FitCheck:Geographically Informed Obesity and Calorie Tracking System

A Report Submitted in Partial Fulfilment of the Requirements for the

SN Bose Internship Program, 2024

Submitted by

Shrey Tolasaria (2115062)
Kaushik Borah (2115034)
Hritik Baranwal (2115052)

Under the guidance of

Dr. Vipin Chandra Pal
Assistant Professor
Department of Electronics & Instrumentation Engineering
National Institute of Technology Silchar

Department of Electronics & Instrumentation Engineering
NATIONAL INSTITUTE OF TECHNOLOGY SILCHAR
Assam

June-July, 2024

DECLARATION

"FitCheck: Geographically Informed Obesity and Calorie Tracking System"

We declare that the art on display is mostly comprised of our own ideas and work, expressed in our own words. Where other people's thoughts or words were used, we properly cited and noted them in the reference materials. We have followed all academic honesty and integrity principles.

Shrey Tolasaria (2115062)
Kaushik Borah (2115034)
Hritik Baranwal (2115052)

Department of Electronics and Instrumentation Engineering
National Institute of Technology Silchar, Assam

ACKNOWLEDGEMENT

We would like to express our deepest gratitude to our supervisor, Dr. Vipin Chandra Pal, EIE, NIT Silchar, for his invaluable direction, encouragement, and assistance throughout this project. His continuous guidance, insightful suggestions, and unwavering support were instrumental in the successful completion of this work. He provided us with new ideas, encouraged us to iterate and improve our project repeatedly, and was always available to assist us, offering us the freedom and flexibility to explore various approaches.

We also extend our sincere thanks to him for his patience, support, and invaluable contributions, which made it possible to complete this project properly and on schedule. His help with data sets, answering our queries, and providing innovative solutions whenever we were stuck was crucial to the project's success.

Shrey Tolasaria (2115062)
Kaushik Borah (2115034)
Hritik Baranwal (2115052)

Department of Electronics and Instrumentation Engineering
National Institute of Technology Silchar, Assam

ABSTRACT

FitCheck is a comprehensive tool designed to address the growing concern of obesity, particularly in India, by leveraging machine learning and advanced calorie tracking technologies. The core of FitCheck is a decision tree-based machine learning model that predicts an individual's weight level by analyzing multiple factors, including BMI, waist size, gender, age, and geographical location. The model places significant emphasis on abdominal obesity, a critical health issue in India, and provides insights into its prevalence and impact.

In addition to weight level prediction, FitCheck offers tailored calorie recommendations to assist users in achieving their weight management goals. These recommendations are categorized into four distinct levels: maintenance, mild weight loss (0.25 kg/week), moderate weight loss (0.5 kg/week), and extreme weight loss (1 kg/week). The calculation of these recommendations considers the user's basal metabolic rate (BMR) and is further refined by their physical activity level, which is classified into five categories: Little/No Exercise, Light Exercise, Moderate Exercise (3-5 days/week), Very Active (6-7 days/week), and Extra Active (Physically demanding job).

Enhancing the user experience, FitCheck integrates a calorie calculator powered by Google's Gemini Pro Vision model. This innovative feature allows users to upload food images for precise calorie estimation, thereby simplifying the process of tracking daily calorie intake.

Overall, FitCheck provides a unique blend of geographical and machine learning insights, offering a user-friendly and scientifically grounded approach to weight management.

List of Tables

Contents

Chapter 1

Introduction

Obesity has become a global health challenge, with its prevalence rapidly increasing across various populations, including India. The multifaceted nature of obesity, influenced by factors such as lifestyle, dietary habits, and genetics, makes it a complex condition to address. In the Indian context, geographical diversity, coupled with unique cultural and dietary patterns, significantly impacts the distribution and severity of obesity. This project, FitCheck, aims to tackle this issue by leveraging machine learning algorithms and advanced image recognition technologies to predict and manage weight levels, particularly focusing on the Indian population.

The primary objective of FitCheck is to analyze various factors contributing to an individual's weight level, such as Body Mass Index (BMI), waist size, gender, age, and geographical location. By integrating these factors, the model can predict a person's weight category and offer personalized recommendations for calorie intake and weight management. Notably, recent studies suggest that a composite of BMI and waist circumference may serve as a more accurate obesity metric for Indians at high risk for type 2 diabetes. This is particularly emphasized in the study by Nagarathna et al. (2020) [1], which highlights the importance of waist size as a critical factor in assessing obesity-related health risks, especially in the Indian population.

