All data is my personal health data, I downloaded this data from the health app. My data is as follows:

A screenshot of a phone

Description automatically generated

Active Energy

A screenshot of a phone

Description automatically generated

Heart Rate Variability

A screenshot of a phone

Description automatically generated

Heart rate

A screenshot of a phone

Description automatically generated

Daily Steps

A screenshot of a phone

Description automatically generated

Standing Time

A screenshot of a phone

Description automatically generated

Rest Energy

A screenshot of a phone

Description automatically generatedA screenshot of a phone

Description automatically generated

Distance Walk and Run

Walk Speed

Gait Asymmetry

A screenshot of a phone

Description automatically generatedA screenshot of a phone

Description automatically generated

Floor Ascended

A screenshot of a phone

Description automatically generatedA screenshot of a phone

Description automatically generated

Walking Step Length

Double Support Time

Then, in order to use this data with Python, I compiled the data into Excel and created a table for each variable, allowing me to read this data from Python.

A screenshot of a computer

Description automatically generated

First of all, I imported the data into colab and started reading the variable with the pandas library. First, I started analyzing the data and checked the first 5 lines by making a .head from the data. Then, by .describing the data, I obtained count, mean, unique and similar properties for both numeric and non-numeric data.

A screenshot of a computer

Description automatically generated

Later, since I wanted to see HealthData.xlsx in a tabular form on the code, I tabulated the data using tabulate.

A close-up of a paper

Description automatically generated

Next, we used Python's matplotlib module to generate a bar chart depiction of a time-series dataset. To prepare the data for time-series analysis, a particular column must first be converted to a datetime format. The data is then arranged chronologically by setting this column as DataFrame index. A bar chart is produced once DataFrame is configured in this way. The plot is made with bars that reflect values of a specific column in the dataset, corresponding to daily measurements, and the scale of the figure is specified.

A graph of a step

Description automatically generated with medium confidence

Then, a line chart was prepared for the floor ascended variable, changing according to each day. This made it easier for us to analyze the data in the line chart.

A graph with a line

Description automatically generated

Then I prepared a box plot for Walking Step Length. This box plot shows the maximum and minimum values and is prepared with the average of each day's step length range.

A diagram with blue lines

Description automatically generated

Then I created heatmap thats color intensity denotes degree of connection between several variables in the dataset that in this instance seem to be associated with physical activity, including "Daily Steps," "Distance Walk and Run," and "Walking Step Length Mid."

The correlation between two variables represented by the row label and the column label is represented by each square on the heatmap. A complete positive correlation, or one in which both variables grow proportionately as one increases, is indicated by correlation value of 1.00. As would be predicted, the diagonal line of squares, where the labels for the row and column correspond to the same variable, displays a perfect correlation of 1.00. The dark color of the other off diagonal squares indicates negative correlations which point to inverse link between those variable pairs. For instance, a drop in "Walking Step Length Mid" may be associated with increase in "Daily Steps," and vice versa. A perfect negative correlation is shown by a scale on the right side of heatmap that goes from -1.0 to 1.0, with 0 denoting no correlation and 1.0 denoting a perfect positive correlation. Quickly grasping the pairwise correlations between several variables is made easier with the aid of this visualization, which is useful for a variety of tasks in data analysis, including recognizing underlying patterns in the data and feature selection in machine learning.

A graph of a heatmap

Description automatically generated

In order to predict target variable from the characteristics, a linear regression model is created and fitted to the training set. The model is used to generate predictions on the test set after training. The Mean Squared Error (MSE) metric, that calculates average of the squares of the errors between the anticipated and actual values, is used to quantify the accuracy of these predictions. A numerical representation of the model's predictive ability is given by the resulting MSE, where lower values indicate more accurate predictions.

A screenshot of a computer

Description automatically generated

I then used RandomForestRegressor to reduce the accuracy rate and MSE value, thus obtaining a lower MSE value.

A screenshot of a computer program

Description automatically generated