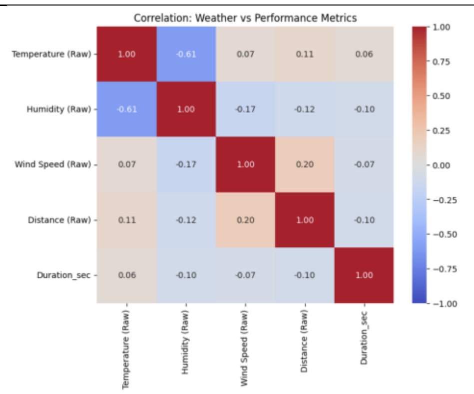
Data Analysis' Team (Week 1-6)

Project 3: ReflexionPro

Mentor: Ben Dang

Co-leadership: Billy Thai & Shin Nguyen

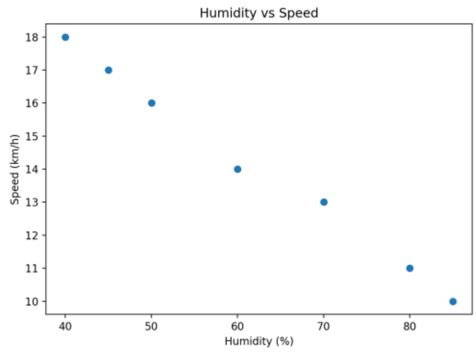
Name & Student	Task	Contribution & Evidence (Code & Output)
1. Billy Thai (223339662)	- Expanded dataset & retrain model - Develop model (Fatigue prediction, injury risk assessment, recovery time) - Visualise dashboard using Power BI (ongoing)	Expanded dataset - Transform the Garmin activity log into an analysis-ready dataset that captures workload and physiological context. Link: Fatigue Prediction Model - Designed and implemented an end-to-end machine-learning pipeline that predicts next-day fatigue status for athletes by analysing their wearable-derived training data. - Generated permutation importance plots, which highlighted acute load and chronic load as the two most influential predictors.
2. Shin Nguyen	- I have successfully prepared, cleaned, and processed raw fitness tracker datasets (Running & Cycling) into an analysis-ready format, implementing robust preprocessing steps such as datetime conversion, duration transformation, numeric coercion, and missing value handling. - I designed and implemented machine learning models (Random Forest Regressor) to predict running	 - The correlation heatmap was generated to explore relationships among the input variables and the target variable. The heatmap highlights both **positive and negative correlations**, with darker shades representing stronger relationships. This visualization helps to identify which features are most influential for predicting the outcome and to detect possible multicollinearity among predictors.

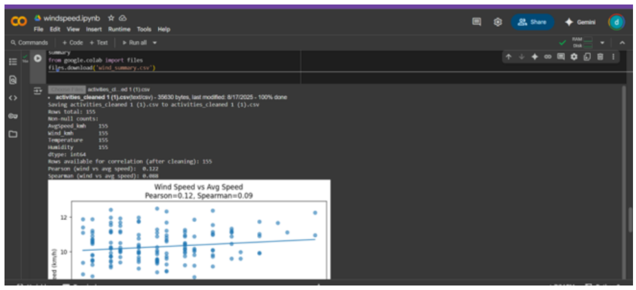
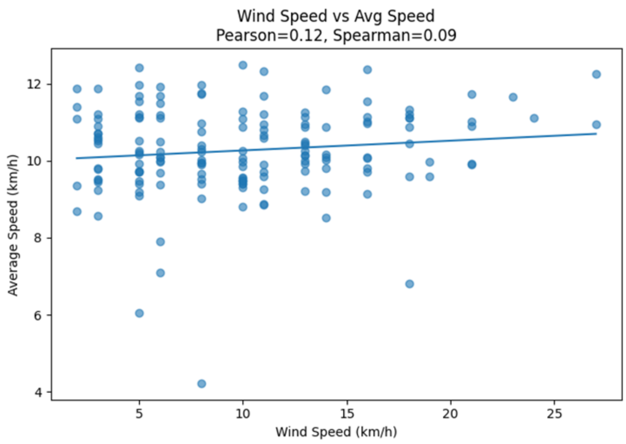
<table>
<tr>
<td></td>
<td>duration based on distance, elevation, pace, and environmental data, achieving an average R^2 score of 0.7863, and created correlation heatmaps as well to analyse performance and environmental impact</td>
<td><code># Train and Evaluate Running Duration Model
# Initialize model
rf_model = RandomForestRegressor(random_state=42)