1.1 Geographical Influence on BMI and Obesity

Geography plays a pivotal role in determining BMI and obesity prevalence due to significant variations in lifestyle, dietary habits, and physical activity levels across different regions of India. For instance, urban areas may have higher obesity rates due to sedentary lifestyles and increased consumption of

processed foods, while rural areas might exhibit different patterns influenced by agricultural practices and traditional diets. By considering geographical location in the model, FitCheck provides more accurate and region-specific predictions and recommendations, making the tool highly relevant for the diverse Indian population.

1.2 The Concept of Basal Metabolic Rate (BMR) and Calorie Recommendations

Basal Metabolic Rate (BMR) is a critical factor in determining an individual's daily calorie needs. BMR represents the number of calories required to maintain basic physiological functions at rest, such as breathing, circulation, and cell production. FitCheck utilizes BMR calculations to provide personalized calorie recommendations based on the user's weight management goals. By adjusting the recommended calorie intake according to the user's physical activity level and desired weight loss target, the app ensures that the suggestions are both practical and effective.

1.3 The Importance of Waist Size in Obesity Management

Waist circumference is increasingly recognized as a significant indicator of health risks associated with obesity, particularly abdominal obesity. Unlike general obesity, which may be distributed throughout the body, abdominal obesity refers to the excessive accumulation of fat around the abdomen, which is more strongly linked to cardiovascular diseases, type 2 diabetes, and metabolic syndrome. Given its relevance, waist size is a crucial factor in the FitCheck model. The study by Nagarathna et al. (2020) [1] underscores that combining BMI with waist circumference provides a more comprehensive assessment of obesity in Indians, particularly those at risk of type 2 diabetes. By incorporating this composite metric, FitCheck enhances its ability to predict obesity-related health risks more accurately.

1.4 Objectives

The key objectives of this project include:

- **Analyzing the Indian Population:** To develop a model that accurately reflects the obesity trends and challenges specific to India,

considering the unique dietary habits, lifestyle, and regional variations across the country.

- **Investigating Machine Learning Techniques:** To explore and implement various machine learning techniques, particularly decision tree algorithms, for predicting weight levels based on a composite of factors such as BMI, waist size, gender, age, and geographical location.

This chapter sets the stage for the development and application of FitCheck, outlining the motivations behind the project and the specific goals it aims to achieve.

Chapter 2

Methodology

2.1 Data Preprocessing and Cleaning

The initial dataset we sourced contained a comprehensive range of columns. However, many of these columns were irrelevant for our specific analysis. For clarity, we eliminated unnecessary columns and focused on the essential ones.

The raw data included the following columns:

Table 2.1: Original Dataset Columns

PatientId
Zone
State District
Age
Gender
Waist
Height
Weight
Diabetes SelfDecl
Diafather
Diamother
Moderate
Vigorous
Dailyphysical
Hba1c
DiabetesCalc
Diabetes
ClinicalNotes
Bmi
WcRiskScore
BmiRiskScore
BmiWcRiskScore
IdrsWaist

For our analysis, we focused on the following cleaned dataset columns:

Figure 2.1: Screenshot of Cleaned Data

id	Zone	Age	Gender	Waist	Height	Weight	Moderatel	Vigorouspl	Dailyphysi	Bmi	BmiWcRiskScore
0	W	35	Female	70	157	47	0	0	3	19.06771	1
1	E	38	Female	81	145	65	0	3	1	30.91558	4
4	W	42	Female	102	159	68	0	0	2	26.89767	3
7	E	37	Female	76	142	44	2	3	2	21.82107	1
8	E	47	Male	87	160	60	2	3	3	23.4375	2
10	S	59	Female	83	158	57	0	0	4	22.83288	2
13	E	30	Female	72	151	84	0	3	1	36.84049	4
14	E	37	Female	81	151	60	0	3	3	26.31464	3
15	E	32	Female	72	140	65	0	3	1	33.16327	4
17	E	43	Male	81	140	65	2	3	3	33.16327	4
19	E	46	Male	82	162	65	1	1	2	24.76757	2
25	E	24	Male	92	158	58	2	1	1	23.23346	3
26	E	25	Female	87	146	69	0	3	1	32.37005	4
29	E	27	Female	76	148	63	2	3	3	28.76187	3
30	E	35	Female	87	149	45	2	3	3	20.26936	2
35	W	46	Female	82	149	57	0	0	2	25.67452	3
37	NW	52	Female	86	148	52	1	0	4	23.73996	3

In the cleaned dataset, we have removed the following columns:

- State District
- Diabetes SelfDecl
- Diafather
- Diamother
- Hba1c
- DiabetesCalc
- Diabetes
- ClinicalNotes
- BmiRiskScore
- IdrsWaist

These columns were excluded as they were either redundant or irrelevant to our analysis objectives.

