Coursework: Application of Machine Learning Methods to Real-World Data

Youssef Ghoname

Lynda Djellouli

1. Methodological Approach

The work accomplished in this paper aims to give an answer to the Human Activity Recognition problem through the development of a classification model taking into consideration high-dimensional data distinguished by its imbalanced and noisy characteristics. The developed pipeline uses sensor data, integrating classical ML with advanced signal processing, feature extraction, visualisation interpretability and corrective logic and tools. To ultimately classify with a high enough accuracy six activities (*Walking, Upstairs, Downstairs, Sitting, Standing, and Jogging*).

The complexity of the task is correlated with its importance. The dataset is imbalanced because of the unpredictable actions of everyone. Activities like walking, siting or jogging differ from an individual to another depending on their level of physical capacities or even their age. Considering this set of data was tricky since it is overwhelmed with walking and jogging samples, isolating on the way other activities as standing or downstairs. In addition to that, multiple activities share the same range of values for different features, on great example is the impact the similarities between walking, upstairs and downstairs can have on the dataset and the difficulties it generates for the model to make accurate predictions. Furthermore, over 80 features (columns) are available encompassing sensor or movement distinctive attributes, therefore proper features selection and enrichment is key to the success of the model in its predictions.

Even if the dataset is rich in features, its imbalanced classes contribute to its complexity. However, since both sensor data and metadata (statistical data extracted from sensor data) are available, nourishing the dataset with more significative features is feasible in theory and in practice. The question this study tries to answer is of great importance given its real-world applicability in many aspects of our everyday life or field specific like fitness or healthcare.

This project aims into building a high-performance model with an overall satisfactory balanced accuracy to accommodate the minority class sensitivity, mostly downstairs and upstairs. Incorporating post prediction correction to strengthen the overall score of the model. To address the outlined challenges, a comprehensive pipeline was implemented involving:

Final Approach (Multi-Stage)

Stage	Technique
Signal Preprocessing	Custom feature engineering from raw signals
Feature Selection	PCA + ANOVA
Class Balancing	Class-specific
Model Training	Hyperparameter-tuned LightGBM
Final Output	Confident predictions + Kaggle submission

Table 1: Different component of the pipeline (model)

This pipeline was thoroughly selected following multiple testing, experiments, validation and optimization steps to improve both global/balanced accuracy and per class accuracy and recall. Aiming to find the right classifications solutions, many models were tested, and features extracted:

A. Raw Feature Enrichment: This project is designed to be used for health-related questions. Therefore, biomechanical insights were extracted from the time series files to meet this purpose completing the provided metadata, and it was possible using specific libraries (SciPy, StatsModels, and NumPy). We extracted more than 60 dynamic, spectral, and statistical features including: jerk, spectral entropy, zero-crossing rate, Approximate entropy, autocorrelation, coefficient of variation, Axis-wise skew, kurtosis, MAD, and correlations. These features simulate biomechanical properties (step impact, oscillation, stride irregularity), enhancing separability between dynamic activities like Upstairs/Downstairs.

B. Dual-Stage Feature Selection: First ANOVA F-statistics (SelectKBest) to retain features statistically associated with class variance. Then SHAP analysis based on LightGBM's tree structure to identify class-specific informative features like z_skew, jerk_std or corr_yz. This process retained important yet statistically non-obvious features crucial for inter-class discrimination. Ultimately, combining filter and explainability-based feature selection allows both global relevance and localized activity interpretability.

C. LightGBM with Oversampling: Resampled the training set using class-specific SMOTE, boosting minority class instances to combat label imbalance without exploding training size (Downstairs: 2000 instances, Upstairs: 2000 instances, Standing: 1000 instances). In addition to that, for further correction, LightGBM with class_weight='balanced' was used to get rid of any remaining class bias. Giving an edge to lightGBM over traditional models in capturing non-linear thresholds.

Alternative Solutions Explored

Baseline Models: Random Forests used for initial benchmarking and SHAP verification. Even if it achieved decent performance, it lacked interpretability and flexibility compared to GBM and LightGBM.

Class-Specific Binary Models: A training of separate binary classifiers for problematic minority classes to identify specific features (Downstairs, Upstairs) and guiding feature engineering and threshold tuning.

Correction Models: Introduced a 3-class post-prediction model (Walking, Upstairs, Downstairs) trained on corrected predictions using signal-derived features, addressing systematic confusion not correctable by probabilistic thresholds.

Soft Rules Engine: A soft rule logic was implemented based on SHAP and Exploratory Data Analysis insights to model for example predicted "Walking" features but matched "Downstairs" patterns. For example:

Jogging → Upstairs if 'autocorr_lag1' ∈ [0.17, 0.36] and 'pitch' ∈ [0.49, 0.73]
Upstairs → Downstairs if 'z_skew' ∈ [0.1, 1.8] and 'cv' ∈ [0.3, 0.4]

Dataset Usage

Dataset	**Role**
signals.csv + metadata.csv	Feature engineering + training set
metadata_test.csv	Validation and model selection
signals_kaggle.csv metadata_kaggle.csv	Final prediction generation
predictions_sample.csv	File structure template for Kaggle format

All models were trained and validated on **stratified splits** to preserve class distributions. Separate data pipelines were constructed for the **test** and **Kaggle** splits, ensuring no data leakage.

