Activity Level Prediction and Health Insights Using Fitness Tracker Dataset

Supraja Rama Meka[1], Md Amiruzzaman[1], Md. Rajibul Islam[2*] and Rizal Mohd Nor[3]

[1]Department of Computer Science, West Chester University, 700 S High St, West Chester, 19383, PA, USA.
[2*]Department of Electrical and Electronic Engineering, The Hong Kong Polytechnic University, 11 Yuk Choi Road, Hung Hom, Kowloon, Hong Kong S.A.R, China.
[3]Department of Computer Science, International Islamic University, Kuala Lumpur, Malaysia.

*Corresponding author(s). E-mail(s): mdrajibul.islam@polyu.edu.hk;
Contributing authors: suprajarama24@gmail.com; mamiruzzaman@wcupa.edu; rizalmohdnor@iium.edu.my;

Abstract

This study examines the application of machine learning to classify user activity levels using fitness tracker data. The dataset includes one million records with features such as steps, heart rate, sleep hours, active minutes, workout type, weather, and mood. A pipeline was developed involving data preprocessing, feature engineering, SMOTE-based class balancing, and model evaluation through cross-validation. Four supervised learning models such as Random Forest, Logistic Regression, K-Nearest Neighbors, and XGBoost were trained and compared. Overfitting was a key challenge, largely due to overreliance on the steps feature, which led to inflated accuracy and label leakage. To mitigate this, a composite activity level was engineered by combining normalized ***steps***, ***heart_rate_avg***, and ***active_minutes***, resulting in a more informative target variable. Models trained on this revised label showed enhanced generalization and better class-wise performance, with ensemble methods achieving the highest macro F1-scores. Evaluation metrics, including accuracy, precision, recall, and F1-score, were used to assess the models under both original and engineered labeling strategies. Overall, the findings underscore the importance of robust label construction and diverse feature sets in developing effective systems for behavioral data analysis, highlighting the potential of fitness tracker data for personalized health monitoring and lifestyle interventions.

Keywords: Fitness Tracker Data, Machine Learning, Activity Prediction, Feature Engineering, Health Insights

1 Introduction

The rapid proliferation of wearable fitness trackers has revolutionized how individuals monitor their physical activity, offering valuable insights into health and wellness. These devices generate large volumes of data, encompassing various metrics such as steps taken, calories burned, heart rate, and sleep patterns [1]. Analyzing such data not only helps users understand their daily routines but also enables professionals to identify patterns that can improve health outcomes. Predicting activity levels based on fitness tracker data is a crucial step toward achieving personalized fitness recommendations and developing health-focused applications.

This study explores the predictive power of fitness tracker data in determining activity levels categorized as Low, Moderate, or High. The dataset employed for this study includes 1,000,000 entries and features 12 attributes that capture an individual's physical activity, contextual conditions, and subjective mood. Of these, key features such as steps, calories burned, sleep hours, heart rate, workout type, mood, and weather conditions were selected for analysis. Less relevant features, such as user ID and date, were excluded to improve modeling effectiveness.

Despite the promise of fitness tracker data, challenges like class imbalance and overfitting complicate the predictive modeling process. For instance, the ”steps” attribute strongly correlates with activity levels, creating a risk of overfitting and overshadowing the contributions of other features. To address these challenges, this study employs the Synthetic Minority Oversampling Technique [2] to balance the dataset and systematically investigates the impact of excluding the ”steps” feature. Since removing the steps feature led to a significant drop in performance and did not yield the desired generalization, a composite activity score was engineered to provide a more holistic and generalizable representation of user activity.

The study utilizes multiple machine learning algorithms to assess their suitability for activity-level prediction. Random Forest, Logistic Regression, K-Nearest Neighbors (KNN), and XGBoost are evaluated for their performance, with metrics such as accuracy, precision, recall, and F1-scores used for comparison [3]. Furthermore, cross-validation ensures that the results are consistent and generalizable. The insights from this study contribute to the development of predictive systems that leverage fitness tracker data for personalized health monitoring and actionable recommendations. pecifically, features like steps taken and calories burned are direct indicators of physical activity, heart rate reflects exertion, and sleep patterns may influence energy levels—collectively impacting prediction performance and enabling more accurate classification of activity levels.

By exploring the intricate relationships between features and their influence on predictive accuracy, this study lays the groundwork for integrating fitness data into broader health management frameworks. It underscores the importance of diverse features and balanced datasets while offering a roadmap for future advancements in this domain.

2 Related Works

The application of machine learning techniques to fitness tracker data has garnered significant attention, driven by the potential to extract meaningful insights for health monitoring and activity prediction. This section highlights prior works that relate to the current study and establishes the methodological and contextual foundation for this research.

Breiman (2001) introduced Random Forests, a robust ensemble learning method recognized for its high accuracy and interpretability. Random Forests are particularly effective in handling high-dimensional datasets and provide valuable feature importance metrics, which have been integral to understanding the contribution of key attributes like steps in our study [4]. Similarly, Friedman (2001) proposed the Gradient Boosting Machine, foundational to modern gradient-boosted models like XGBoost. Its ability to manage complex data distributions and imbalances aligns with the objectives of this research to enhance classification accuracy [5].