Evaluate with our reusable CV function
evaluate_model(rf_model, X_run, y_run, scoring='r2')</code>

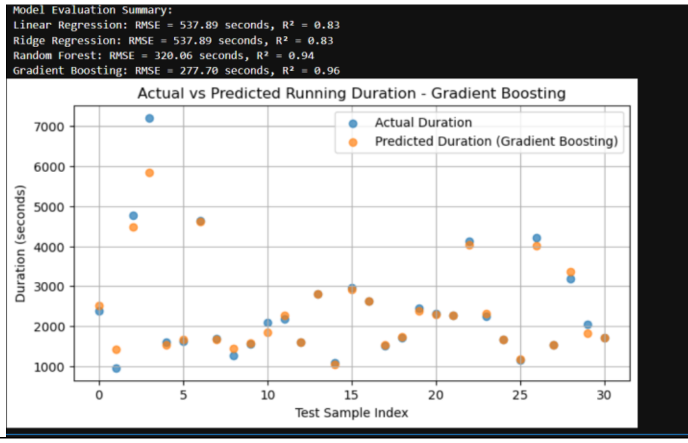
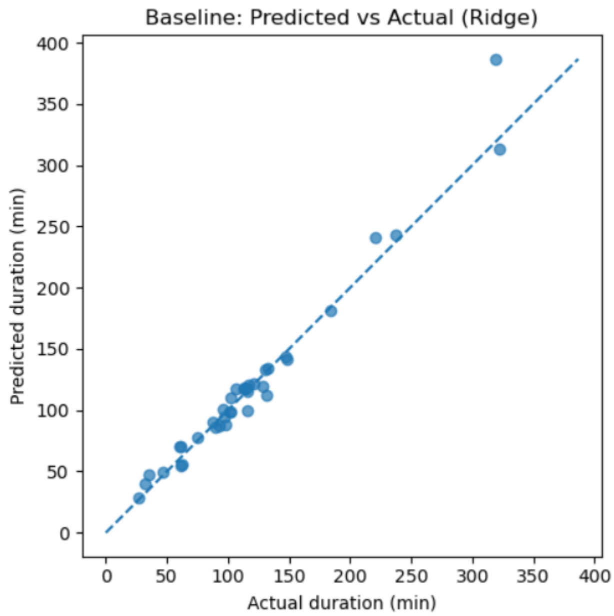
<code>Cross-validation r2 scores: [0.93727729 0.10664602 0.92689275 0.98167289 0.97914234]
Average r2: 0.7863
array([0.93727729, 0.10664602, 0.92689275, 0.98167289, 0.97914234])</code>

- The cross-validation R^2 scores for the model were: [0.937, 0.107, 0.927, 0.982, 0.979]. These results indicate that the model performed well on most folds, with R^2 values close to 1.0, suggesting a strong predictive power.
- The average R^2 score of 0.7863 demonstrates that the model generally explains a high proportion of the variance, but improvements may be needed to ensure more consistent performance across folds.</td>
</tr>
<tr>
<td>3. Benjamin Cole</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. ASMITHRA RAVICHANDRAN</td>
<td>- Correlate environmental factors (Wind speed, Humidity, Temperature) with athlete performance
- Collect and clean sample humidity data
- Link humidity values with athlete performance metrics (speed, stamina)
- Generate correlation matrix and visualisation</td>
<td>• Wrote a Python script to clean and merge humidity + performance dataset.
• Calculated correlation between humidity and athlete speed.
• Generated a scatter plot showing negative correlation (as humidity ↑, speed ↓).
• Correlation Matrix:
<code>(venv) asmithraravichandran@Asmithras-Air humidity_task % cd ~/Desktop/humidity_task
source venv/bin/activate

(venv) asmithraravichandran@Asmithras-Air humidity_task % export MPLCONFIGDIR=./.mplcache
(venv) asmithraravichandran@Asmithras-Air humidity_task % python humidity_correlation.py
Matplotlib is building the font cache; this may take a moment.
Correlation Matrix:
          humidity   speed
humidity 1.000000 -0.997141
speed   -0.997141 1.000000
Saved: humidity_vs_speed.png
(venv) asmithraravichandran@Asmithras-Air humidity_task %</code>
• Scatter plot:

</td>
</tr>
<tr>
<td>5. WIMANSA DESHANI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Chiranth Gowda</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Dhivyaa Lakshimi (224744156)</td>
<td>- Correlate environmental factors (Wind speed, Humidity,</td>
<td>- Cleaned the Garmin dataset (<code>activities_cleaned.csv</code>) and prepared for analysis.</td>
</tr>
</table>

	Temperature) with athlete performance - Completed **Wind Speed** analysis	- Focused on Wind Speed vs Athlete Avg Speed. - Used Google Colab Notebook (`windspeed.ipynb`) to run analysis. - Calculated correlation: - **Pearson = 0.12** - **Spearman = 0.09** - Produced outputs: - **Summary table** (`wind_summary.csv`) - **Scatter plot** (Wind Speed vs Avg Speed with regression line) - **Correlation heatmap** (Wind, AvgSpeed, Temp, Humidity)  
8. Hrithik Joseph (223058217)	Build a predictive model for running duration	* Developed an Accurate Predictive Model for Running Duration I successfully built and implemented a robust regression model that predicts running duration using key input variables such as distance, elevation gain, and pace *Enhanced Model Reliability through Multi-Model Evaluation and Visualization I evaluated multiple machine learning algorithms—including Linear Regression, Ridge Regression, Random Forest, and Gradient Boosting—to identify the best performing model based on RMSE and R^2

<table>
<tr>
<td></td>
<td></td>
<td>metrics. I also created clear visualizations comparing actual and predicted durations, enhancing interpretability and actionable insights for the coaching team.


</td>
</tr>
<tr>
<td>9. Jeswin Joy (224832817)</td>
<td></td>
<td>I loaded the dataset, standardised key columns, checked missing values, explored distributions, and flagged the outliers.
I normalised the unit duration from hours to minutes and distance to kilometres, removed the duplicates and incomplete rows.
I trained two baselines to predict duration in minutes: Linear Regression and Ridge with cross-validated regularisation in a scaling pipeline.
I reported MAE, RMSE, and R^2 and plotted predicted vs actual values.

</td>
</tr>
<tr>
<td>10. Madhav Grover</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11. Nathan Riley</td>
<td>Progressed the predicting sleep efficiency using the sleep efficiency dataset</td>
<td>Sleep Efficiency Prediction
The sleep efficiency dataset tracks athletes and non-athletes sleep and rates the efficiency of that sleep</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>(ongoing) Srihari and I working on our assigned task regarding documentation</td>
<td>by analysing metrics such as no. of exercises performed, alcohol consumed, duration, etc. Using models, I’m attempting to analyse and predict these ratings so it can be used as a tool for athletes to achieve the highest rating possible so it can lead to peak performance by maximising recovery. I have incorporated a couple new models, and having determined RF as the optimal model for the prediction, I’m at the concluding stages where I’m inputting new data which can hopefully strengthen the model after comparing it to some previous entries. I’ll begin looking into new things I can target this week. The task with Srihari is ongoing with discussions still going as well.
<pre>
Mean Absolute Error (MAE): 0.036
Root Mean Squared Error (RMSE): 0.048
R-squared (R^2): 0.874

upd_random_row = upd_columns.sample(n = 1)

print(upd_random_row)

     Age Gender Sleep duration REM sleep percentage Deep sleep percentage \
195   61   0   7.0   23   23

     Light sleep percentage Awakenings Caffeine consumption \
195   54   2.0   50.0

     Alcohol consumption Smoking status Exercise frequency
195   5.0   0   0.0

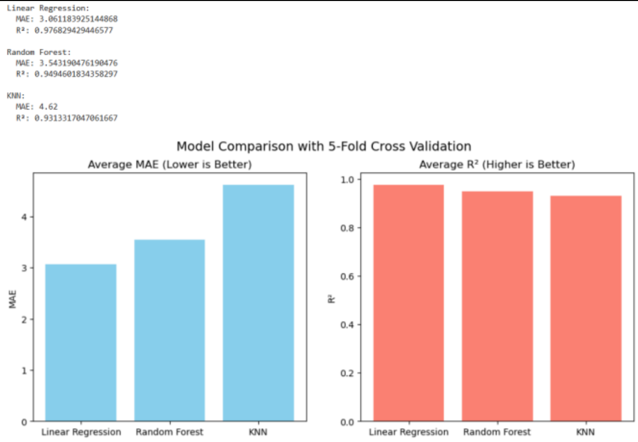
# Create a Dataframe Prediction using: Age = 35, Caff Consumption = 100, Alc Consumption = 2, Ex Freq = 3
upd_rf_prediction = pd.DataFrame(upd_random_row, columns = upd_columns.columns)