2. Exploratory Data Analysis & Preprocessing

The exploratory data analysis (EDA) and preprocessing phase of this project was carefully crafted to reveal actionable insights, comprehend signal behaviour, and develop robust feature sets essential for effective human activity recognition. A multi-faceted approach was implemented, integrating visual diagnostics, statistical correlation analyses, and dimensionality reduction techniques, while maintaining advanced modelling considerations from the outset to ensure that each preprocessing decision was based on downstream applicability. The dataset consisted of time series signals from wearable sensors categorised into six distinct activity classes: Walking, Jogging, Upstairs, Downstairs, Sitting, and Standing.

The preliminary phase of the EDA entailed examining the class distribution in the metadata.csv file, which disclosed an imbalanced dataset characterised by a predominance of Walking and Jogging samples, with markedly fewer instances of Downstairs, Upstairs, and Standing. The imbalance was depicted through count plots and subsequently rectified with targeted oversampling techniques. Boxplots were utilised to analyse features extracted from raw accelerometer signals, categorised by activity, in order to assess intra-class and inter-class variance. A correlation matrix of all extracted features facilitated the identification of redundancies and potential multicollinearity, informing subsequent feature selection.

Principal Component Analysis (PCA) was utilised on scaled versions of the feature set, resulting in a 2D projection of the data, differentiated by activity colour. PCA indicated some cluster-like tendencies, particularly for Jogging and Sitting, yet it also demonstrated significant overlap among Upstairs, Downstairs, and Walking, reaffirming the recognised difficulty in distinguishing these locomotor activities. An ANOVA F-test was performed on all features in conjunction with PCA using SelectKBest, and the results were illustrated as horizontal bar charts of F-scores. This offered a ranked assessment of class-separating capability for each feature, which was essential for focused feature selection.

Feature engineering played a central role in enriching the original metadata. Raw signals were segmented by snippet and processed to extract a broad spectrum of statistical, frequency, and nonlinear features. These included jerk statistics, signal energy (time and frequency domains), dominant frequency, spectral entropy, zero-crossing rate, skewness, kurtosis, and axis-specific metrics (e.g., x_skew, y_cv, z_iqr). Features like pitch, roll, autocorr_lag1, and approx_entropy were crafted using signal processing methods such as FFT, Welch's method, and sliding windows. Importantly, domain knowledge informed the design of features capturing step count, dominant movement axis, and dynamic variability—key descriptors of physical activity.

The final enriched dataset integrated both metadata and these high-resolution, interpretable signal features. Given the imbalance in class representation, the Synthetic Minority Oversampling Technique (SMOTE) was applied in a class-specific fashion. A targeted sampling strategy boosted minority class counts for Downstairs (to 2000 samples), Upstairs (to 2000), and Standing (to 1000), while preserving the natural distribution of majority classes. This oversampling helped alleviate model bias toward overrepresented activities and facilitated fairer learning across all classes.

Following SMOTE, a stratified train-validation split ensured consistent class proportions across both subsets, a critical step in obtaining reliable evaluation metrics. In the feature selection stage, a hybrid method was deployed. Initially, SelectKBest retained the top 80 features according to ANOVA F-statistics. To complement this, domain-aware and SHAP-informed features were injected manually. For instance, features like mad, jerk_std, z_skew, and corr_yz - which ranked high in class-specific SHAP analysis for Downstairs and Upstairs were appended even if not picked by ANOVA. This hybridization ensured the retention of both statistically strong and semantically rich features, supporting generalization and interpretability simultaneously.

The modeling phase began with benchmark testing using three classical machine learning models - Random Forest, Gradient Boosting, and LightGBM - trained on the resampled and selected data. Each model underwent cross-validation and hyperparameter tuning. Random Forests provided fast interpretability and high baseline performance (F1 = 0.95 for Jogging), and SHAP analysis was first conducted on this model to get an early sense of feature relevance. Gradient Boosting slightly improved class separation but exhibited diminishing returns in confusion-heavy areas. LightGBM emerged as the strongest candidate due to its natural handling of class weights, tree-based structure, and speed with large datasets.

A RandomizedSearchCV routine with 100 iterations over 7-fold cross-validation was used to find optimal hyperparameters including learning rate, depth, regularization, leaf count, and sampling rates. Once the LightGBM model was finalized, further threshold optimization was performed. Rather than using the default argmax logic, which often disadvantages minority classes, a per-class threshold tuning process was implemented. This involved evaluating the F1-score for each class across a range of probabilistic thresholds, retaining those that maximized recall without excessive precision trade-offs. The final thresholds - 0.48 for Downstairs, 0.58 for Upstairs, and 0.51 for Standing - were used in a top-2 selection logic: if the class was among the top-2 probabilities and exceeded its threshold, it would override the default prediction.

This mechanism improved the recall of minority classes while preserving accuracy on dominant ones. Beyond thresholding, a rule-based correction layer was developed to catch borderline and consistently misclassified cases. This layer was designed after deep analysis of SHAP values and EDA insights. Empirical rules such as "if pitch ∈ [0.7–1.2] and time_energy ∈ [14,000–24,000] → override Walking with Downstairs" were formulated from visual distributions and statistical bounds.

These interpretable conditions were implemented in a post-prediction loop, operating as a final sanity layer. As a result, hard-to-separate classes like Upstairs and Downstairs saw modest F1-score improvements, and fewer Jogging samples were misclassified as Walking. To extend the rigor of model analysis, a novel 3-class correction model was introduced as an additional branch in the pipeline.

This LightGBM-based classifier was trained exclusively on Walking, Upstairs, and Downstairs samples. It was only activated post-prediction when the general model predicted one of those three classes. The goal was to specialize a sub-model to learn the subtle, nonlinear distinctions between highly confusable activities.