Chawla et al. (2002)[2] introduced SMOTE (Synthetic Minority Oversampling Technique), which addresses class imbalance by generating synthetic samples for underrepresented classes. This approach is particularly relevant to our study, where class imbalance among activity levels (e.g., Moderate and Low) was a significant challenge. The use of SMOTE in preprocessing ensured fairer predictions and improved model performance.

Rawat and Mishra (2022) provided a comprehensive review of methods for handling class imbalance in classification tasks, emphasizing the critical role of data-level techniques like SMOTE. Their findings align closely with the preprocessing strategies employed in this study, highlighting the necessity of addressing class imbalance for more accurate and generalizable results [6]. Exploratory data analysis (EDA) is foundational to understanding relationships within fitness tracker data. Joshi (2023) conducted an EDA on Fitbit data, uncovering insights into feature relationships such as steps, calories burned, and sleep hours. This aligns with the EDA process in our research, where key features were evaluated for their impact on activity prediction models [7]. Veeraiah et al. (2024) explored the use of machine learning for fitness tracking, demonstrating how data analytics can provide actionable health insights. Their work focused on integrating diverse features like activity levels, sleep, and calories to create predictive models. This study builds upon their findings by leveraging similar features to predict activity levels and provide personalized health insights [8].

Müller et al. (2024) investigated fitness activity recognition using IMU-based sensors and convolutional neural networks (CNNs) for time series classification. While their study utilized deep learning, our research employs simpler, interpretable models like Random Forest and XGBoost to predict activity levels. Both works highlight the significance of feature extraction and engineering in fitness data analysis [9].

Building on these prior works, this study integrates advanced machine learning algorithms, class balancing techniques, and exploratory data analysis to predict activity levels using fitness tracker data. By employing Random Forest and XGBoost for interpretability and leveraging SMOTE for class balancing, this research contributes to the development of robust, generalizable, and actionable systems for personalized health monitoring. These related works collectively provide the theoretical and methodological foundation for the approaches and findings in this study.

3 Methodology

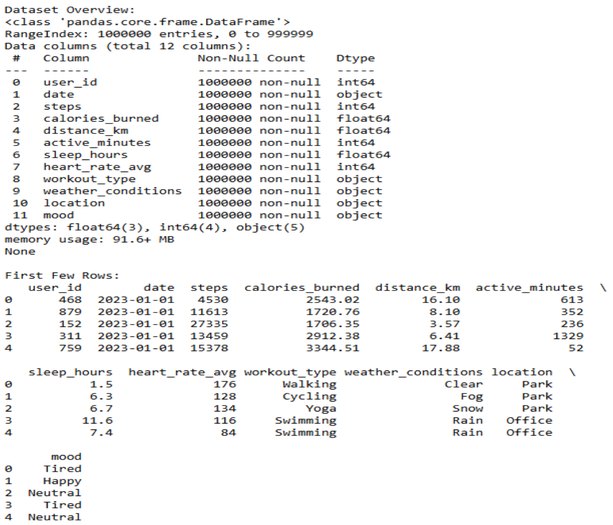
This section outlines the step-by-step approach used for data preprocessing, feature engineering, EDA, model training, and evaluation.

3.1 Data Description

The Fitness Tracker Dataset from Kaggle [10] holds 1,000,000 records with 12 features tracking user activity and related details. The dataset has a User ID and Date to track the participant and the day of the data collection. It registers physical activity in terms of Steps—total steps taken, Calories Burned—an estimation of energy used, and Distance (km)—distance traveled in kilometers. Also included are Active Minutes, representing the total amount of time spent being active throughout the day.

Other health-related details include the number of Sleep Hours, showing how long someone slept, and Heart Rate (Average), referring to the average heart rate that day. Also, the type of activity is mentioned in the Workout Type feature, such as Walking or Yoga. More contextual data is found with Weather Conditions—like Clear or Rain—and Location—like Park or Office—where the activity occurred. It also has Mood, where users logged how they are feeling, such as Happy or Neutral.