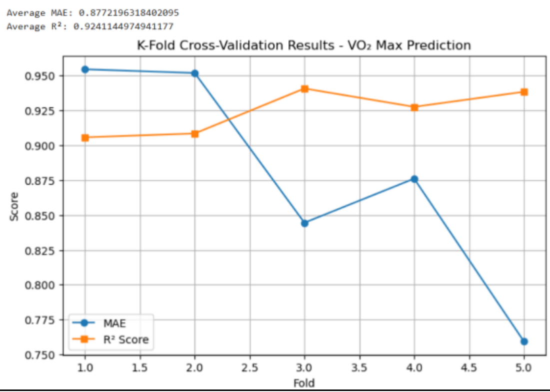
# Predict sleep efficiency
upd_predicted_eff = np.round(upd_best_rf.predict(upd_rf_prediction), 2)

print(upd_predicted_eff)

[0.66]
</pre>
Once again we've gone through the process and fine tuned the RF model for predicting sleep efficiency. After running through the process of selecting a row at random and running the new predicition through the updated Random Forest model, we were given a predicted Sleep Efficiency rating of 0.66. Upon inspection of the table, we go to row 197 (195 + 1 + 1 since the row count starts at 0 and we account for title row) and see the actual Sleep Efficiency rating is 0.67.

A difference of 0.01 is closer again and therefore are even more satisfied with this model when it comes to predictions. We can try inputting our own variables for a prediction or can even try building a model that predicts sleep duration or sleep efficiency based on a couple of variables alone.</td>
</tr>
<tr>
<td>12. Saumya Parasbhai</td>
<td>Perform cross validations on datasets</td>
<td>Cricket Performance Prediction
developed and evaluated a number of prediction models that use batting characteristics like balls faced, strike rate, boundaries, and sixes to estimate the number of runs scored. Five-fold cross-validation was used for KNN, Random Forest, and Linear Regression. With the lowest mean absolute error and the highest R2 score, the results demonstrated that Linear Regression consistently performed better than the others. This demonstrated that the model was dependable for analyzing cricket performance since runs scored were well clarified by linear relationships between batting inputs and results.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td>


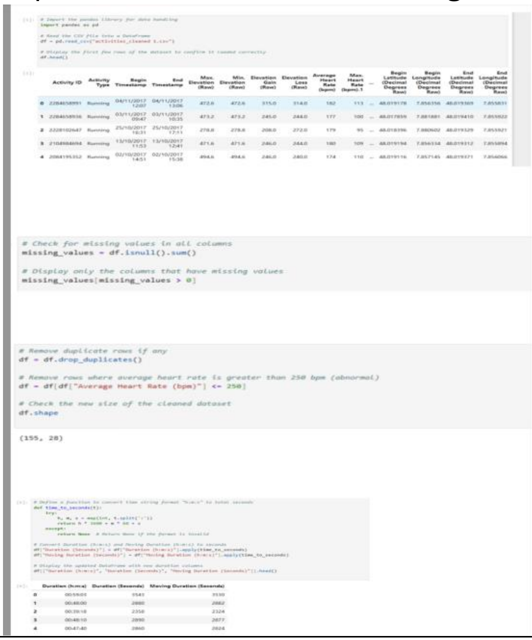
VO_2 Max Prediction
Using physiological and training data from Garmin, a cross-validated regression model was developed to forecast athletes' VO_2 Max values. With a high R2 and a low mean absolute error across folds, the Random Forest model showed good predictive power. Consistent model stability was validated by visualizing fold-wise results, suggesting that the model performs well when applied to athlete data that has not yet been seen. The possibility of using wearable data to track physical activity and provide data-driven performance insights is confirmed by these findings.