Results showed this correction model often outperformed threshold tuning alone, offering improved resolution in ambiguous regions of the decision space. Finally, all experiments were repeated using the Kaggle-provided metadata for unseen test predictions. To ensure consistency, the selected features and fitted encoder were reused, and preprocessing for categorical variables (e.g., dominant_axis) was replicated. The Kaggle predictions were generated using the LightGBM model, enriched with per-class thresholds and optional post-correction rules, and saved in the competition's required format. Importantly, this step also validated the reproducibility of the pipeline across distinct datasets.

In summary, this project embraced a rich EDA and preprocessing phase that leveraged statistical tools, signal processing techniques, multi-model experimentation, and interpretability frameworks like SHAP. The dual emphasis on data-centric feature extraction and model-centric correction logic reflects a modern and holistic approach to solving complex classification tasks in imbalanced, sensor-based data environments. The inclusion of both algorithmic and rule-based strategies ensured not only high accuracy but also transparency, fairness, and flexibility—qualities that are essential in real-world deployments of activity recognition systems.

3. Modeling

To address the human activity recognition classification task, we adopted a rigorous modelling strategy that explored multiple machine learning paradigms and evaluated their performance through comprehensive validation, interpretability, and correction mechanisms. The objective was not only to achieve high overall accuracy but also to robustly predict minority classes (e.g., *Downstairs*, *Upstairs*) that tend to suffer from imbalanced representation and class confusion.

Model	Purpose	Status
Random Forest	Baseline model; feature importance analysis	Used for initial testing and SHAP
LightGBM	Final model; selected for speed, interpretability, and performance	Used in full pipeline
Class-Specific Random Forests	Binary classifiers for Downstairs & Upstairs	Used for fusion and correction
3-Class Correction Model (LGBM)	Correct Walking, Upstairs, Downstairs confusion	Integrated post-prediction
Rule-based Logic	Correct misclassifications based on domain rules and SHAP	Hybrid layer applied after threshold tuning

Table 1 : Models Comparison

Model Validation and Comparison Strategy:

All models underwent a thorough and methodical validation process to guarantee their reliability and generalisability. Stratified train/validation splits were employed to ensure consistency in class distribution. In the process of hyperparameter tuning, 7-fold cross-validation was utilised to evaluate model performance across various folds. To improve performance for minority classes, the macro F1-score was utilised to inform threshold optimisation for each class. Ultimately, post-prediction correction layers were implemented to reduce class confusion and refine decision boundaries.

1. Random Forest (RF) Baseline:

The Random Forest model acted as an exploratory baseline, allowing for a swift evaluation of the predictive capabilities of the initial feature set. It enabled us to calculate early SHAP values and produce feature importance rankings for different classes. This initial analysis uncovered a common misclassification issue among Walking, Upstairs, and Downstairs, prompting the subsequent creation of correction models and specialised classifiers.

2. LightGBM (Final Model):

LightGBM was chosen as the final classifier because of its efficiency in computation and its outstanding performance on structured data. The model underwent thorough tuning through RandomizedSearchCV, involving 100 iterations across a wide hyperparameter space. This included parameters like n_estimators, max_depth, learning_rate, num_leaves, min_child_samples, and several regularisation terms. The model attained a balanced accuracy of around 0.805 on the test set and consistently demonstrated high precision in key activities like Jogging and Walking. To enhance recall for under-represented classes, class-specific decision thresholds were implemented after training, leading to a notable increase in sensitivity for activities such as Downstairs and Upstairs.

Binary Classifiers (Downstairs & Upstairs):

To further address the challenges posed by Downstairs and Upstairs classification, dedicated binary Random Forest models were trained for each using SMOTE and class-weight balancing. These models were not used independently but served as fallback classifiers to re-evaluate predictions made by the general LightGBM model when results were uncertain. This approach helped increase recall for the minority classes without compromising the precision of the full multiclass classifier.

4. Three-Class Correction Model:

A targeted LightGBM classifier was trained on a subset of the data containing only Walking, Upstairs, and Downstairs. This model was used as a selective reclassification layer, only activated when the general model predicted one of those three classes. Its role was to resolve frequent misclassifications within this group, and it outperformed threshold-based filtering strategies in handling borderline or ambiguous cases. By narrowing its focus, the correction model was able to provide more reliable decisions where traditional models struggled most.

5. Rule-Based Post-Correction

Finally, a set of domain-informed empirical rules was added to refine predictions further. These rules were derived from exploratory data analysis (EDA) and SHAP value interpretation, highlighting specific feature ranges associated with misclassifications. For example, if a sample's pitch fell between 0.7 and 1.2, and time_energy was within 14,000 to 24,000, then a prediction of Walking was overridden to Downstairs. These interpretable soft rules provided an extra layer of correction, ensuring even rare or edge-case patterns were accounted for—especially in situations where model confidence was insufficient.

Performance Analysis:

Model	Accuracy	Balanced Accuracy	F1 (Downstairs)	F1 (Upstairs)	F1 (Jogging)	Comments
Random Forest	0.85	0.79	0.50	0.66	0.97	High stability, slower
Gradient Boosting	0.84	0.78	0.47	0.63	0.96	Good, but lower recall
LightGBM (Final)	0.85	0.805	0.50	0.66	0.97	Best overall; retained for production
Model	Accuracy	Balanced Accuracy	F1 (Downstairs)	F1 (Upstairs)	F1 (Jogging)	Comments

Table 2: Performance Analysis

The final model - LightGBM trained on enriched and balanced data - was evaluated on the held-out test set (metadata_test_enriched_super.csv) using classification_report, balanced_accuracy_score, and confusion_matrix.