Missing values were filled using a forward-fill method, then data was cleaned and prepared. A new feature, Activity Level, was added to the dataset, grouping the activity intensity into three classes: Low, Moderate, and High, based on the number of steps. Finally, categories like workout type and mood were converted to numbers using one-hot encoding in order to make the data easily analyzable.



```
Dataset Overview:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000000 entries, 0 to 999999
Data columns (total 12 columns):
 #   Column              Non-Null Count    Dtype
---  ------              --------------    -----
 0   user_id             1000000 non-null  int64
 1   date                1000000 non-null  object
 2   steps               1000000 non-null  int64
 3   calories_burned     1000000 non-null  float64
 4   distance_km         1000000 non-null  float64
 5   active_minutes      1000000 non-null  int64
 6   sleep_hours         1000000 non-null  float64
 7   heart_rate_avg      1000000 non-null  int64
 8   workout_type        1000000 non-null  object
 9   weather_conditions  1000000 non-null  object
 10  location            1000000 non-null  object
 11  mood                1000000 non-null  object
dtypes: float64(3), int64(4), object(5)
memory usage: 91.6+ MB
None

First Few Rows:
   user_id        date  steps  calories_burned  distance_km  active_minutes  \
0      468  2023-01-01   4530          2543.02        16.10             613
1      879  2023-01-01  11613          1720.76         8.10             352
2      152  2023-01-01  27335          1706.35         3.57             236
3      311  2023-01-01  13459          2912.38         6.41            1329
4      759  2023-01-01  15378          3344.51        17.88              52

   sleep_hours  heart_rate_avg workout_type weather_conditions location  \
0          1.5             176      Walking              Clear     Park
1          6.3             128      Cycling                Fog     Park
2          6.7             134         Yoga               Snow     Park
3         11.6             116     Swimming               Rain   Office
4          7.4              84     Swimming               Rain   Office

      mood
0    Tired
1    Happy
2  Neutral
3    Tired
4  Neutral
```

Fig. 1 Dataset Overview

3.2 Preprocessing and Feature Engineering

Preprocessing is one crucial step in any data project [3], and the handling of missing values in this study was first in line. For filling gaps in the data, forward-filling by the last recorded value was applied. It ensures that the data is constant and complete and does not include any false or wrong information.

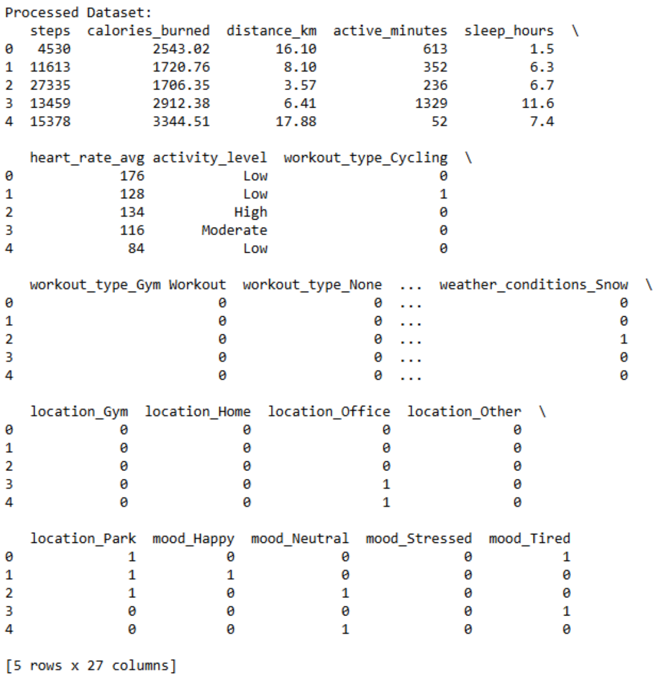
The core component of the preprocessing phase was the construction of the *activity_level* feature [11], which served as the primary target variable for classification. Initially, this feature was created using a simple threshold-based rule derived solely from the steps column. According to this method, days with fewer than 5,000 steps were categorized as Low activity, between 5,000 and 10,000 steps as Moderate, and above 10,000 steps as High activity. While this approach was intuitive and aligned with common fitness tracking benchmarks, it introduced a significant modeling limitation: the labels were directly dependent on a single feature, potentially causing overfitting and reducing the model's ability to generalize.

To address this, a more advanced and data-driven version of *activity_level* was engineered. This involved calculating a composite activity score by integrating multiple physical activity indicators. Specifically, the features *steps*, *heart_rate_avg*, and *active_minutes* were normalized using min-max scaling to ensure uniform contribution. These normalized values were then combined using a weighted sum: $activity\_score = 0.5 \times \text{normalized}(\text{steps}) +$

$0.3 \times \text{normalized}(heart\_rate) + 0.2 \times \text{normalized}(active\_minutes)$. This formula was designed to capture a broader perspective of physical exertion by considering both movement and intensity.

The continuous *activity_score* was then converted into categorical *activity_level* labels using quantile-based binning. The data was divided into three equal-sized groups representing Low, Moderate, and High activity levels. This method not only ensured class balance but also removed the overreliance on any single predictor, offering a more realistic representation of user activity.