The cleaned data is now ready to be saved into a CSV format, which will be used for training our machine learning model. By following this process, we ensure that the data is well-structured and focused on the parameters crucial for our analysis.

2.2 Exploratory Data Analysis (EDA) and Visualization

In this section, we perform Exploratory Data Analysis (EDA) to understand the dataset better and visualize various aspects of the data using Matplotlib and Seaborn.

2.2.1 EDA

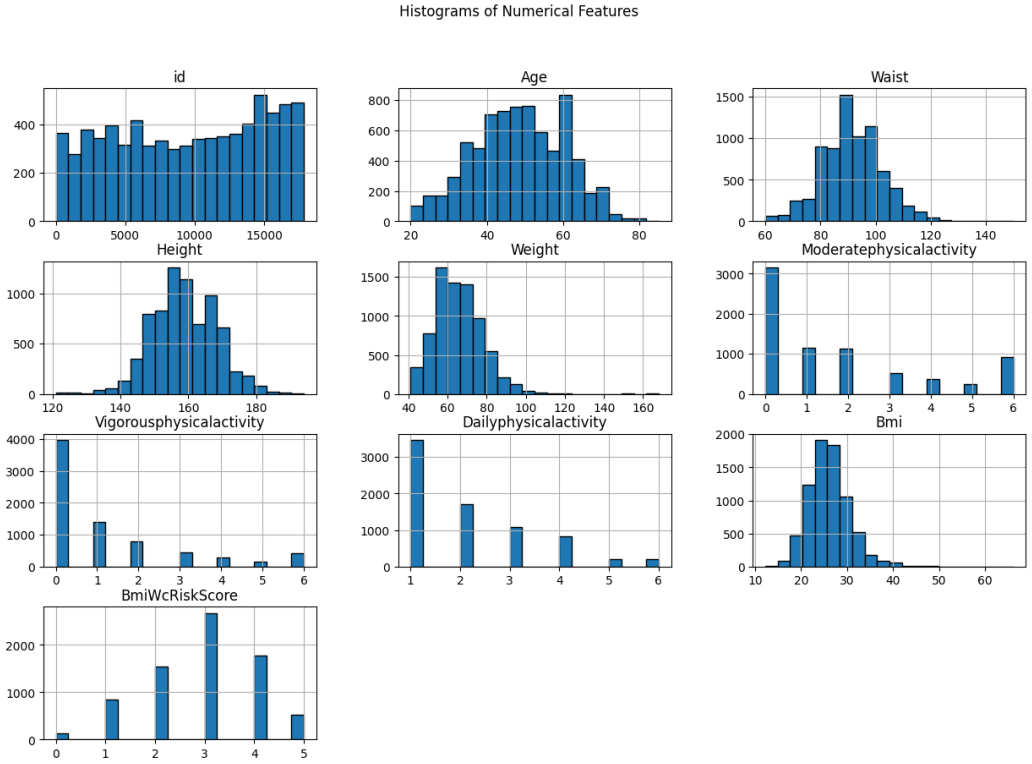
The initial examination of the dataset revealed the following:

- **First few rows of the dataset:** The first few rows of the dataset provide a glimpse of the structure and values of the dataset. The data includes columns such as Age, Waist, Height, Weight, and physical activity levels.
- **Basic Information:** The dataset consists of 7496 entries with 12 columns, including numerical and categorical features.
- **Summary Statistics:** Summary statistics show the distribution of values for numerical features, including measures such as mean, standard deviation, minimum, and maximum.
- **Missing Values:** There are no missing values in the dataset, indicating that the data is complete and ready for analysis.

