Fitness Tracker WebApp Report

1.1 Overview

The Fitness Tracker WebApp uses machine learning to predict the number of calories burned based on various user parameters such as age, gender, BMI, exercise duration, heart rate, and body temperature. By entering personal details through a user-friendly interface, users can receive predictions about their calorie expenditure, explore similar results, and gain insights into how their parameters compare to others in the dataset.

1.2 Key Features

1.2.1 **User Input Parameters:**

•The sidebar allows users to input parameters such as:

•**Age:** Age range from 10 to 100 years
•**BMI:** Body Mass Index (15 to 40)
•**Duration:** Exercise duration in minutes (0 to 35)
•**Heart Rate:** Exercise heart rate in beats per minute (60 to 130)
•**Body Temperature:** Body temperature during exercise (36 to 42 degree C).
•**Gender:** Select between Male and Female

1.2.2 **Prediction of Calories Burned:**

The model predicts the number of calories burned based on the user's input using a Random Forest Regressor.
The prediction is based on a dataset that includes features like age, BMI, heart rate, and more, trained to predict calories burned.

1.2.3 **Comparison with Similar Results:**

The app identifies and displays five random entries from the dataset that are similar to the user's prediction, based on the predicted calories burned (within a ±10 kcal range).

1.2.4 **General Information:**

The app provides insights into how the user's parameters compare to the broader dataset:

Age: The percentage of people older than the user

Exercise Duration: How the user's exercise duration compares to others

Heart Rate: How the user's heart rate compares to others during exercise

Body Temperature: How the user's body temperature

1.3 Data Preprocessing and Model Training

1.3.1 **Data Import and Merging:**

The data is loaded from two CSV files: calories.csv (calories burned) and exercise.csv (user exercise data).

The two datasets are merged based on the common User_ID column, and the User_ID column is dropped for analysis.

1.3.2 **BMI Calculation:**

•A new BMI feature is added to the dataset by using the formula:

BMI=Weight(Height/100)2BMI = \frac{{\text{{Weight}}}}
{{(\
text{{Height}} / 100)^2}}

•This BMI value is used as a key feature in training the model.

1.3.3 **Train-Test Split:**

The dataset is split into training and testing sets using an 80-20 split.

1.3.4 **One-Hot Encoding:**

Categorical variables (e.g., gender) are one-hot encoded, which allows the model to handle these as numerical values.

1.3.5 **Model Selection and Training:**

A **Random Forest Regressor** model is used to predict the number of calories burned. This model is selected for its ability to handle complex datasets with multiple features and capture non-linear relationships.

The model is trained on the training data, and the prediction is made based on the user's input.

1.4 Key Results

1.4.1 **Prediction:**

The app provides a prediction of the number of kilocalories burned based on the user's input.

For example, if the user inputs parameters indicating moderate exercise, the app may predict a value like 150 kcal.

1.4.2 **Similar Results:**

The app shows users five random records from the dataset that are similar to their predicted calorie expenditure, helping users compare their predicted results with real-world data.

1.4.3 **Comparison Insights:**

Age Comparison: The app shows what percentage of people are older than the user.

Exercise Duration Comparison: It shows how the user's exercise duration compares to others.

Heart Rate and Body Temperature Comparison: Users can see how their heart rate and body temperature during exercise compare to others.

1.5 Model Performance

Model Type: Random Forest Regressor
Features Used: Age, BMI, Duration, Heart Rate, Body Temperature, and Gender.
Training Accuracy: While the app does not display explicit model accuracy, the Random Forest model was trained with an 80-20 split, and it generally performs well for regression tasks.
Prediction: The output is a prediction of calories burned, which is displayed with a confidence level based on the input parameters.

1.6 Technical Details

1.6.1 **Libraries Used:**

•**Streamlit:** For building the web app interface.

•**Pandas:** For data manipulation and processing.

•**Scikit-learn:** For model training and evaluation (Random Forest Regressor, train-test split, etc.).

•**Matplotlib and Seaborn:** For potential future visualizations

1.6.2 **Model Selection:**

•Random Forest was selected due to its ability to handle complex and non-linear data, making it ideal for predicting calories burned based on various physiological factors.

1.7 Future Enhancements

1.7.1 **Model Improvement:**

•**Hyperparameter Tuning:** To improve the model's performance, hyperparameters such as the number of trees (n_estimators), depth of trees (max_depth), and features to consider (max_features) can be fine-tuned.

1.7.2 **Additional Visualizations:**

•Adding visualizations, such as bar charts or histograms, to compare user input against the dataset more intuitively.

1.7.3 **Extended User Parameters:**

•Additional features like workout type, workout intensity, or more granular user data could further refine predictions.

1.7.4 **Real-time Data Integration:**

•Integrating real-time data from fitness trackers (e.g., Fitbit, Apple Watch) could make the app more personalized.

---

2 Conclusion

The Fitness Tracker WebApp provides a personalized experience for predicting calories burned during exercise based on user input. It leverages machine learning, specifically Random Forest, to generate accurate predictions and insights into how users compare with a broader dataset. This makes it a useful tool for individuals looking to track their fitness progress and understand their body’s performance during physical activity.