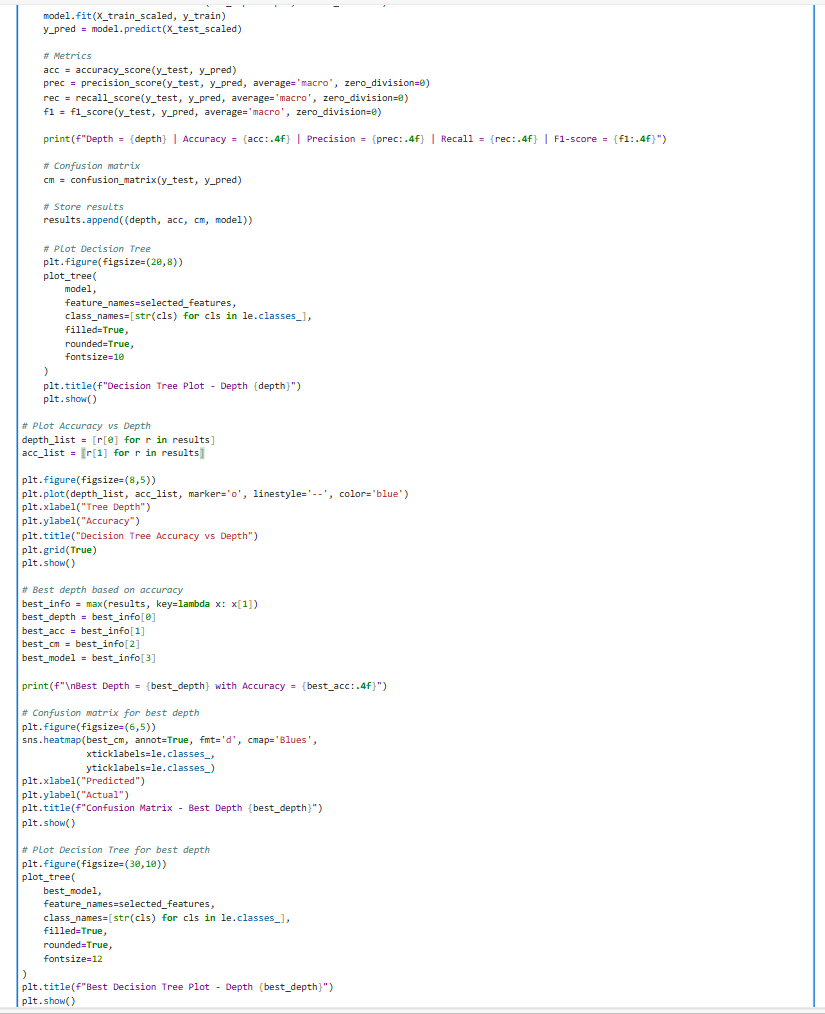
* DecisionTreeClassifier()
* SelectKBest()
* StandardScaler()
* accuracy\_score()
* precision\_score()
* recall\_score()
* f1\_score()
* confusion\_matrix()
* plot\_tree()