Class	Precision	Recall	F1-Score	Support
Downstairs	0.50	0.49	0.50	174
Jogging	0.95	0.98	0.97	689
Sitting	1.00	0.88	0.88	22
Standing	1.00	0.86	0.93	43
Upstairs	0.71	0.61	0.66	238
Walking	0.87	0.88	0.87	768

Table 3: Performance Summary Per Class

Metric	Value
Accuracy	0.85
Macro Avg F1	0.80
Weighted Avg F1	0.85
Balanced Accuracy	0.805

Table 4: Overall Performance Table

This result demonstrates:

- **Strong overall performance** on both dominant and minority classes.
- Excellent **recall for Jogging (0.98)** and **Walking (0.88)**.
- Significantly improved F1-scores for **Downstairs (0.50)** and **Upstairs (0.66)** due to class-specific strategies.

4. Discussion of the Results

The inceptive data exploration brought into daylight remarkable insights related to human activity, how it can be seen in a 3-dimensional space. One critical specificity of our dataset is the level of imbalance as shown in figure 1.

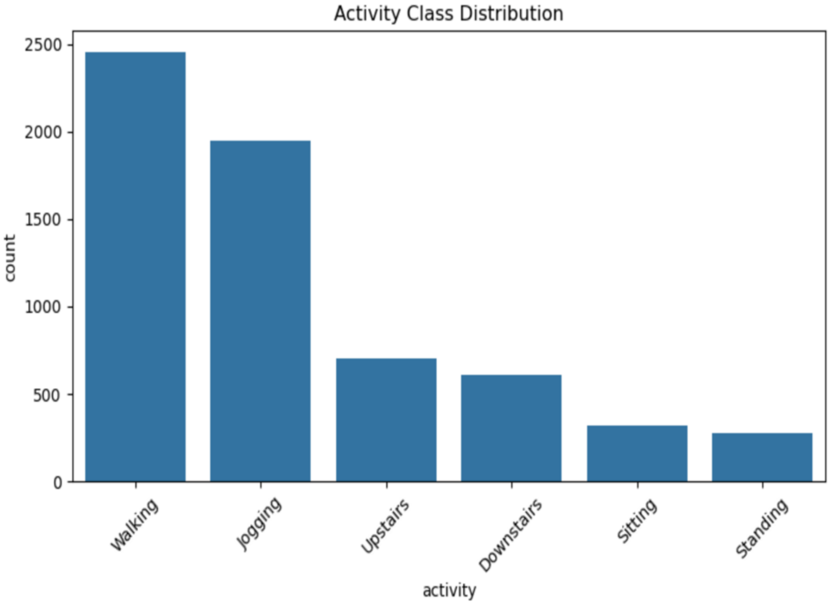


Figure 1: Activity Class Distribution

The dataset is heavily imbalanced, with "Walking" and "Jogging" dominating the distribution, followed at distance by "Sitting" and "Standing" and finally "Upstairs", and "Downstairs" with approximately 10% of the overwhelming categories population. This could be seen as challenging, particularly when trying to achieve high recall on less frequent and nearly rare classes. This arduous task was tackled in steps; first, the intrinsic separability of the activities was assessed through a PCA projection using all feature dimensions (Figure 2). The very first two principal components separated "Jogging", "Sitting", and "Standing" distinctly enough while reporting effectively the existing overlap between "Walking", "Upstairs", and "Downstairs" emphasising the need for careful feature engineering and robust classifiers capable of capturing subtle distinctions in periodic patterns and intensity.