Finally, categorical variables such as *workout_type*, *weather_conditions*, *location*, and *mood* were transformed using one-hot encoding, converting them into a numerical format suitable for machine learning algorithms. Irrelevant columns such as *user_id* and *date*, which do not carry predictive value, were removed to reduce noise and prevent data leakage. These preprocessing steps laid the foundation for training robust and interpretable models that reflect real-world activity patterns more accurately.



```
Processed Dataset:
   steps  calories_burned  distance_km  active_minutes  sleep_hours  \
0   4530          2543.02        16.10             613          1.5
1  11613          1720.76         8.10             352          6.3
2  27335          1706.35         3.57             236          6.7
3  13459          2912.38         6.41            1329         11.6
4  15378          3344.51        17.88              52          7.4

   heart_rate_avg activity_level  workout_type_Cycling  \
0             176            Low                     0
1             128            Low                     1
2             134           High                     0
3             116       Moderate                     0
4              84            Low                     0

   workout_type_Gym Workout  workout_type_None  ...  weather_conditions_Snow  \
0                         0                  0  ...                        0
1                         0                  0  ...                        0
2                         0                  0  ...                        1
3                         0                  0  ...                        0
4                         0                  0  ...                        0

   location_Gym  location_Home  location_Office  location_Other  \
0             0              0                0               0
1             0              0                0               0
2             0              0                0               0
3             0              0                1               0
4             0              0                1               0

   location_Park  mood_Happy  mood_Neutral  mood_Stressed  mood_Tired
0              1           0             0              0           1
1              1           1             0              0           0
2              1           0             1              0           0
3              0           0             0              0           1
4              0           0             1              0           0

[5 rows x 27 columns]
```

Fig. 2 Preprocessed Dataset.

3.3 Data Visualization

Data visualization provided insights into the structure and distribution of the dataset, helping to identify patterns and inform subsequent modeling decisions.

- Activity Level Distribution: A count plot was used to visualize the distribution of activity levels (Fig 3). It was found that there is a class imbalance; most of the entries are in the category ”High” for the activity level. This caused the need for applying techniques such as Synthetic Minority Oversampling Technique (SMOTE) when building the model to make sure all levels of activities are fairly represented.

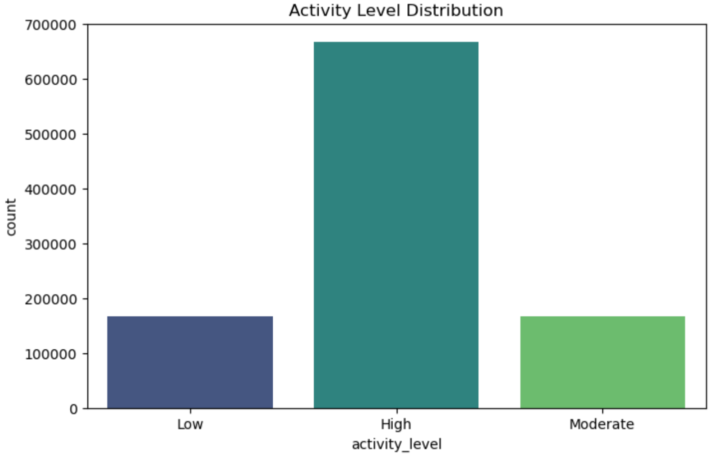


Fig. 3 Activity Level Distribution.

- Correlation heatmap: A heatmap was created to show the relationship between features. The heatmap (Fig 4) showed strong dependencies, such as the relationship between ”steps” and ”activity level”. This plot helped a lot in finding the dominant features and understanding the underlying structure of the dataset—something that guided decisions like feature selection and handling of multicollinearity.
- Total Steps Per Day: The line chart (Fig 5) represents the daily pattern of the sum of steps taken over time, showing ups and downs with spikes and dips in between. It gives clarity on the temporal patterns, including periods of activity or rest, which might correspond to external factors such as weekends, holidays, or weather conditions. Such visualization is useful during feature engineering—to come up with features based on time, such as averages over a week or trends over a season, to improve the model’s prediction. It further helps in the evaluation of class imbalances, as the periods of high or low activity may correspond to some specific activity levels: Low, Moderate, High. It gives insight into the behavior of users and answers whether, in reality, model predictions conform to the actual world patterns, increasing the study’s accuracy and interpretability.

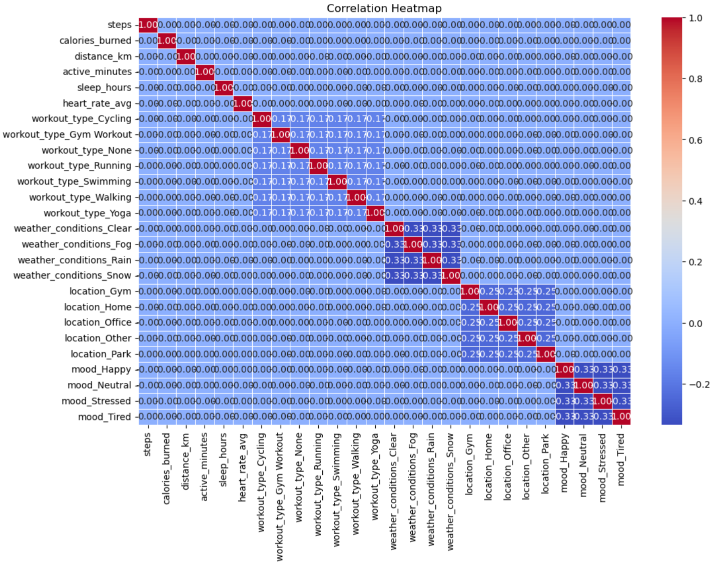


Fig. 4 Correlation Heatmap.