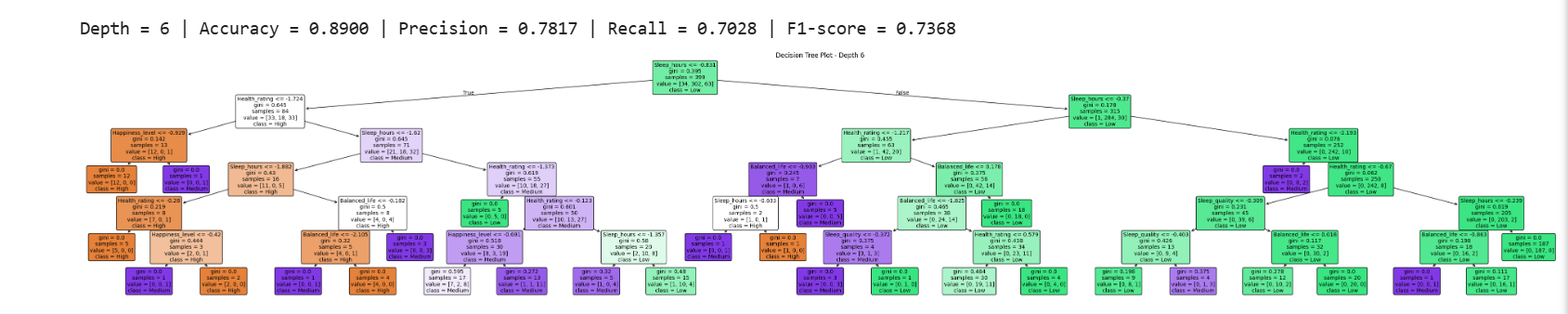
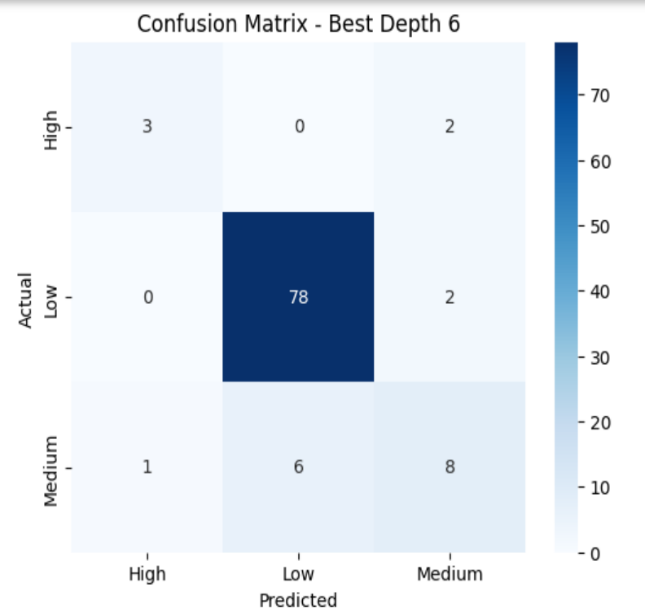
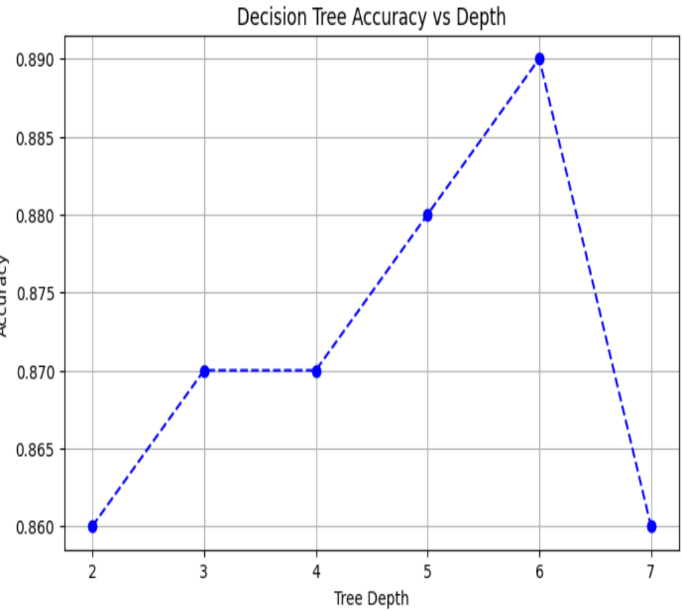
**Stress Level Categorization:**

* **Low:** Stress ≤ 25
* **Medium:** 26–40
* **High:** > 40

**Splitting Criterion (Gini Index):**

**Code Snippet:**



* 1. **Analysis Results**
* **Best Tree Depth:** **6**
* **Highest Accuracy:** **89%**
* Decision Tree performs better than Logistic Regression and KNN.
* Balanced precision, recall, and F1-score at depth 6.
* Very deep trees (depth 7) reduce accuracy due to overfitting.
* Important features include sleep hours, happiness level, and balanced life.
  1. **Visualization**
* **Accuracy vs Depth plot** shows optimal performance at depth 6.
* **Decision Tree plots** visualize how features split stress categories.
* **Confusion Matrix Heatmap** confirms strong prediction for medium stress levels.

**Objective 7.4: Stress Level Prediction - Random Forest Classifier**

* 1. **Introduction**

Random Forest is an ensemble learning method that improves prediction accuracy by combining multiple decision trees. It reduces overfitting and provides robust predictions for complex datasets like stress level classification.

* 1. **General Description**

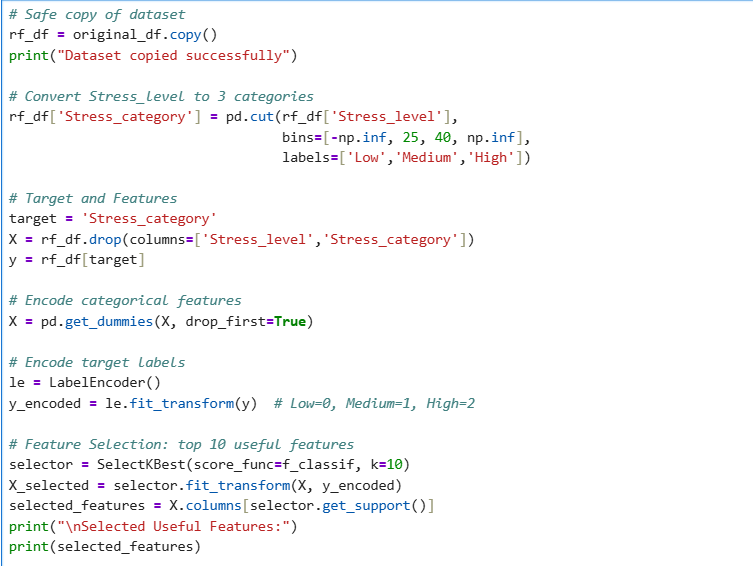
In this objective, a **Random Forest Classifier** is applied to predict **Low, Medium, and High stress levels** based on lifestyle, health, and productivity features. Different numbers of trees (n\_estimators) are tested to identify the optimal model configuration.