2.2.2 Visualization

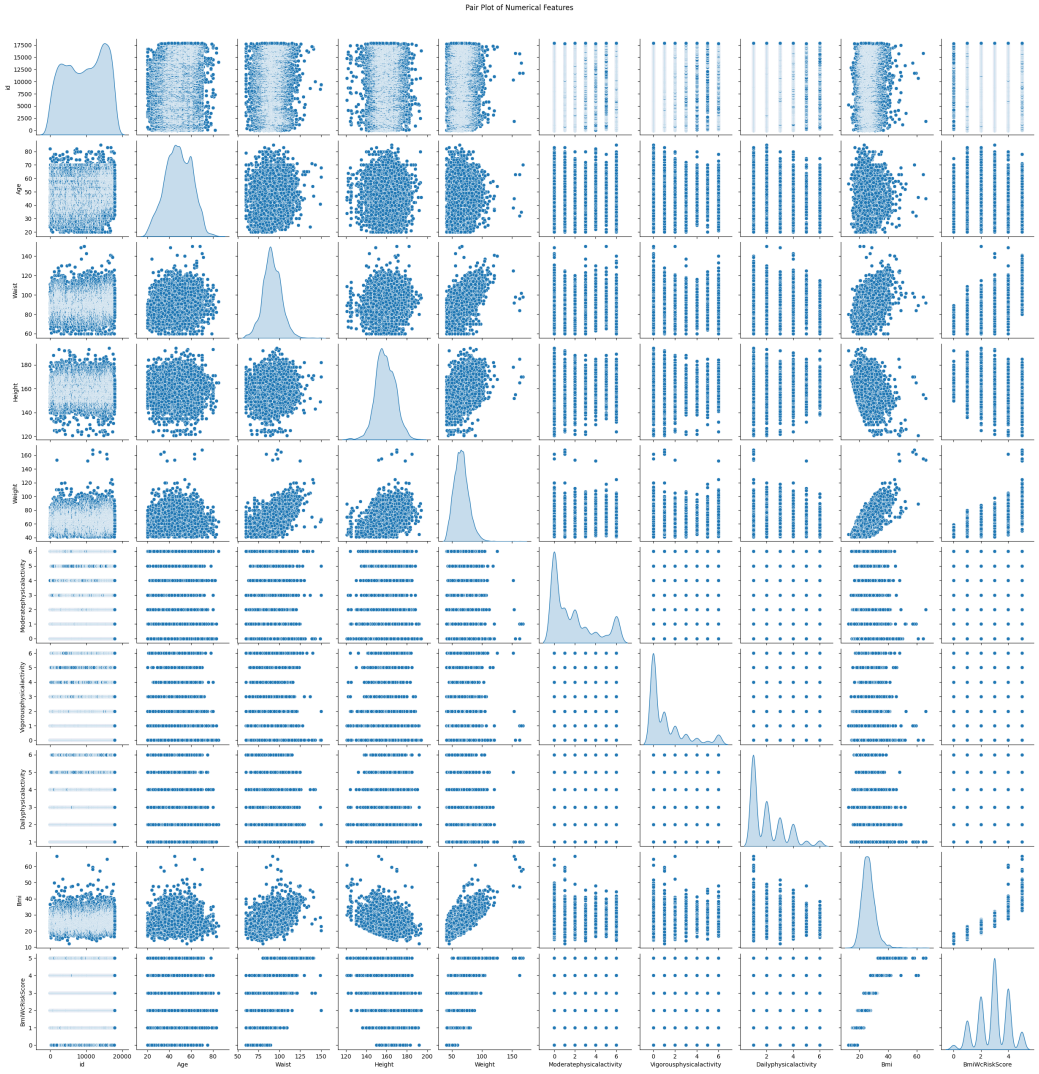
We performed several visualizations to gain insights into the dataset:

Figure 2.2: Histograms of All Numerical Features



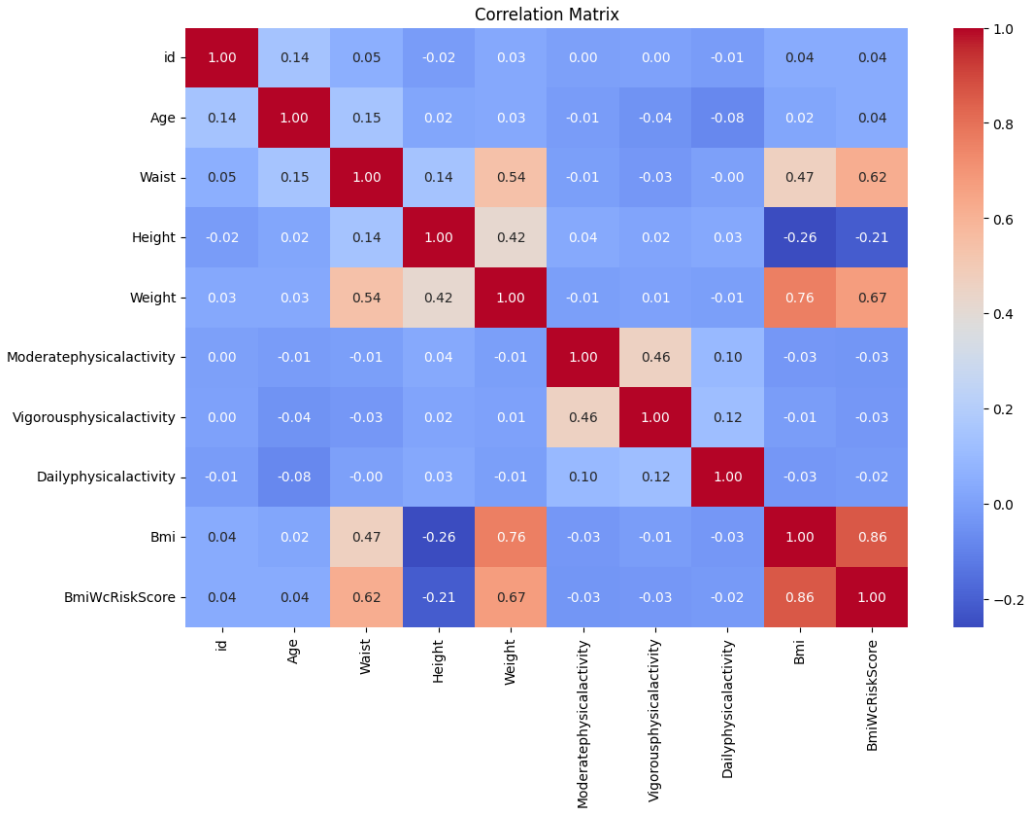
Histograms of all numerical features provide a view of the distribution of each feature. This helps in understanding the frequency distribution and potential skewness in the data.