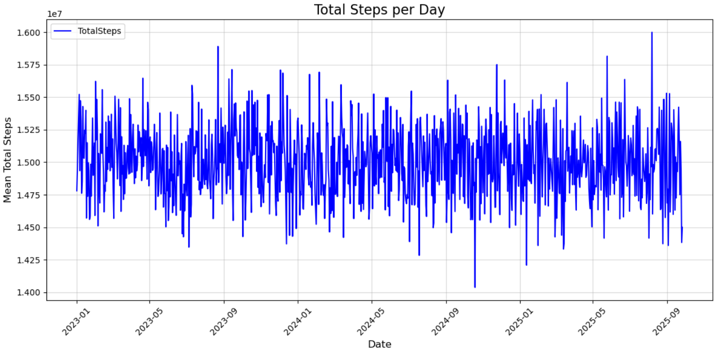


Fig. 5 Total Steps Per Day

3.4 Machine Learning Models

This study used four different machine learning models to examine their efficacy in predicting activity level, each possessing strengths and weaknesses. The Random Forest model is an ensemble method that combines multiple decision trees [12]. It is well-suited for handling high-dimensional data and capturing complex, non-linear patterns. In this study, Random Forest effectively modeled activity levels using composite feature-based labels. It handled class imbalance well, especially when combined with SMOTE. The model also provided useful feature importance scores, which helped interpretability.

The Logistic Regression model is a linear approach for multiclass classification [13]. It finds the best linear boundaries to separate different activity levels. It was able to classify activity levels using the composite labels, but struggled to capture complex relationships between features, as it relies on straight-line decision boundaries. While the model is simple and interpretable, it is less effective for data with nonlinear patterns. Its performance benefits most in datasets where class separation is clearly linear.

It had been tested on the KNN model, a distance-based nonparametric classifier, both in standard and distance-weighted forms [14]. It was applied to classify activity levels using the engineered composite labels. The model captured local relationships well but remained sensitive to feature scaling and computational cost. Although it achieved strong performance, it is less scalable for larger datasets due to its instance-based nature.

Lastly, the XGBoost model, which is a gradient boosting algorithm optimized for speed and accuracy [15], showed very strong results across the multiple scenarios. It was trained using the composite activity level labels and demonstrated excellent performance in classifying activity levels. Its ability to include regularization and analyze feature importance made it highly effective.

These models all helped to give very insightful feature importances while revealing the trade-offs that go with interpretability, computational cost, and prediction accuracy.

3.5 Model Evaluation

Model assessment was carried out through various metrics and techniques for performance and generalization evaluation. Initially, accuracy was used as a measure of general correctness when the activity level was derived solely from the steps feature. This approach resulted in exceptionally high accuracy across all models but raised concerns about overfitting and label-feature dependency.

To address this, an alternative version of *activity_level* was created using a composite score based on normalized values of steps, heart rate, and active minutes. With this new label, the models continued to perform well, but the overall accuracy was slightly reduced—offering a more realistic reflection of generalization.

Along with accuracy, the performance in each activity level was evaluated based on precision, recall, and F1-score. In the steps-based version, precision and recall were near perfect for all classes, especially for dominant ones like High activity. However, when the composite score was used, the evaluation metrics remained strong across all models, with more balanced scores for Moderate and Low classes—indicating improved fairness and reduced overfitting.

Confusion matrices were also used for visualizing the true vs. predicted labels, helping to identify patterns of misclassification. With steps-only labels, misclassifications were minimal but largely due to the direct overlap between the label and the input feature. With composite labels, the confusion matrices reflected more realistic class separation, where minor misclassifications were observed across all three activity levels.

In order to ensure generalizability of the models, cross-validation with 5-fold splits was carried out in both settings. This increased the consistency of performance across different data subsets and helped confirm that ensemble models like Random Forest and XGBoost remained reliable even with the new composite labeling approach.

4 Experiments and Results

4.1 Baseline Results

Preliminary tests using the entire dataset and including the feature "steps" were run in order to get a baseline performance. The models achieved remarkably high accuracy, most likely due to the strong correlation between "steps" and activity levels and raised concerns about models' potential overfitting.

- The Random Forest model gave an accuracy of 100%. In classifying the activity levels—High, Moderate, and Low—it shows exceptional predictive performance. The classification table (Table 1) showed that the model performed with precision, recall, and F1-scores of 1.00 for all classes, indicating no misclassification. Similarly, both the macro and weighted averages were 1.00, showing consistency across all activity levels and not influenced by class size. This perfect outcome points out that the model relied heavily on the "steps" feature, which does raise some concern about overfitting and generalizing to new data. However, this model underscores the importance of the "steps" feature in the accurate prediction of activity levels.