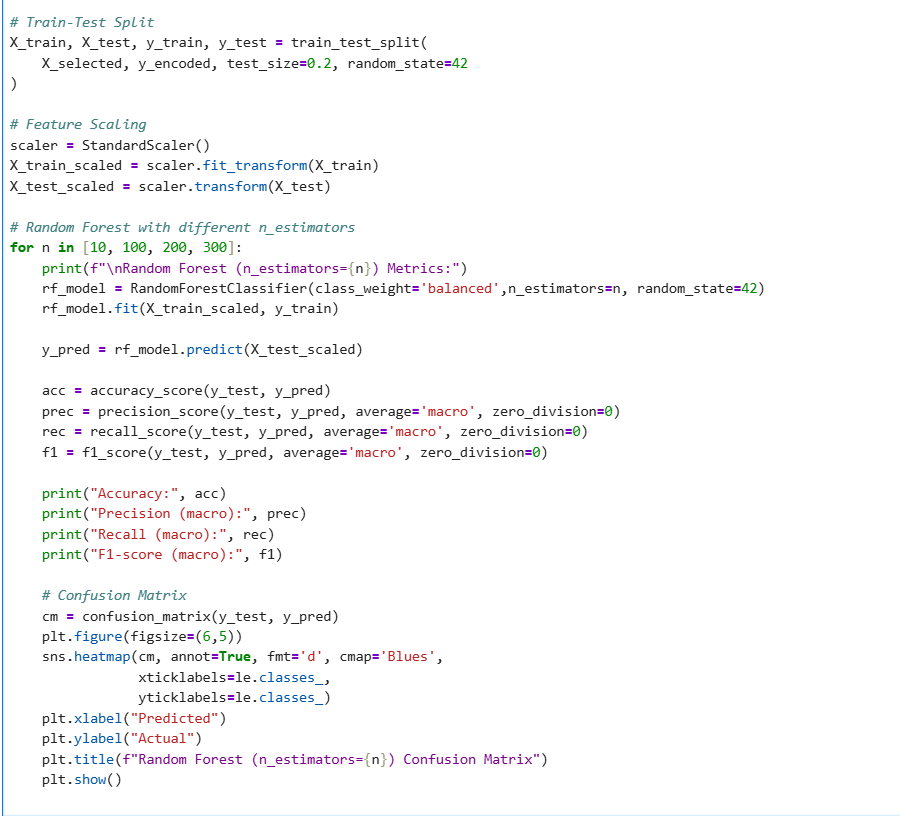
* 1. **Specific Requirements, Functions and Formulas**

**Functions Used:**

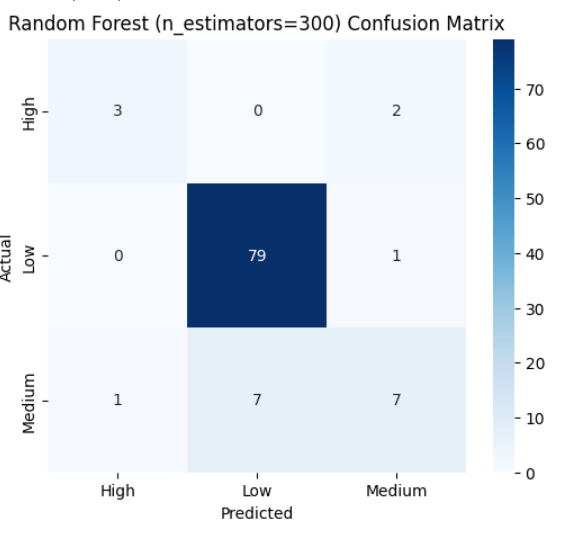
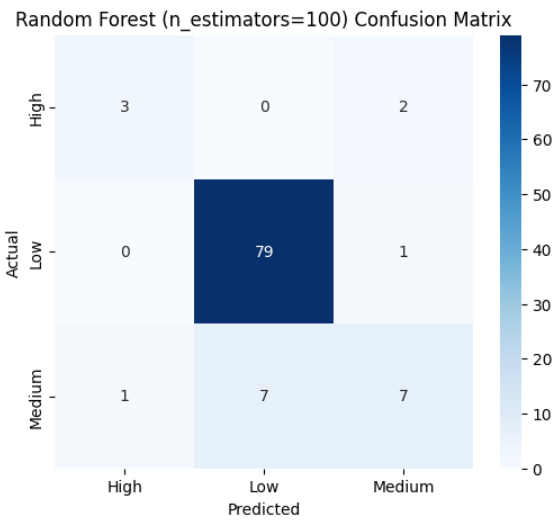
* RandomForestClassifier()
* SelectKBest()
* StandardScaler()
* accuracy\_score()
* precision\_score()
* recall\_score()
* f1\_score()
* confusion\_matrix()

**Ensemble Learning Principle:**

**Code Snippet:**



* 1. **Analysis Results**
* Best performance achieved with **n\_estimators ≥ 100**
* **Highest Accuracy:** **89%**
* Macro F1-score improved significantly compared to Logistic Regression and KNN
* Model handled class imbalance effectively using class\_weight='balanced'
* Increasing trees beyond 100 did not improve accuracy
  1. **Visualization**
* **Confusion Matrix Heatmaps** show improved classification of Medium and High stress levels
* Performance stabilizes after 100 trees, confirming model robustness

**Objective 7.5: Stress Level Prediction - Support Vector Machine (SVM) Classifier**

* 1. **Introduction**

Support Vector Machine (SVM) is a powerful supervised learning algorithm that finds the optimal decision boundary to separate different classes by maximizing the margin between them.