Figure 2.3: Pair Plot of All Numerical Features



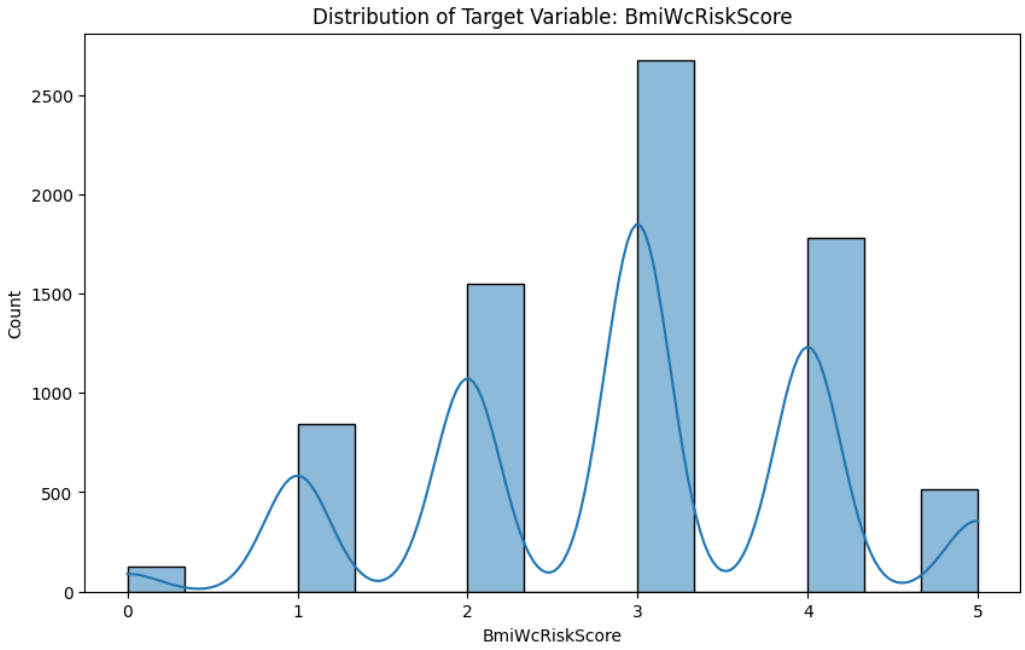
Pair Plot of all numerical features shows the relationships between pairs of numerical variables, helping identify patterns and correlations among them.