Table 1 Classification Report for Random Forest Model

Class	**Precision**	**Recall**	**F1-Score**	**Support**
High	1.00	1.00	1.00	133,199
Low	1.00	1.00	1.00	33,603
Moderate	1.00	1.00	1.00	33,198
Accuracy			**1.00**	**200,000**
Macro Avg	1.00	1.00	1.00	200,000
Weighted Avg	1.00	1.00	1.00	200,000

- The XGBoost model achieved an accuracy of 99.72%, showing its strong ability to deal with large datasets. From the classification report(Table 2), one can see that the model provided very good results, with precision, recall, and F1-score all being close to 1.00 for each class, which means it gave very accurate predictions. Class 0 and 1 had perfect scores, while class 2 scored a little lower at 0.99, indicating slight misclassifications in that class. The macro and weighted averages were approximately 1.00, meaning that performance was consistent over all activity levels. The result showed that XGBoost is efficient in handling complex data relationships; however, slight misclassifications in class 2 indicate further analysis to make the class predictions more accurate.

Table 2 Classification Report for XGBoost Model

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0 (High)	1.00	1.00	1.00	133,199
1 (Moderate)	1.00	1.00	1.00	33,603
2 (Low)	0.99	0.99	0.99	33,198
Accuracy			**0.9972**	**200,000**
Macro Avg	1.00	1.00	1.00	200,000
Weighted Avg	1.00	1.00	1.00	200,000

- With this approach, the K-Nearest Neighbors (KNN) model achieved a high accuracy of 99.87%, hence proving to be effective in classifying the activity level. As shown in the classification report(Table 3), it resulted in precision, recall, and F1-scores of 1.00 for classes 0 and 1; class 2 scored 0.99 in precision and obtained a score of 1.00 for both recall and F1-score. The macro average and weighted average results also scored a value close to 1.00, hence showing a balanced performance among all classes. These results mean that KNN delivers highly accurate predictions, although through a distance-based computation, which might face computational challenges if the dataset becomes large. Although it has these features, the model is consistent among all classes hence reliable for this kind of dataset.

Table 3 Classification Report for KNN Model

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0 (High)	1.00	1.00	1.00	133,199
1 (Moderate)	1.00	1.00	1.00	33,603
2 (Low)	0.99	1.00	1.00	33,198
Accuracy			**0.9987**	**200,000**
Macro Avg	1.00	1.00	1.00	200,000
Weighted Avg	1.00	1.00	1.00	200,000

- The Logistic Regression model scored an accuracy of 92.15% (Table 4), which is fairly good but was a bit lower in comparison to other models. It employed linear relationships of the data for good performance, mostly for classes 0 and 1, with precision, recall, and F1-score values over 0.90. For class 2, the precision was lower (0.72), but recall was higher (0.87), meaning that the model was better at identifying positive cases for the latter but worse in terms of false positives. It had a macro average and weighted average of 0.88 and 0.93, respectively, to further indicate general reliability while reflecting difficulties in fitting more complex and nonlinear patterns than other models.

4.2 Addressing Overfitting

- To evaluate the dependency on the *steps* feature and address potential overfitting, all models were retrained after excluding this feature from the dataset. The removal resulted in a significant decline in performance across all classifiers, particularly in predicting Moderate and Low activity levels.

Table 4 Classification Report for Logistic Regression Model

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0 (High)	0.98	0.93	0.96	133,199
1 (Moderate)	0.94	0.92	0.93	33,603
2 (Low)	0.72	0.87	0.79	33,198
Accuracy			**0.9215**	**200,000**
Macro Avg	0.88	0.91	0.89	200,000
Weighted Avg	0.93	0.92	0.92	200,000

The Random Forest and Logistic Regression models saw their accuracy drop to approximately 65%, performing relatively well for High activity classification but failing to distinguish between the other two classes. The K-Nearest Neighbors (KNN) model experienced the most severe degradation, with accuracy falling to 35.87%, reflecting its strong reliance on *steps* for distance-based classification. Although XGBoost proved to be the most resilient, achieving 66.59% accuracy, it also struggled to correctly identify Moderate and Low activity levels. These findings highlight the critical role of the *steps* feature and the risk of overfitting when activity level labels are derived from a single dominant variable.

Table 5 Classification Performance Without *steps* Feature

Model	**Accuracy**	**F1-Score (High)**	**F1-Score (Moderate)**	**F1-Score (Low)**	**Macro F1-Score**
Random Forest	65.87%	0.79	0.01	0.01	0.27
Logistic Regression	65.27%	0.79	0.02	0.03	0.28
K-Nearest Neighbors	35.87%	0.49	0.22	0.22	0.31
XGBoost	66.59%	0.80	0.00	0.00	0.27

4.3 Cross-Validation

A 5-fold cross-validation was conducted to evaluate the consistency and generalizability of the models (Table 6). Random Forest demonstrated outstanding stability with a mean accuracy of 99.99% and an extremely low standard deviation of 7.49e-06 (Table 6), indicating minimal variation across folds. Similarly, XGBoost showed robust performance with a mean accuracy of 99.83% and a standard deviation of 0.10%, confirming its reliability for large datasets. The KNN model performed consistently when the "steps" feature was included, achieving a mean accuracy of 99.95% with a low standard deviation of 7.35e-05, but showed greater variability without the feature. Logistic Regression displayed a mean accuracy of 91.43%, but its higher standard deviation of 0.34% highlighted its sensitivity to imbalanced classes. Overall, the results emphasize that models like Random Forest and XGBoost perform well with key features like "steps", but their accuracy declines significantly without it.