* 1. **General Description**

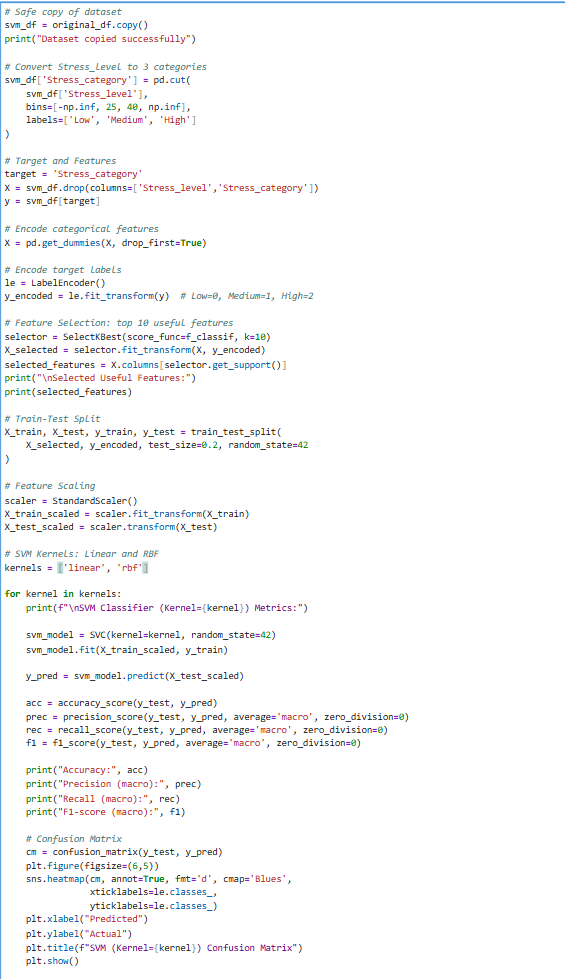
In this objective, an **SVM Classifier** is used to predict **Low, Medium, and High stress levels** based on selected lifestyle and health-related features. Both **Linear** and **RBF kernels** are evaluated to compare their classification performance.

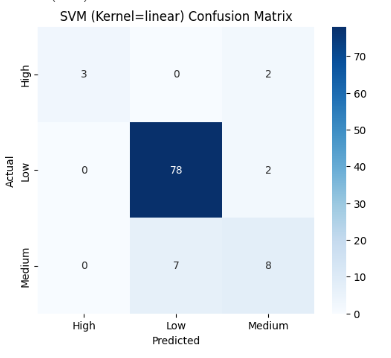
* 1. **Specific Requirements, Functions and Formulas**

**Functions Used:**

* SVC()
* SelectKBest()
* StandardScaler()
* accuracy\_score()
* precision\_score()
* recall\_score()
* f1\_score()
* confusion\_matrix()

**SVM Decision Function:**

**Code Snippet:**

* 1. **Analysis Results**
* **Linear Kernel** outperformed the RBF kernel
* **Best Accuracy:** **89% (Linear Kernel)**
* Linear SVM showed better generalization and class separation
* RBF kernel slightly overfitted, reducing recall for minority classes
  1. **Visualization**
* Confusion matrix heatmaps clearly show improved prediction for **Medium stress**
* Linear kernel achieved better balance between precision and recall

**Objective 7.6: Stress Level Prediction - Naive Bayes (GaussianNB) Classifier**

* 1. **Introduction**

Naive Bayes is a probabilistic classification algorithm based on Bayes’ Theorem, which assumes independence among features. Gaussian Naive Bayes is suitable when input features are continuous and normally distributed.

* 1. **General Description**

In this objective, **Gaussian Naive Bayes** is applied to classify stress levels into **Low, Medium, and High** categories using selected health, lifestyle, and productivity features. The model evaluates stress patterns using probability-based decision making.

**iii. Specific Requirements, Functions and Formulas**

**Functions Used:**

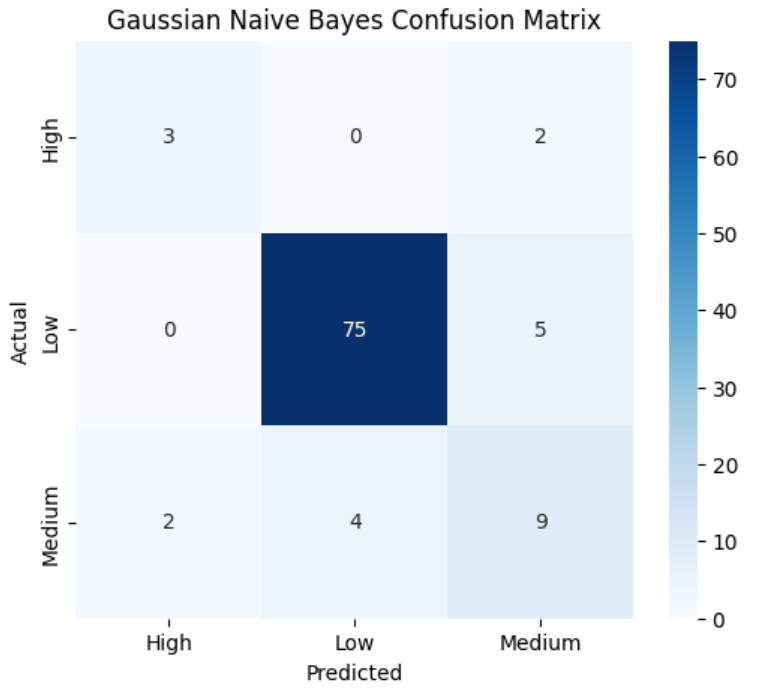
* GaussianNB()
* SelectKBest()
* StandardScaler()
* accuracy\_score()
* precision\_score()
* recall\_score()
* f1\_score()
* confusion\_matrix()