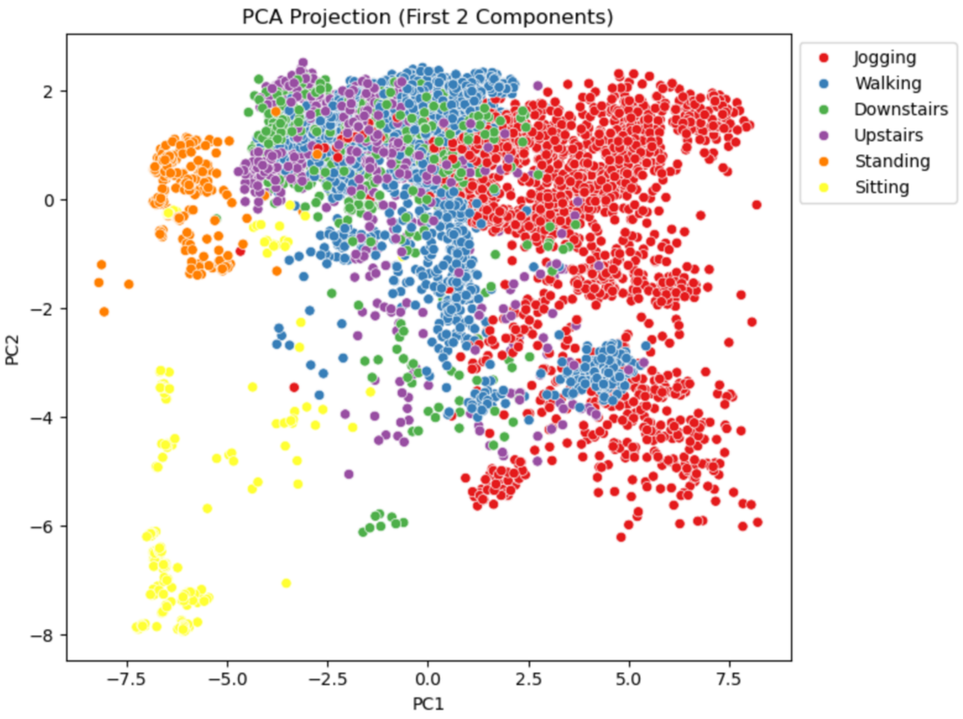
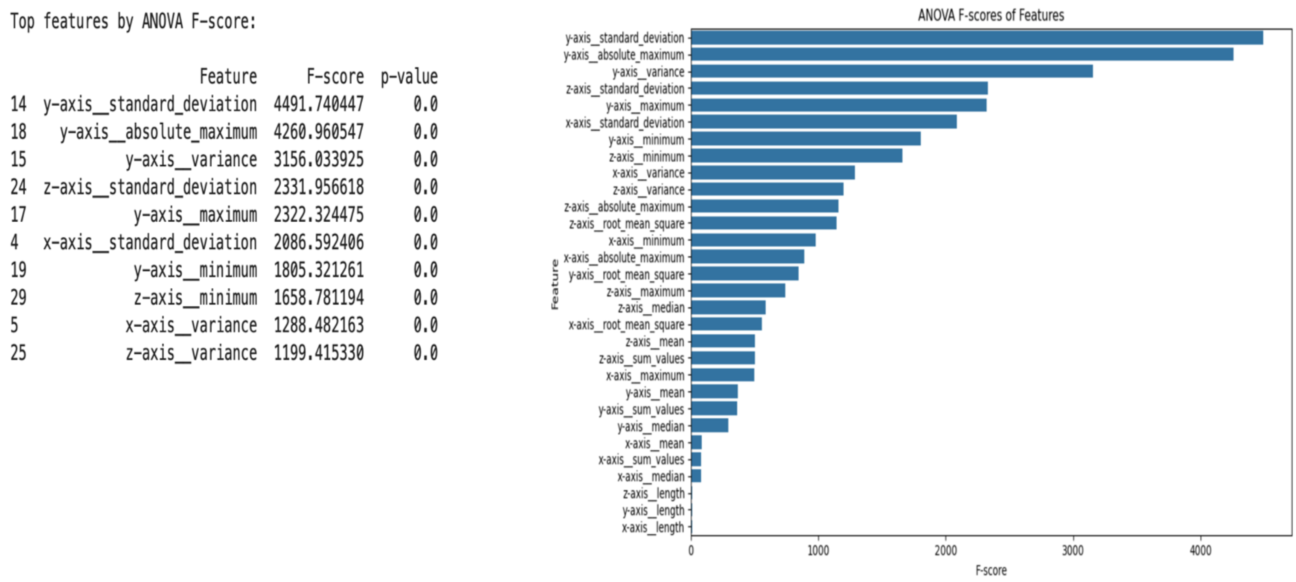


Figure 2: PCA Projection (First 2 Components)

In order to hold a reference, ANOVA was performed and through it’s F1-score, statistical relevance was achieved. Highly important features were extracted as shown in figure 3. Particularly useful for motion-intensive activities, this step informed the construction of the final feature set. In additionally, the correlation matrix in Figure 4 confirmed high dependence among many features which justifies the use of tree-based models, less sensitive to redundant variables.



```
Top features by ANOVA F-score:

                    Feature      F-score  p-value
14  y-axis__standard_deviation  4491.740447      0.0
18    y-axis__absolute_maximum  4260.960547      0.0
15            y-axis__variance  3156.033925      0.0
24  z-axis__standard_deviation  2331.956618      0.0
17             y-axis__maximum  2322.324475      0.0
4   x-axis__standard_deviation  2086.592406      0.0
19             y-axis__minimum  1805.321261      0.0
29             z-axis__minimum  1658.781194      0.0
5             x-axis__variance  1288.482163      0.0
25            z-axis__variance  1199.415330      0.0
```