Figure 2.4: Correlation Matrix



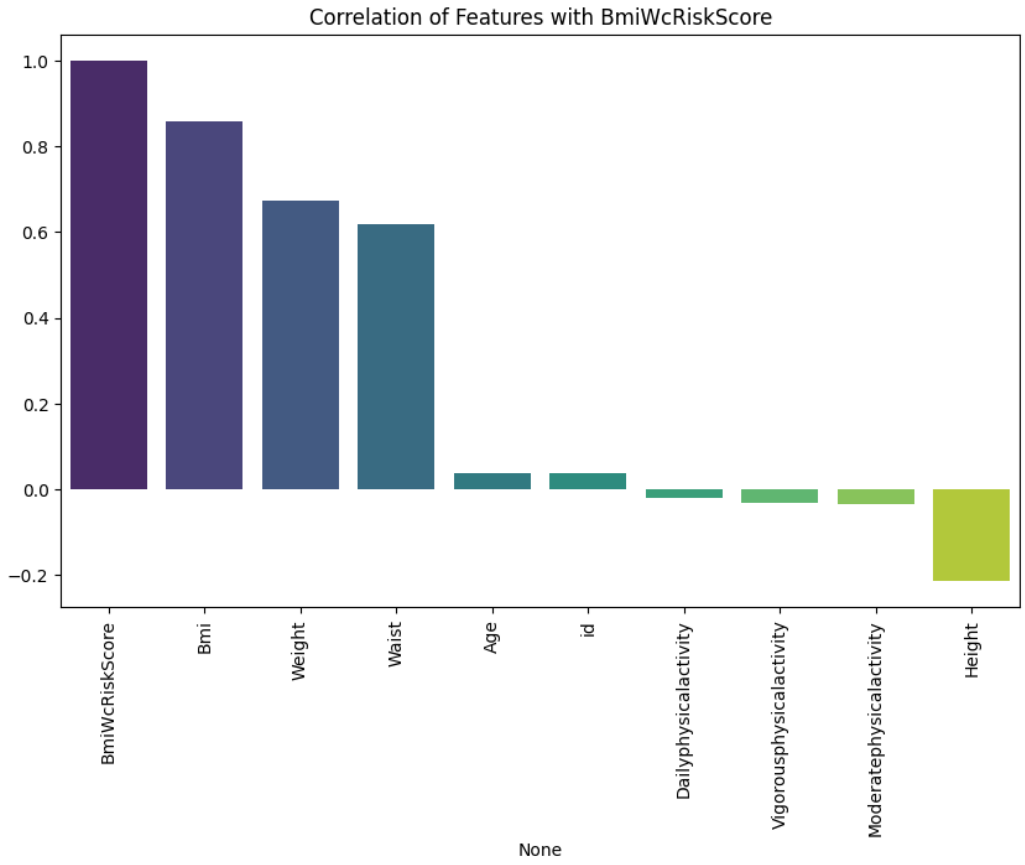
Correlation Matrix visualizes the correlations between numerical features, highlighting how strongly features are related to each other.

Figure 2.5: Distribution of Target Variable: BmiWcRiskScore



Distribution of Target Variable: BmiWcRiskScore shows the frequency distribution of the target variable, giving insight into its overall distribution and central tendencies.

Figure 2.6: Correlation of Features with BmiWcRiskScore



Correlation of Features with BmiWcRiskScore displays how each feature correlates with the target variable, which is crucial for understanding which features have the most influence on the target variable.

The visualizations and statistical analysis provide a comprehensive understanding of the dataset and its characteristics, aiding in informed decision-making for further modeling and analysis.

2.3 Machine Learning Models

In this chapter, we discuss several machine learning models used for classification tasks: Decision Tree Classifier, Random Forest Classifier, Naive Bayes Classifier, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN). For each model, we provide an overview of its purpose, working mechanism, and evaluation metrics.

2.3.1 Naive Bayes Classifier

Purpose

The Naive Bayes Classifier is used for classification tasks and is particularly effective for text classification and problems with categorical data. It operates under the assumption that features are independent given the class label.

Working

Naive Bayes applies Bayes' theorem with the "naive" assumption of feature independence. It calculates the posterior probability of each class based on input features and selects the class with the highest probability.

Evaluation

Performance is evaluated using accuracy, precision, recall, and F1-score. The model's results are compared with those of other classifiers to assess its effectiveness.

2.3.2 Decision Tree Classifier

Purpose

The Decision Tree Classifier creates a model that predicts the value of a target variable based on several input variables. It is known for its simplicity and interpretability.

Working

Decision Trees recursively split the data into subsets based on features that provide the best separation. The tree is built by choosing the feature that maximizes information gain or minimizes impurity.

Evaluation

Evaluation metrics include accuracy, confusion matrix, and classification report. Cross-validation is used to ensure the model's robustness.

2.3.3 Random Forest Classifier

Purpose

Random Forest is an ensemble method that combines multiple decision trees to improve classification accuracy and reduce overfitting.

Working

It builds multiple decision trees using subsets of data and features. The final prediction is based on the majority vote from all the trees.

Evaluation

Performance is evaluated using accuracy, precision, recall, F1-score, and feature importance. Cross-validation is used to assess the model's generalizability.

2.3.4 Support Vector Machine (SVM)

Purpose

SVM is used for classification and regression tasks, especially in high-dimensional spaces and with non-linearly separable classes.

Working

SVM finds the hyperplane that best separates classes in feature space. It uses kernel functions to transform data into a higher-dimensional space where a linear separation is possible.

Evaluation

Performance is evaluated using accuracy, precision, recall, F1-score, and different kernel functions and regularization parameters.

2.3.5 K-Nearest Neighbors (KNN)

Purpose

KNN is a non-parametric method used for classification and regression tasks. It makes predictions based on the closest training examples in the feature space.

Working

KNN assigns the class or value based on the majority vote or average of the K-nearest neighbors. Distance metrics such as Euclidean or Manhattan are used to find the nearest neighbors.