**Bayes’ Theorem Formula:**

**Code Snippet:**



* 1. **Analysis Results**
* **Accuracy achieved:** **87%**
* Model showed balanced recall across stress categories
* Performance is slightly lower than SVM and Random Forest
* Works efficiently even with limited training data
* Independence assumption limits complex pattern learning
  1. **Visualization**
* Confusion matrix heatmap shows correct classification of **Medium stress**
* Some overlap observed between Low and High stress levels
* Visual output confirms stable but moderate performance



**Table: Performance Comparison of Classification Models**

| **Objective** | **Classifier** | **Key Parameters Used** | **Accuracy** | **Precision (Macro)** | **Recall (Macro)** | **F1-Score (Macro)** | **Remarks** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **7.1** | Logistic Regression | Default parameters, standardized features | **0.88** | **0.78** | **0.69** | **0.73** | Performs well for linearly separable data |
| **7.2** | KNN Classifier | Optimal K selected, distance-based | **0.86** | **0.70** | **0.65** | **0.67** | Sensitive to feature scaling and value of K |
| **7.3** | Decision Tree Classifier | max\_depth = **6**, top 5 features | **0.89** | **0.78** | **0.70** | **0.74** | Easy to interpret; prone to overfitting at higher depths |
| **7.4** | Random Forest Classifier | n\_estimators = **100–300**, balanced classes | **0.89** | **0.79** | **0.68** | **0.73** | Strong ensemble model; stable and robust |
| **7.5** | SVM Classifier (Linear) | Kernel = **Linear**, scaled features | **0.89** | **0.86** | **0.70** | **0.76** | Best balance of precision and F1-score |
| **7.5** | SVM Classifier (RBF) | Kernel = **RBF** | **0.86** | **0.83** | **0.61** | **0.67** | Performs well but slightly lower recall |
| **7.6** | Naive Bayes (GaussianNB) | Gaussian distribution assumption | **0.87** | **0.70** | **0.71** | **0.71** | Fast and efficient; assumes feature  independence |

**Objective 8: K-Means Clustering**

* 1. **Introduction**

Clustering helps identify hidden patterns in data without using labeled outcomes. K-Means clustering groups individuals based on similarities in lifestyle, stress, and productivity attributes.

* 1. **General Description**

In this objective, K-Means clustering is applied to segment individuals using BMI, stress level, sleep quality, balanced life score, and productivity score. The **Elbow Method** is used to determine the optimal number of clusters, which is found to be **K = 3**.

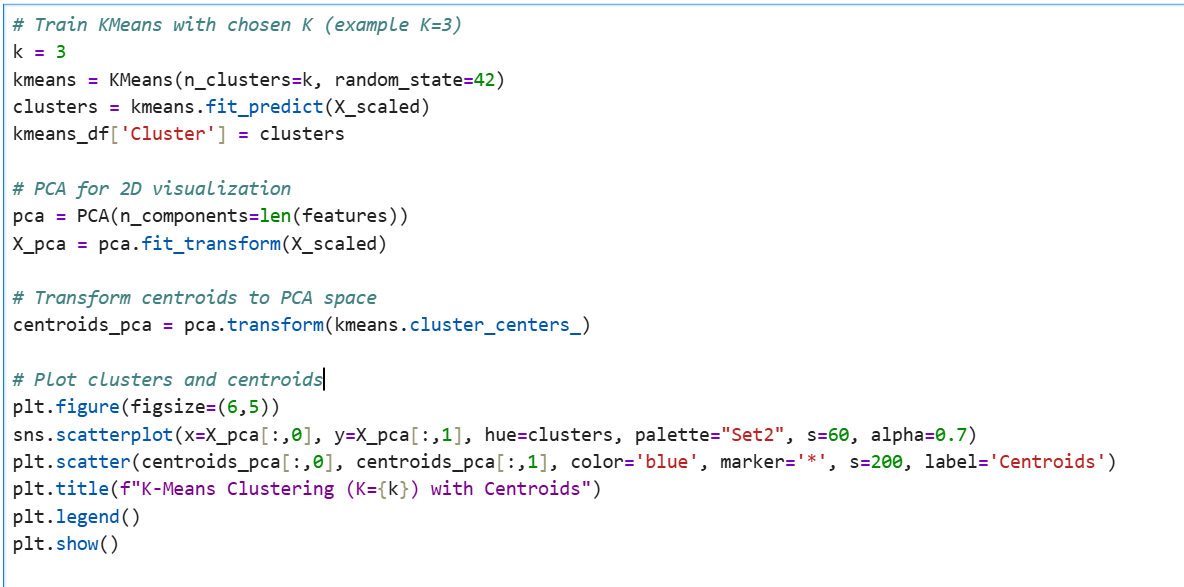
* 1. **Specific Requirements, Functions and Formulas**