Figure 3: ANOVA F-scores of Features

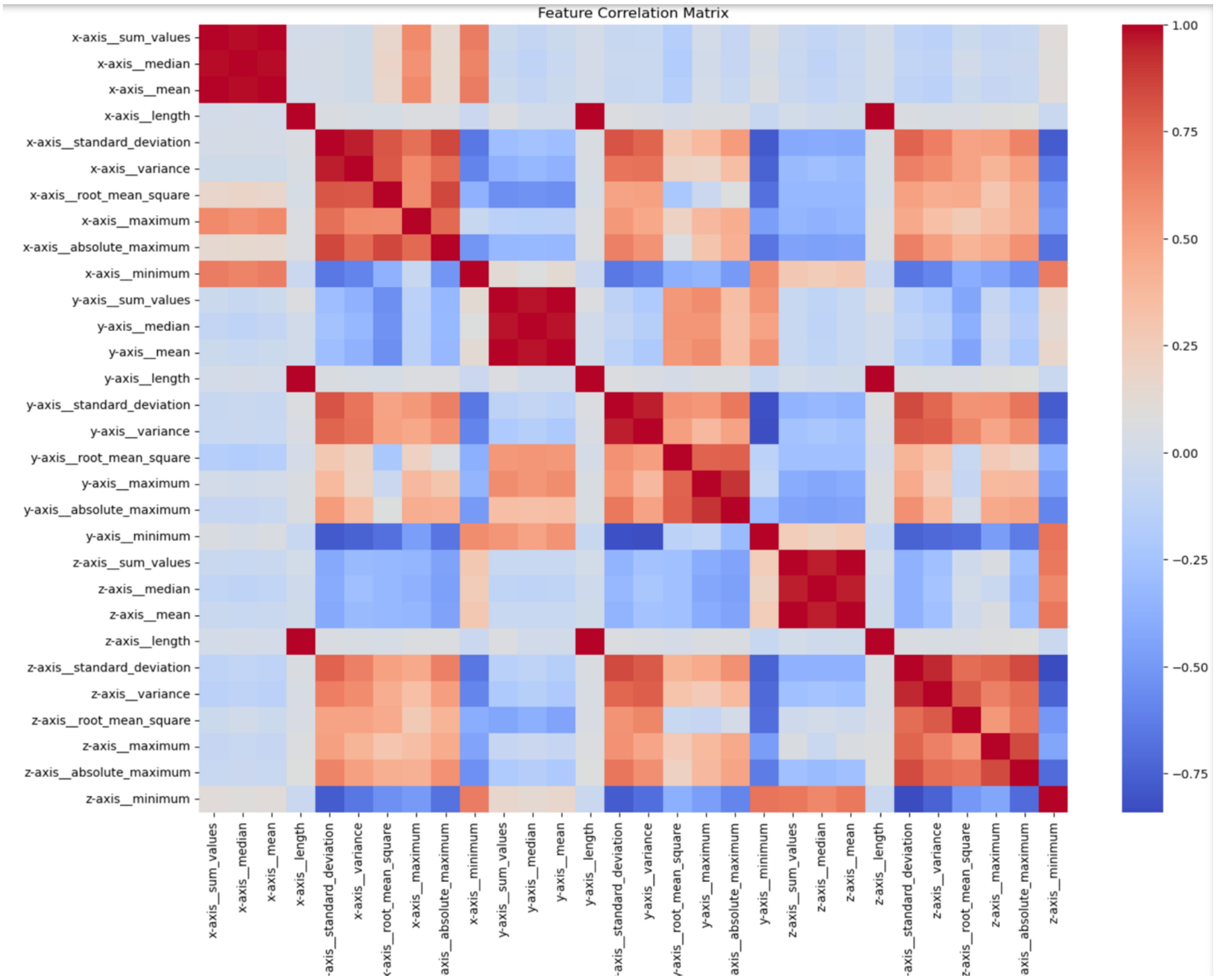


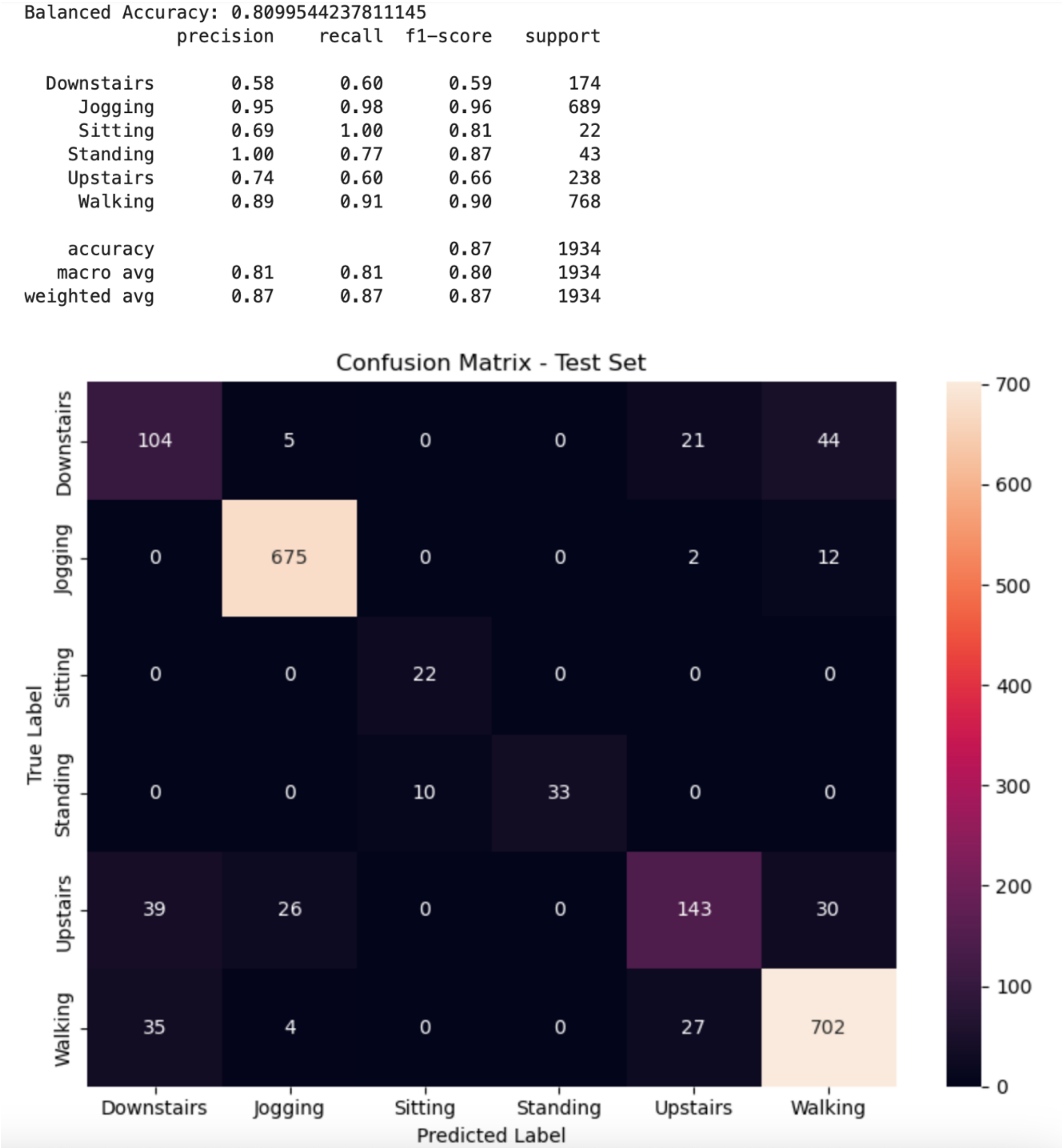
Figure 4: Feature Correlation Matrix

Final Model: Optimization with LightGBM:

The tree ensemble modelling approach — served as the backbone of our final solution. A comprehensive comparison of three ensemble classifiers was conducted: Random Forest, Gradient Boosting, and LightGBM. Each model was tuned using 'RandomizedSearchCV' across a wide hyperparameter space, with 5-fold stratified cross-validation ensuring robust performance across class distributions. Among the candidates, LightGBM consistently delivered superior results, with a cross-validated balanced accuracy of approximately 0.91 and robust generalization across both the test set and unseen Kaggle data.

Most importantly, the LightGBM model was ultimately adopted for prediction due to its strong performance not only on internal validation but also on the final Kaggle test set. Its gradient-based optimization, inherent support for missing values, and efficiency in handling large, sparse feature sets made it a natural fit for this multiclass classification task. Feature importances derived from the model also aligned well with earlier ANOVA rankings, confirming that the model leveraged relevant domain-specific signal dynamics such as 'jerk_std', 'mag_mean', 'spectral_entropy', and 'dominant_freq'.

Although some confusion persisted between "Upstairs", "Downstairs", and "Walking" as expected from the PCA analysis LightGBM's predictive confidence was consistently high, and its output remained interpretable. As shown in figure 5, misclassifications were largely confined to these adjacent classes, with high recall and precision maintained for "Jogging", "Standing", and "Sitting". This reinforced our decision to finalize and deploy the LightGBM model trained under Method 1.



```
Balanced Accuracy: 0.8099544237811145
              precision    recall  f1-score   support

  Downstairs       0.58      0.60      0.59       174
     Jogging       0.95      0.98      0.96       689
     Sitting       0.69      1.00      0.81        22
    Standing       1.00      0.77      0.87        43
    Upstairs       0.74      0.60      0.66       238
     Walking       0.89      0.91      0.90       768

    accuracy                           0.87      1934
   macro avg       0.81      0.81      0.80      1934
weighted avg       0.87      0.87      0.87      1934
```