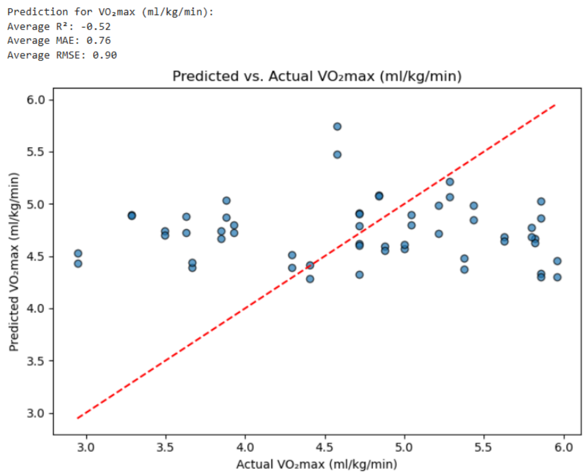
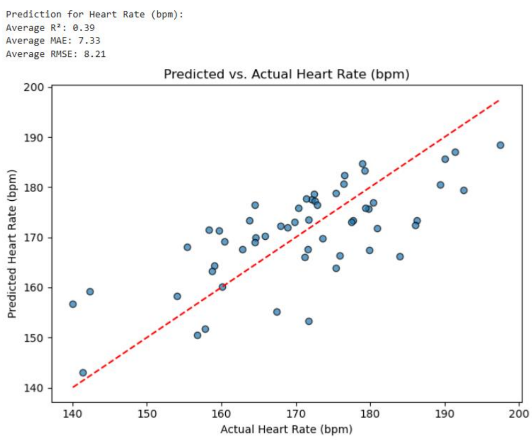
</td>
</tr>
<tr>
<td>13. Scott Ailaiti (224348919)</td>
<td>Exploratory Data Analysis (EDA)

Data Cleaning & Preprocessing

Feature Engineering

Data Transformation (Normalization & Encoding)</td>
<td>I contributed to the data preprocessing pipeline by performing essential cleaning and transformation tasks on the provided running dataset. My responsibilities included loading and previewing the data, checking and confirming the absence of missing values, and removing duplicate rows and extreme outliers (e.g., heart rates above 250 bpm) to ensure data quality. I also implemented a time conversion function that transformed "Duration (h:m:s)" and "Moving Duration (h:m:s)" columns into total seconds to facilitate accurate numerical analysis. These foundational steps laid the</td>
</tr>
</table>

	Dataset Splitting (Train/Test) Implement Initial Model Prototypes Experiment Tracking (Note capture & screenshots)	groundwork for future modeling by ensuring the dataset was clean, consistent, and ready for further exploration and machine learning tasks. 
14. Siddh Parasbhai	Perform Cross Validation on the datasets and get insights.	An end-to-end machine learning pipeline was created to forecast the aerobic capacity (VO_2max) of cyclists based on power output data from 20-kilometer time trials. With an R2 of -0.52, the model showed low efficacy, failing to capture significant physiological correlations between VO_2max and cycling performance. The limited physiological range of VO_2max values (3.0-6.0 ml/kg/min) concealed basic model flaws, even with numerically minor mistakes (MAE: 0.76 ml/kg/min). The main failure point identified by permutation importance analysis was insufficient feature representation, underscoring the necessity of including extra physiological indicators such as heart rate variability, cadence patterns, and individual anthropometric data in order to establish reliable power-to-VO_2max relationships for precise fitness evaluation.

		 created a predictive system that uses power output measurements to estimate the heart rate response in real time during cycling time trials. With clinically relevant error margins (MAE: 7.33 bpm) and moderate explanatory power (R2: 0.39), the model captured fundamental patterns of exertion but lacked individual precision. Power output fluctuations were found to be the main driver by feature importance analysis, which also revealed significant gaps in the accounting for environmental components and cumulative fatigue effects. The methodology has promise for monitoring coarse exertion in training scenarios; nevertheless, in order to attain actionable accuracy for performance optimization, temporal factors such as acute:chronic workload ratios, temperature acclimatization measurements, and individual recovery patterns must be integrated. 
15. Srihari Chandran		
16. Xiangning Wang	**Analyse personal activity metrics to identify fitness pattern**	1. Total_minutes 2. Active_minutes_pct 3. Active_distiance_pct