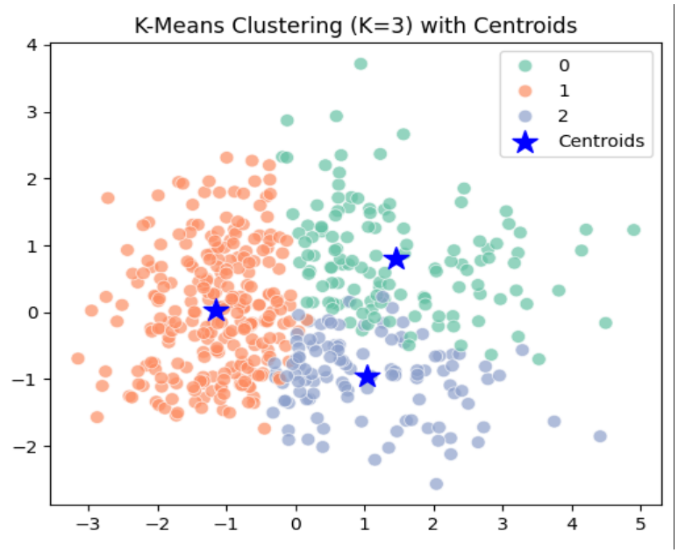
**Functions Used:**

* KMeans()
* StandardScaler()
* PCA()
* fit\_predict()

**Distance Formula (Euclidean Distance):**

**Code Snippet:**

****

* 1. **Analysis Results**
* Elbow Method shows a clear bend at **K = 3**
* Three meaningful clusters were formed based on health and lifestyle patterns
* Each cluster represents a distinct stress–productivity profile
* Feature scaling improved cluster separation
  1. **Visualization**
* Elbow curve confirms **3 optimal clusters**
* PCA-based scatter plot shows clear cluster separation
* Centroids indicate representative profiles for each cluster

**Objective 9: Hierarchical Clustering**

* 1. **Introduction**

Hierarchical clustering is an unsupervised learning technique used to group data points based on similarity. It helps visualize natural groupings in data through a tree-like structure called a dendrogram.

* 1. **General Description**

In this objective, hierarchical clustering is applied to analyze relationships among individuals based on BMI, stress level, sleep quality, balanced life, and productivity score. The dendrogram is used to determine the optimal number of clusters.