Evaluation

Performance is evaluated using accuracy, precision, recall, F1-score, and choice of K value.

Each of these models is analyzed based on its strengths and weaknesses, and their performance on the dataset is compared to determine the best approach for our analysis.

2.4 Model Evaluation

We have trained each machine learning model on the dataset. The evaluation of these models is crucial to assess their performance and select the most effective model for our task. The evaluation involves using various metrics to gauge how well each model performs.

2.4.1 Evaluation Metrics

- **Accuracy:** Measures the proportion of correct predictions among the total predictions.
- **Precision:** Indicates the proportion of true positive predictions among all positive predictions.
- **Recall:** Represents the proportion of true positive predictions among all actual positives.
- **F1-Score:** The harmonic mean of precision and recall, providing a single metric that balances both.

The models are evaluated based on these metrics to determine which one provides the best performance for our specific dataset.

2.5 Model Persistence

To facilitate the reuse of trained models without the need for retraining, we use serialization tools such as ‘joblib‘ or ‘pickle‘. This ensures that the best-performing models can be stored and reloaded efficiently.

```
import joblib

# Save the best model
joblib.dump(best_model, 'best_model.pkl')

# Load the model
loaded_model = joblib.load('best_model.pkl')
```

Model persistence enables us to manage and deploy models effectively.

2.6 Deployment

For deploying our machine learning models and providing an intuitive user interface, we use the Streamlit Python library. Streamlit allows us to create interactive web applications that make it easy to showcase and use our models.

```
import streamlit as st

# Basic setup for Streamlit application
st.title('Machine Learning Model Deployment')
st.write('Welcome to the model deployment application.')

# Load the trained model
model = joblib.load('best_model.pkl')

# User input for prediction
input_data = st.text_input('Enter input data:')

if st.button('Predict'):
    prediction = model.predict([input_data])
    st.write('Prediction:', prediction)
```

Streamlit provides a straightforward way to deploy models and create interactive applications that users can engage with.

Chapter 3

Tools Used

3.1 Python

Python is the primary programming language used for developing our application. It is highly regarded for its versatility and extensive libraries, making it an excellent choice for data science and machine learning projects.

3.2 NumPy, Pandas, Matplotlib

- **NumPy:** This library is fundamental for numerical computations in Python. It provides support for arrays and matrices, along with a collection of mathematical functions to operate on these arrays.
- **Pandas:** Pandas is essential for data manipulation and analysis. It offers data structures like DataFrames, which are useful for handling and analyzing structured data efficiently.
- **Matplotlib:** Used for data visualization, Matplotlib enables the creation of static, animated, and interactive plots and graphs, helping in the graphical representation of data.

3.3 Scikit-Learn

Scikit-Learn is a powerful and widely-used machine learning library in Python. It provides simple and efficient tools for data mining and data analysis, including implementations of various machine learning algorithms such as Decision Trees, Random Forests, Naive Bayes, Support Vector Machines (SVM), and K-Nearest Neighbors (KNN).

3.4 Google Colab

Google Colab is a cloud-based platform that allows for Python code execution in a Jupyter notebook environment. It was used for training our models due to its powerful cloud computing resources, which were essential for handling the large dataset of over 7000 entries.

3.5 Joblib/Pickle

Joblib and Pickle are libraries used for model persistence. They enable us to save trained machine learning models and load them later without the need to retrain, thus facilitating efficient use and deployment of the models.

3.6 Google Gemini API

The Google Gemini API was utilized to obtain nutritional information from food images. It provides capabilities for estimating calorie content and analyzing the nutritional value of various meals.

3.7 Streamlit

Streamlit is a Python library used for developing interactive web applications quickly. It was employed to create the user interface for our application, allowing seamless integration of Python models and providing an easy-to-use interface for end-users.

3.8 Git/GitHub

Git and GitHub were used for version control and code management. Git helps track changes in the codebase, while GitHub provides a platform for collaboration, code sharing, and version control across the development team.

Chapter 4

Results

In this chapter, we present the accuracy results obtained from various machine learning models used in our project. The performance of each model is measured in terms of accuracy, which is the proportion of correctly predicted instances out of the total instances.