4.4 Performance with Composite Activity Level

Initial attempts to reduce overfitting by removing the steps feature and applying cross-validation did not yield the expected improvements in generalization.

Table 6 Cross Validation

Model	**Mean Accuracy**	**Standard Deviation**
Random Forest	0.99999625	7.49e-06
Logistic Regression	0.914365	0.0034
KNN	0.999513	7.35e-05
XGBoost	0.998265	0.0010

To adress this, we engineered a new labeling strategy using a composite activity score. This score was created using a weighted combination of three normalized indicators: *steps*, *heart_rate_avg*, and *active_minutes*. These features were selected to capture not just movement, but also physical exertion and engagement duration. The composite score was calculated using the formula:

$$\begin{aligned}\text{activity\_score} = 0.5 \times \text{normalized(steps)} + 0.3 \times \text{normalized(heart\_rate)} \\ + 0.2 \times \text{normalized(active\_minutes)}\end{aligned} \tag{1}$$

This continuous score was then discretized into three balanced activity levels — Low, Moderate, and High — using quantile-based binning. This labeling approach reduced the risk of feature-label leakage and created more evenly distributed classes for training. With the new activity level labels, all models showed marked improvement in balance and generalization.

- The Random Forest model achieved an accuracy of 98.40%, with near-perfect precision, recall, and F1-scores across all three classes. It performed especially well in distinguishing Moderate activity, which had previously suffered in the step-based labeling approach. This result highlights the model's ability to capture non-linear feature interactions and benefit from balanced target classes. The model maintained consistency across classes, making it a strong and stable performer.
- Logistic Regression reached an accuracy of 70.97%, showing significant improvement over its performance without steps. It performed reasonably well in classifying High and Moderate activity levels but showed lower precision and recall for the Low class, reflected in an F1-score of 0.56. This indicates that while the model benefits from the composite label, its linear nature still limits its ability to separate complex class boundaries. Nonetheless, it demonstrated better generalization compared to when trained on the step-derived labels.
- The KNN model demonstrated robust performance with an accuracy of 90.02%. It handled all three activity levels well, with F1-scores above 0.85 for each class. The balanced label distribution improved its neighborhood-based predictions, particularly for the Moderate and Low classes that previously suffered from class imbalance. KNN's success here confirms that when given well-distributed, meaningful features, it can effectively detect nuanced patterns based on proximity.

- XGBoost emerged as the top-performing model with an accuracy of 99.45% and F1-scores near or equal to 1.00 across all classes. Its gradient boosting framework and built-in regularization allowed it to fully leverage the composite features while avoiding overfitting. The model's high performance confirms its suitability for multi-class classification problems involving engineered and imbalanced data, especially when rich features are available.

Table 7 Classification Performance with Composite Activity Level

Metric	**Random Forest**	**Logistic Regression**	**K-Nearest Neighbors**	**XGBoost**
Accuracy	98.40%	70.97%	90.02%	99.45%
Precision (High)	0.99	0.72	0.93	1.00
Recall (High)	0.99	0.80	0.92	1.00
F1-Score (High)	0.99	0.76	0.92	1.00
Precision (Moderate)	0.97	0.83	0.92	1.00
Recall (Moderate)	0.98	0.78	0.93	1.00
F1-Score (Moderate)	0.98	0.80	0.93	1.00
Precision (Low)	0.99	0.58	0.85	0.99
Recall (Low)	0.99	0.55	0.85	0.99
F1-Score (Low)	0.99	0.56	0.85	0.99
Macro F1-Score	0.98	0.71	0.90	0.99

Discussion

Our analysis underscores the critical limitations of relying solely on step count for activity recognition. While ensemble models such as XGBoost and Random Forest initially achieved near-perfect accuracy using step-based labels, performance dropped markedly—particularly for Moderate and Low activity levels—when the step feature was removed. Simpler models like Logistic Regression and KNN were even more affected, highlighting both overfitting and a lack of robustness.

To address this, we introduced a composite activity label derived from normalized steps, heart rate, and active minutes. This multi-feature label improved class balance and model generalization. Retrained models showed substantial performance gains, with XGBoost and Random Forest maintaining high accuracy (99.45% and 98.40%, respectively), and KNN and Logistic Regression also improving notably.

SMOTE was used to address class imbalance, yielding moderate benefits, though simpler models remained sensitive to synthetic noise. Cross-validation confirmed the consistency and reliability of ensemble models across folds. These findings highlight the importance of robust label engineering and feature diversity in developing interpretable and deployable activity recognition models.