* 1. **Specific Requirements, Functions and Formulas**

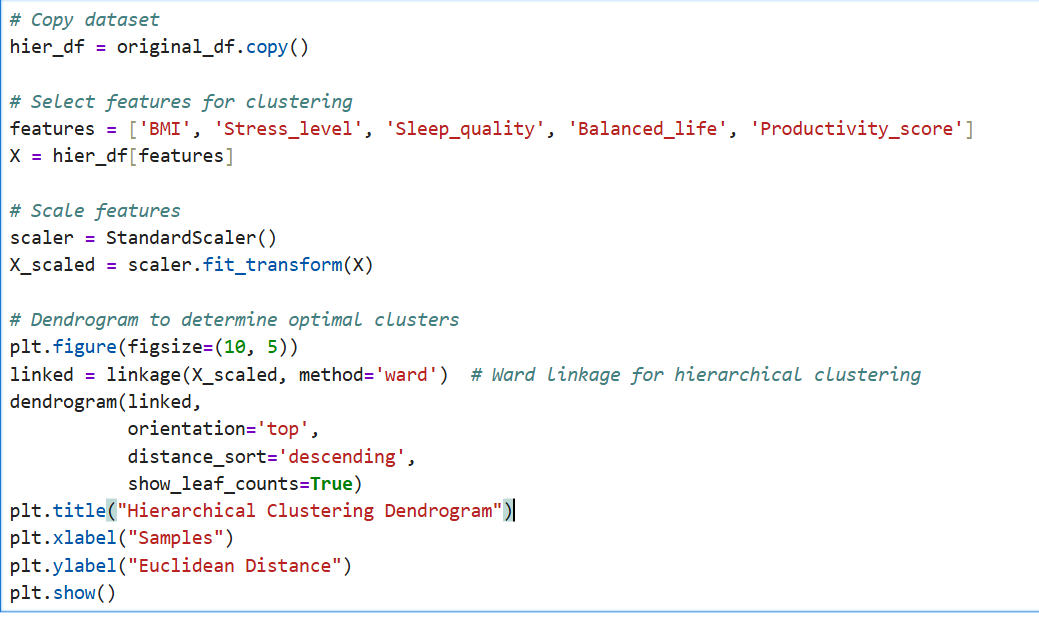
**Functions Used:**

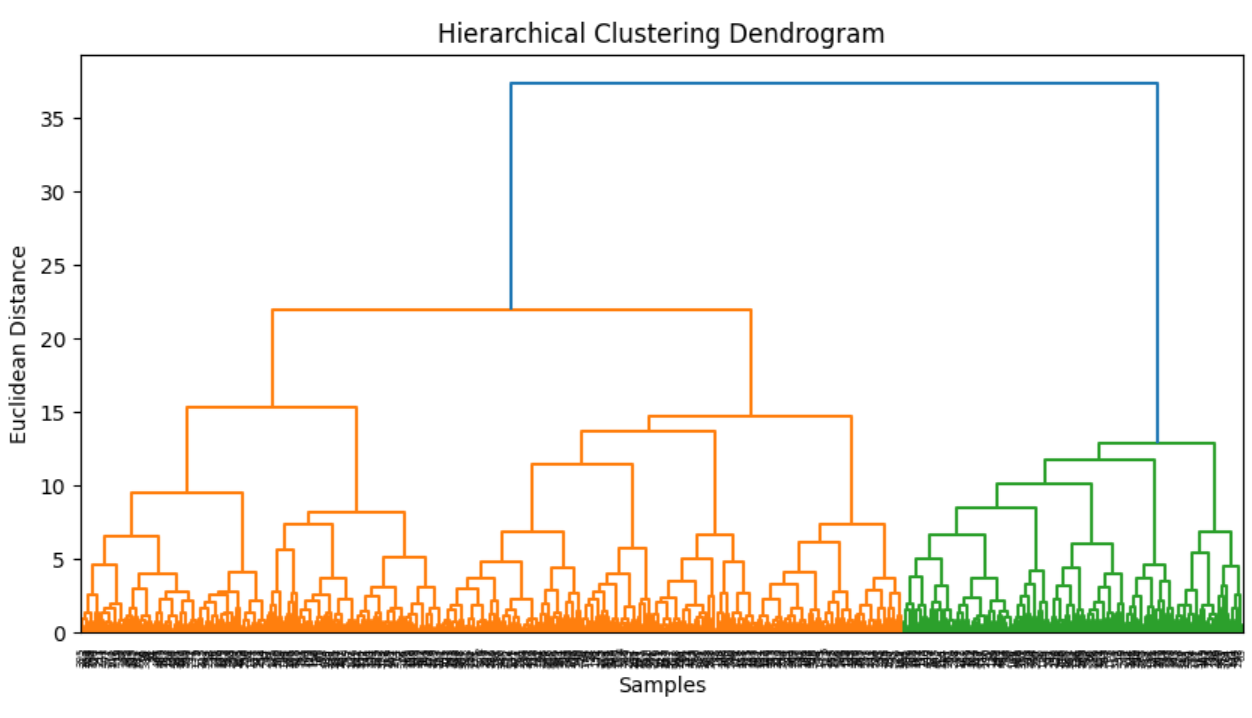
* StandardScaler()
* linkage()
* dendrogram()

**Linkage Method:**

* **Ward’s Method** (minimizes variance within clusters)

**Distance Formula (Euclidean Distance):**

**Code Snippet:**

* 1. **Analysis Results**
* The dendrogram shows clear hierarchical structure
* Clusters are formed based on similarity in health and lifestyle features
* Large vertical distances indicate well-separated clusters
* Optimal cluster count can be chosen by cutting the dendrogram at an appropriate height
  1. **Visualization**
* Dendrogram visually represents cluster formation
* Height of merges reflects dissimilarity between clusters
* Helps compare hierarchical grouping with K-Means results

**CONCLUSION**

The project aimed at analyzing a comprehensive health dataset to provide insights into **BMI prediction, stress level prediction, and clustering of health behaviors**. The study leveraged both supervised and unsupervised machine learning techniques, demonstrating the utility of predictive analytics for health management.

**Objectives 1–5: Data Understanding, Cleaning, and Exploratory Analysis**

1. **Data Understanding and Preprocessing**

* The dataset included features such as **Age, Gender, BMI, Sleep\_hours, Exercise\_time, Stress\_level, Happiness\_level, Balanced\_life, Productivity\_score, and lifestyle habits**.
* Missing values were handled appropriately, and categorical variables were encoded for model compatibility.
* Feature scaling and selection were applied to ensure models were trained on relevant and normalized data.

1. **Exploratory Data Analysis (EDA)**

* Correlation analysis revealed significant relationships between **BMI, Sleep\_quality, Exercise\_time, and Health\_rating**.
* Distribution plots highlighted patterns, such as the majority of participants having normal BMI, moderate stress levels, and varied sleep quality.
* Boxplots and heatmaps provided insights into feature variability, supporting subsequent model selection and feature engineering.

1. **Feature Selection and Dimensionality Reduction**

* **SelectKBest** was applied to identify the most relevant predictors for BMI and stress.
* Features like **Age, Exercise\_time, Junk\_food\_freq, Sleep\_hours, Happiness\_level, and Balanced\_life** were consistently important across multiple models.
* PCA was used for clustering visualization to reduce dimensionality while preserving variance.

1. **Data Visualization and Pattern Recognition**

* Visualizations such as scatterplots, histograms, and heatmaps were crucial for understanding relationships between lifestyle factors and health outcomes.
* Patterns such as higher stress correlating with poor sleep quality and low productivity were identified, supporting the choice of predictive models.

1. **Initial Statistical Analysis**

* Correlation matrices and descriptive statistics quantified the impact of lifestyle factors on BMI and stress.
* Early insights suggested non-linear relationships, indicating the need for both linear and non-linear modeling techniques in later objectives.