4.1 Model Accuracy

- **Decision Tree Accuracy:** 1.00
- **Random Forest Accuracy:** 0.99
- **Naive Bayes Accuracy:** 0.72
- **SVM Accuracy:** 0.35
- **K-Nearest Neighbors (KNN) Accuracy:** 0.72

4.2 Accuracy Comparison

The following diagram visually represents the accuracy of each model:

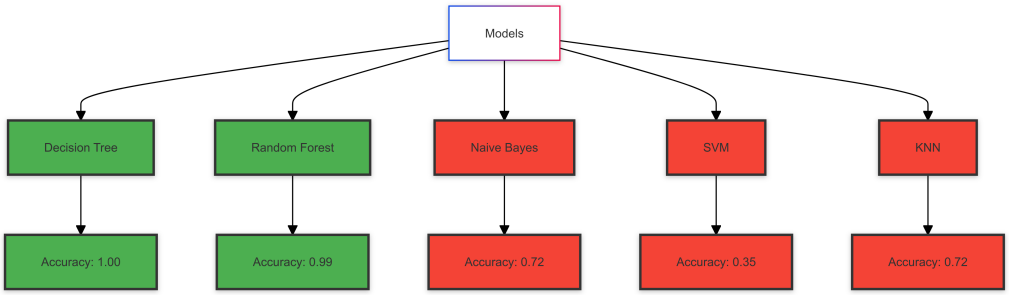


Figure 4.1: Accuracy Comparison of Machine Learning Models

Chapter 5

Future Work

As we look forward to enhancing and expanding the capabilities of our application, several key areas for future development have been identified. These improvements aim to make the application more robust, user-friendly, and tailored to the needs of its users.

5.1 Development of a Custom Computer Vision Model

Currently, our application relies on the Google Gemini API to analyze food images and estimate calorie content. While this API provides accurate results, it is primarily trained on a global dataset, which might not be fully optimized for the diversity of Indian cuisine. In the future, we plan to develop and integrate our own computer vision model specifically trained on an extensive Indian food dataset. This model would allow for more accurate calorie analysis and nutritional assessment tailored to Indian dietary habits. This enhancement will significantly improve the relevance and accuracy of dietary information provided to our users.

5.2 User Profile Integration

At present, the application does not support user profiles, which limits personalized interactions and insights. One of the major enhancements planned for the future is the integration of user profiles. By implementing this feature, users will be able to create and manage their accounts, allowing the application to store their past data such as calorie intake, weight trends, and activity levels. This stored data will enable the application to provide personalized

health insights, track progress over time, and offer tailored recommendations based on historical trends. This will greatly enhance user engagement and the overall utility of the application.

5.3 Personalized Diet Recommendation System

Another critical future enhancement is the development of a personalized diet recommendation system. This system will analyze a user's fitness goals (e.g., weight loss, muscle gain, maintenance) and food habits to generate customized diet plans. The diet recommendations will consider factors such as nutritional needs, calorie requirements, and individual preferences or restrictions. This feature will provide users with actionable guidance on their dietary choices, helping them achieve their fitness goals more effectively. By offering personalized meal plans, the application will become an indispensable tool for users striving to improve their health and well-being.

5.4 Conclusion

The planned future developments for our application are focused on providing more personalized, accurate, and culturally relevant health and diet insights. By developing a custom computer vision model, integrating user profiles, and introducing a personalized diet recommendation system, we aim to significantly enhance the user experience and effectiveness of our application. These improvements will ensure that our platform remains a valuable resource for individuals looking to manage their health and nutrition effectively.

Chapter 6

Conclusion

This project has been an enriching learning experience, especially for us as beginners in the field of machine learning. Through the development of our application, we have gained invaluable insights into the practical application of machine learning concepts. This project not only expanded our technical skills but also taught us the importance of teamwork, collaboration, and brainstorming innovative solutions to real-world problems.

Our app, FitCheck, is a solution-oriented platform designed to help users manage their weight and nutrition effectively. The app enables users to predict their weight based on various inputs, understand the caloric intake required to achieve specific weight loss goals, and track their food calories using image analysis powered by the Google Gemini model—all in one place. This seamless integration of features makes FitCheck a comprehensive tool for health management.

Furthermore, this project has provided us with the opportunity to acquire new and trending skills in machine learning and generative AI. These skills are increasingly relevant in today's tech landscape and will undoubtedly be beneficial in our future endeavors.

The final version of our app can be accessed through the following link: FitCheck App →.

The source code for this project is available on GitHub: FitCheck GitHub Repository →.

6.1 Final Remarks

In conclusion, the development of FitCheck has been a challenging yet rewarding journey. It has provided us with practical experience in applying machine learning to solve a real-world problem, while also helping us de-

velop essential skills in team collaboration and innovative thinking. We are confident that FitCheck will serve as a valuable tool for individuals seeking to manage their weight and improve their health through informed decisions and personalized recommendations.

References

[1] R. Nagarathna, S. S. Patil, H. Nagendra, R. SK, M. Venkatrao, and A. Singh. Data for: A composite of bmi and waist circumference may be a better obesity metric in indians with high risk for type 2 diabetes: an analysis of nmb-2017, a nationwide cross sectional study. Mendeley Data, V1, 2020.