Conclusion

This study demonstrated the effectiveness of using fitness tracker data to predict user activity levels and offered critical insights into model behavior under various feature configurations. Initially, the steps feature emerged as the dominant predictor, driving high accuracy across all models. However, its removal revealed significant performance degradation—particularly in identifying Moderate and Low activity levels—highlighting issues of overfitting and feature-label dependency.

To overcome this, a composite activity level was engineered using normalized *steps*, *heart_rate_avg*, and *active_minutes*, resulting in a more balanced and representative target variable. Models trained on this new label exhibited improved generalization and class-wise performance, especially ensemble methods like Random Forest and XGBoost, which achieved near-perfect accuracy and consistency. While simpler models such as Logistic Regression and KNN showed improvement, their limitations in handling non-linear relationships and reduced feature sets remained apparent.

SMOTE and cross-validation further supported the robustness of ensemble models in handling imbalanced data, but also underscored the limitations of synthetic oversampling when underlying feature richness is insufficient. These findings emphasize the importance of both robust feature engineering and meaningful label construction in building reliable activity prediction systems.

Future work should focus on integrating additional contextual and physiological data, improving model interpretability, and enhancing adaptability across diverse user populations. This study provides a foundation for developing intelligent, personalized fitness monitoring tools capable of supporting healthier behaviors and informed lifestyle decisions.

5 Conflict of interest

The authors declare that they have no conflict of interest.

References

[1] Fuller, D., Colwell, E., Low, J., Orychock, K., Tobin, M.A., Simango, B., Buote, R., Van Heerden, D., Luan, H., Cullen, K., Slade, L., Taylor, N.G.A.: Reliability and validity of commercially available wearable devices for measuring steps, energy expenditure, and heart rate: Systematic review. JMIR mHealth and uHealth **8**(9), 18694 (2020). https://doi.org/10.2196/18694

[2] Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research **16**, 321–357 (2002). https://doi.org/10.1613/jair.953

[3] Kuhn, M., Johnson, K.: Applied Predictive Modeling. Springer, ??? (2013). https://doi.org/10.1007/978-1-4614-6849-3. https://link.springer.com/book/10.1007/978-1-4614-6849-3

[4] Breiman, L.: Random forests. Machine Learning **45**(1), 5–32 (2001). https://doi.org/10.1023/A:1010933404324

[5] Friedman, J.H.: Greedy function approximation: A gradient boosting machine. Annals of Statistics **29**(5), 1189–1232 (2001). https://doi.org/10.1214/aos/1013203451

[6] Rawat, S.S., Mishra, A.K.: Review of methods for handling class-imbalanced in classification problems. arXiv preprint arXiv:2211.05456 (2022)

[7] Joshi, V.: Exploratory Data Analysis on Fitbit Fitness Tracker Data. https://medium.com/@vishnu.joshi7521.vj/exploratory-data-analysis-on-fitbit-fitness-tracker-data-80811a0c1a92 (2023)

[8] Veeraiah, V., Ramesh, J.V.N., Koujalagi, A., Talukdar, V., Namdev, A., Gupta, A.: Health fitness tracker system using machine learning based on data analytics. In: Marriwala, N.K., Dhingra, S., Jain, S., Kumar, D. (eds.) Mobile Radio Communications and 5G Networks. MRCN 2023. Lecture Notes in Networks and Systems, vol. 915, pp. 751–759. Springer, ??? (2024). https://doi.org/10.1007/978-981-97-0700-3_57

[9] Müller, P.N., Müller, A.J., Achenbach, P., Göbel, S.: Imu-based fitness activity recognition using cnns for time series classification. Sensors **24**(3), 742 (2024). https://doi.org/10.3390/s24030742

[10] Smayan, A.: Fitness Tracker Dataset. https://www.kaggle.com/datasets/arnavsmayan/fitness-tracker-dataset. Accessed: 2023-12-01 (2023)

[11] Kim, B.-W.: Data preprocessing methods - strategies and best practices: Investigating strategies and best practices for preprocessing data, including cleaning, transformation, and feature engineering. Asian Journal of Machine Learning Research and Applications **2**(1) (2023)

[12] Breiman, L.: Random forests. Machine Learning **45**(1), 5–32 (2001). https://doi.org/10.1023/A:1010933404324

[13] Hosmer, D.W., Lemeshow, S., Sturdivant, R.X.: Applied Logistic Regression, 3rd edn. John Wiley & Sons, ??? (2013). https://doi.org/10.1002/9781118548387. https://doi.org/10.1002/9781118548387

[14] Cover, T., Hart, P.: Nearest neighbor pattern classification. IEEE Transactions on Information Theory **13**(1), 21–27 (1967). https://doi.org/10.1109/TIT.1967.1053964

[15] Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. In:

Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794 (2016). https://doi.org/10.1145/2939672.2939785. https://doi.org/10.1145/2939672.2939785