**Objectives 6–7: Predictive Modeling**

**BMI Prediction (Objectives 6.1–6.6)**

* Regression models including **Linear, Polynomial, KNN, Decision Tree, Random Forest, and SVR** were trained to predict BMI.
* **Random Forest Regressor** achieved the best performance due to its ability to model non-linear interactions and handle feature importance effectively.
* Feature importance analysis highlighted **Age, Exercise\_time, Alcohol, Health\_rating, and Balanced\_life** as significant predictors of BMI.
* SVR also performed well, especially for non-linear trends, but required careful feature scaling.

**Stress Level Prediction (Objectives 7.1–7.6)**

* Stress levels were categorized into **Low, Medium, and High**, and classifiers including **Logistic Regression, KNN, Decision Tree, Random Forest, SVM, and Naive Bayes** were implemented.
* **SVM (Linear Kernel)** and **Random Forest** provided the highest accuracy (~0.89) with strong precision and recall.
* Logistic Regression and KNN offered interpretable and simple solutions, whereas Naive Bayes provided good baseline performance.
* Feature analysis revealed **Sleep\_hours, Happiness\_level, Sleep\_quality, BMI, and Productivity\_score** as key determinants of stress levels.

**Objectives 8–9: Unsupervised Learning (Clustering)**

**K-Means Clustering**

* K-Means was applied to features including **BMI, Stress\_level, Sleep\_quality, Balanced\_life, and Productivity\_score**.
* Using the elbow method, **K=3 clusters** was identified as optimal.
* PCA was applied to visualize clusters in 2D space, showing distinct groups based on combined lifestyle and health metrics.
* Cluster analysis highlighted groups with **healthy, moderate, and high-risk profiles**, aiding targeted interventions.

**Hierarchical Clustering**

* Hierarchical clustering with Ward linkage confirmed similar grouping patterns, visualized via dendrograms.
* This approach provided insight into **nested relationships** between individuals, highlighting gradations in lifestyle and stress patterns that may not be captured by K-Means alone.

**Overall Insights**

1. **Ensemble and kernel-based models** consistently outperform simpler models for both BMI prediction and stress classification.
2. **Lifestyle and health factors**—especially **sleep, exercise, balanced life, happiness, and health ratings**—play a critical role in predicting both BMI and stress levels.
3. **Unsupervised learning** complements supervised models by identifying natural clusters, revealing health and behavior patterns that can inform personalized interventions.
4. Combining predictive modeling with clustering provides a **holistic view** of individuals, enabling proactive health management strategies.

**FUTURE SCOPE**

The present study demonstrates the effectiveness of machine learning in predicting **BMI and stress levels** and analyzing health patterns through clustering. The future scope of this work can be expanded in several ways:

1. **Integration with Wearable Devices**
   * Incorporate real-time data from fitness trackers and smartwatches (e.g., heart rate, step count, sleep patterns) to improve predictive accuracy.
   * Continuous monitoring can help in early detection of health anomalies and stress levels.
2. **Inclusion of Additional Health Metrics**
   * Include dietary information, calorie intake, hydration levels, and mental health assessments for a more holistic analysis.
   * Advanced physiological measurements (like blood pressure, glucose levels) can refine predictive models.
3. **Hybrid Machine Learning Models**
   * Develop ensemble models combining multiple algorithms (e.g., Random Forest + SVR + Neural Networks) to enhance prediction performance.
   * Explore deep learning models for non-linear, high-dimensional feature interactions.
4. **Personalized Health Recommendations**
   * Use clustering insights to design **individualized wellness plans**, targeting diet, exercise, sleep, and stress management.
   * Integration with mobile applications can deliver **real-time suggestions** for lifestyle improvements.
5. **Predictive Maintenance for Health Trends**
   * Implement time-series analysis to track trends in BMI, stress, and lifestyle factors.
   * Predict future health risks and alert users to prevent chronic conditions.
6. **Scalability and Deployment**
   * Deploy models as cloud-based services for wider accessibility.
   * Integration with hospitals, gyms, and wellness centers for practical application in health management systems.
7. **Cross-Domain Applications**
   * Extend models to other domains such as **workplace productivity, mental health assessments, and preventive healthcare planning**.

**REFERENCES**

 **Seaborn Library** – <https://seaborn.pydata.org>

 **Pandas Library** – <https://pandas.pydata.org>

 **Matplotlib Library** – <https://matplotlib.org>

 **Scikit-learn Library** – <https://scikit-learn.org>

 **NumPy Library** – <https://numpy.org>

**LINKS**

**LINKEDIN:**

<https://www.linkedin.com/feed/update/urn:li:activity:7405483902284468224/>

**GITHUB:**

<https://github.com/RamaSaiJahnavi/Health-Lifestyle-Analysis>

**GOOGLE DRIVE:**

<https://drive.google.com/drive/folders/1NA4a_gGuPC1_3wsxIUHPYftJE2SvEWOw?usp=sharing>

**GOOGLE FORM LINK:**

<https://docs.google.com/forms/d/167Vav-k7O1SMqprD4vhdhhKY6cdiz-d3ySmDxfK3WPs/edit#responses>