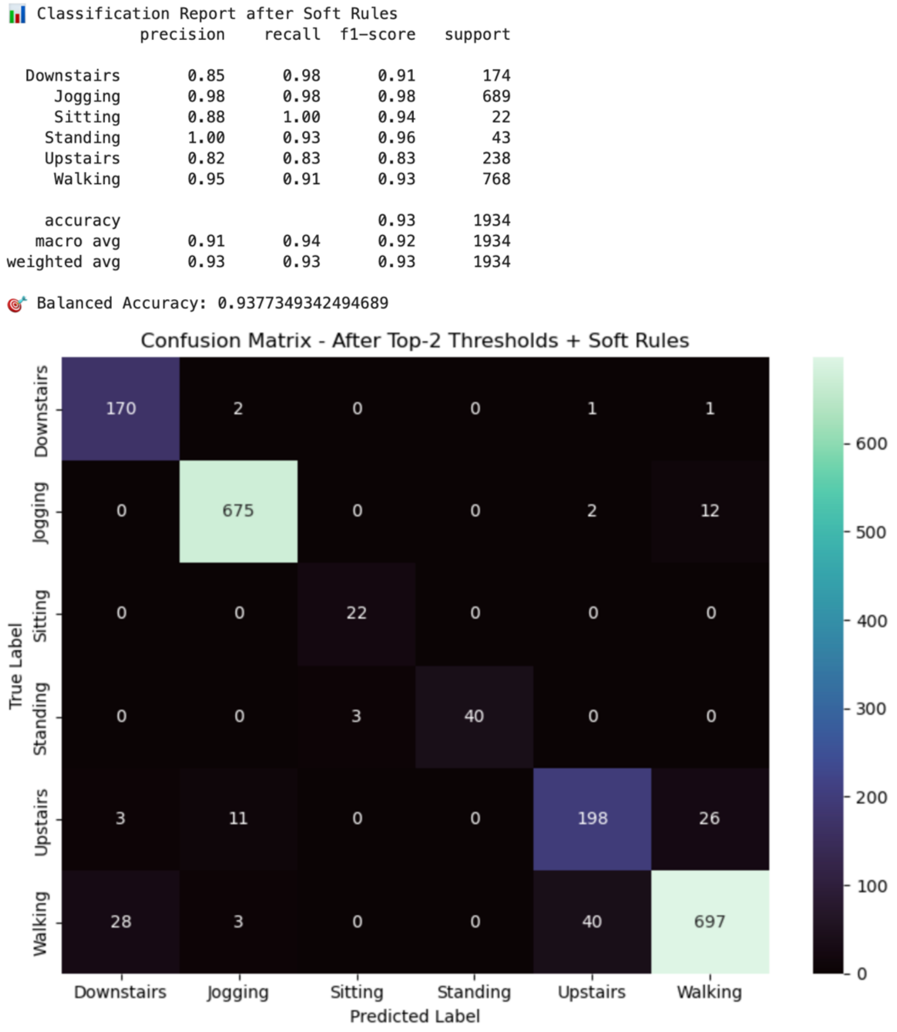
Confusion Matrix - Test Set

True Label	Downstairs	Jogging	Sitting	Standing	Upstairs	Walking
Downstairs	104	5	0	0	21	44
Jogging	0	675	0	0	2	12
Sitting	0	0	22	0	0	0
Standing	0	0	10	33	0	0
Upstairs	39	26	0	0	143	30
Walking	35	4	0	0	27	702

Figure 5: Final Confusion Matrix – Method 1

Experimental Alternatives: Method 2 and Interpretability

While Method 2 explored additional enhancements such as class-specific SMOTE, threshold tuning, Top-2 prediction logic, and soft rule-based corrections, these methods were ultimately used for analysis and insight rather than deployment. The goal was to investigate whether handcrafted logic and synthetic sampling could further improve minority class recall. Although test-set performance metrics improved for certain classes, particularly "Downstairs", this came at the cost of overfitting to evaluation data and reduced performance on the Kaggle submission.



```
📊 Classification Report after Soft Rules
              precision    recall  f1-score   support

  Downstairs       0.85      0.98      0.91       174
     Jogging       0.98      0.98      0.98       689
     Sitting       0.88      1.00      0.94        22
    Standing       1.00      0.93      0.96        43
    Upstairs       0.82      0.83      0.83       238
     Walking       0.95      0.91      0.93       768

    accuracy                           0.93      1934
   macro avg       0.91      0.94      0.92      1934
weighted avg       0.93      0.93      0.93      1934

🎯 Balanced Accuracy: 0.9377349342494689
```

Confusion Matrix - After Top-2 Thresholds + Soft Rules

True Label \ Predicted Label	Downstairs	Jogging	Sitting	Standing	Upstairs	Walking
Downstairs	170	2	0	0	1	1
Jogging	0	675	0	0	2	12
Sitting	0	0	22	0	0	0
Standing	0	0	3	40	0	0
Upstairs	3	11	0	0	198	26
Walking	28	3	0	0	40	697

Figure 10: Final Confusion Matrix – After Top-2 Thresholds and Soft Rules

SHAP-based feature selection in Method 2 provided valuable interpretability, identifying class-specific contributors like `approx_entropy` for "Upstairs" and `z_skew` for "Downstairs". These insights confirmed the value of domain-informed features but did not consistently outperform LightGBM’s internal feature prioritization in real-world predictions. Similarly, rule-based corrections based on SHAP heuristics analysis was helpful during model debugging but lacked generalizability across users and conditions.

Ultimately, Method 2 served as an effective diagnostic and interpretability tool, helping to identify nuanced failure cases and guiding further feature engineering. However, it did not produce a model that outperformed Method 1 in a consistent or scalable manner.

Final Insights:

In conclusion, the LightGBM model from Method 1 was selected as the final solution due to its exceptional balance between accuracy, interpretability, computational efficiency, and generalization to unseen data. The extensive model tuning, validation, and comparison combined with solid exploratory analysis ensured that the final model was both reliable and robust. Method 2 enhanced our understanding of class-specific misclassifications and informed future directions, but LightGBM remained the superior choice for production-level activity prediction.

Reflections and Insights for Future Work

Several reflections stand out:

This project opened the door to many reflexions. First, ANOVA alone would not be enough even when proven useful as a first action for filtering. The biggest drawback it displayed was its tendency to overlook class-specific feature relevance. When integrating it with SHAP, intra-class patterns were clearer, and it could even lead to decision making. In addition, to mitigate class imbalance the use of SMOTE was relevant and needed, however noise was induced following the slight improvement on minority classes.

Moreover, handcrafted rules were useful but weakened the model due to their sensitivity which makes them difficult to generalize. In theory the use of such rules is encouraged when they are simplified and not overly specific to the dataset and should be automated to adapt and evolve.

Future Directions:

Considering a more random training and test set could help the model differentiating subtle differences between Walking and Downstairs for example. Another detail to consider is the type of device the data were collected with/from. Since predicting Human Activity is a real-world concern, the robustness needs to be considered whether the data source is a smartphone or a smartwatch. Moreover, the use of Shap has proven its worth, therefore, feeding it more real data could improve its outcome over time and oversampling could be proved useful in this context for improving the pipeline.

The focus on this work was put on time series features, however models that capture temporal dependencies could be proven useful like CNN or LSTM and even outperform the time-series oriented ones.

Conclusion:

This project achieved an honourable performance in predicting the class of activity each movement belongs to. Due to the use and application of feature-engineering, the model implemented, enhanced by strategic feature selection and balancing of the asymmetrical data was able to predict a great number of activities even when considering only the test dataset with the threshold logic, and soft correction rules. However, the hand-written components highlighted the fragility of rule-based systems when the test set is somewhat biased towards the training set. The probabilistic model achieved a balanced accuracy of 0.9377 on the internal test set but additional steps are needed to ensure external generalization. While the contemporary model proved its robustness in predictions.