Identify metrics related to personal activity within provided dataset

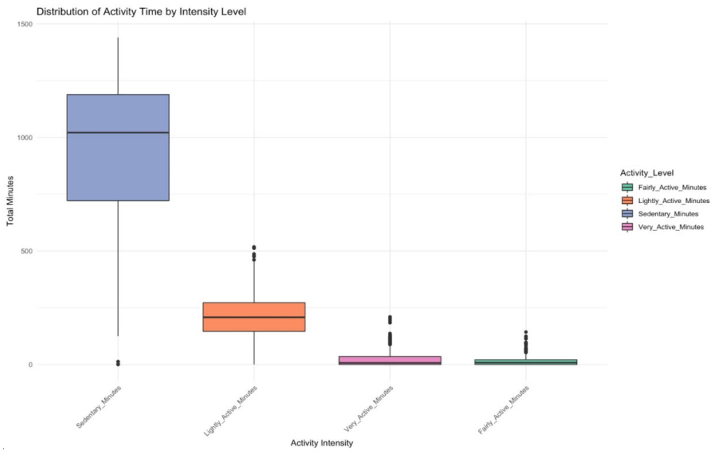
Data summary and exploration, including visualisations and data normalisation

These 3 new variables can be considered as composite variables.

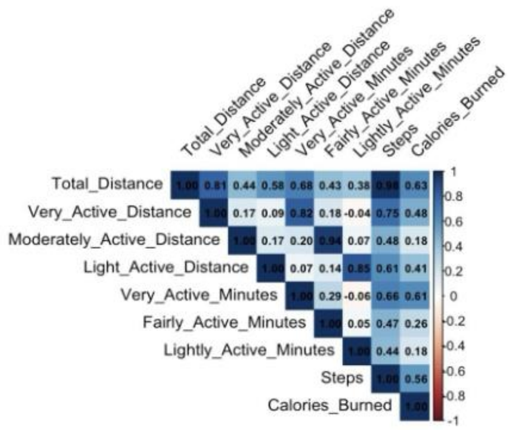
This is the basic summary for activity related variables data

```
> activity_summary
 Total_Distance  Total_minutes    Active_minutes_pct Active_distance_pct     Steps         Calories_Burned
 Min.   : 0.00   Min.   :   2.0   Min.   :  0.00     Min.   : 98.45      Min.   :    4   Min.   :  52
 1st Qu.: 3.37   1st Qu.: 983.5   1st Qu.: 15.25     1st Qu.:100.00      1st Qu.: 4923   1st Qu.:1856
 Median : 5.59   Median :1353.0   Median : 22.29     Median :100.00      Median : 8053   Median :2220
 Mean   : 5.98   Mean   :1203.6   Mean   : 21.80     Mean   : 99.96      Mean   : 8319   Mean   :2361
 3rd Qu.: 7.90   3rd Qu.:1440.0   3rd Qu.: 27.50     3rd Qu.:100.00      3rd Qu.:11092   3rd Qu.:2832
 Max.   :28.03   Max.   :1440.0   Max.   :100.00     Max.   :100.00      Max.   :36019   Max.   :4900
                                                     NA's   :1
```

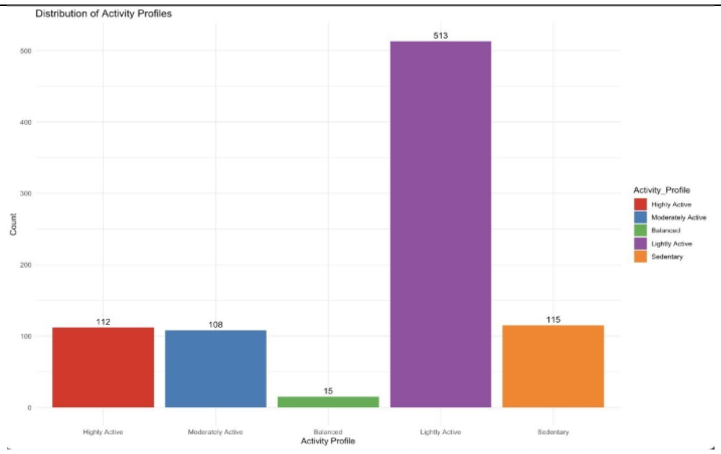
Here is the activity intensity distribution by minutes portrayed in a boxplot:



Here is the correlation matrix between all related variable/metrics



Based on the summary statistics and distribution, we have created the following activity profiles: Highly Active, Moderately Active, Balanced, Lightly Active and Sedentary. Here is the adjusted distribution visualised through bar chart.

<table>
<tr><td></td><td></td><td>
</td></tr>
<tr><td></td><td></td><td></td></tr>
</table>