Application of Machine Learning Methods to Classify Physical Activity

Xia Wu X385WU@UWATERLOO.CA
Department of Management Sciences
University of Waterloo
ID: 20981841

Professor: Dr. Lukasz Golab

1 Introduction

Human activity recognition has demonstrated its utility in various domains, including healthcare, sports, smart homes, and wearable technology. For example, automatic recognition and monitoring of patients' daily activities can play a significant role in managing chronic diseases like obesity, diabetes, and cardiovascular conditions [1]. Implementing an activity recognition system can aid patients in managing their lifestyle effectively and empower physicians to closely monitor their progress, thus providing tailored recommendations.

This project deals with physical activity classification using data collected by three Inertial Measurement Units (IMUs) and one heart rate monitor. IMUs are electronic devices that combine multiple sensors to measure and report the orientation, velocity, and gravitational forces acting on an object. The primary objective of this project is to develop multiclass classification models capable of accurately classifying diverse activities, including walking, running, cycling, jumping, and more. By processing and analyzing the sensor data, I aim to extract meaningful features and patterns that represent activities effectively.

The project entails various stages, including data preprocessing, exploratory analysis, feature engineering, model selection, hyperparameter tuning, model evaluation and comparison. To ensure the system's reliability and adaptability, I will explore a range of machine learning algorithms such as Logistic Regression, Decision Trees, Random Forest, K Nearest Neighbors (KNN) and Support Vector Machine (SVM). Additionally, Principal Component Analysis (PCA) and Recursive Feature Elimination (RFE) method will be employed to extract relevant information from the sensor data and improve the model's predictive performance.

2 Literature Review

Sensor-based human action recognition has been a well-explored topic for several years. Some researchers have already investigated the combination of various motion sensors in activity recognition. Wanmin et al. (2012) performed the K Nearest Neighbor classifier on the accelerometer and gyroscope data collected by iPod Touch and reported the recognition accuracy 100% for sitting and 52.3%–79.4% for up and down stair walking [2]. Furthermore, Muhammad et al. (2014) conducted a study to assess the impact of different sensors on activity recognition performance. Their findings confirmed that both the accelerometer and the gyroscope can be crucial in the activity recognition process, with their significance varying based on factors like the type of activity being recognized and the body position, among others [3].

Various machine learning algorithms have demonstrated impressive performance in classifying human activities. For instance, Daghistani et al. (2016) utilized an ensemble method, combining AdaBoost with other classifiers such as Decision Tree, Logistic Regression, and Multi-Layer Perceptron. Their results revealed that the combination of AdaBoost with Decision Tree achieved the highest accuracy of 94.03% [4]. Similarly, Chawla et al. (2016) conducted a comparative study of four classifiers, namely K-Nearest Neighbor, Support Vector Machine, Artificial Neural Network, and Decision Tree. Their findings indicated that the Artificial Neural Network yielded the highest accuracy of 96.77% [5].

In light of these studies that demonstrate the effectiveness of various machine learning techniques in accurately recognizing and classifying physical activities, my focus is on extracting meaningful features to enhance classification efficiency.

3 Dataset

The PAMAP2 dataset of physical activity monitoring is used in this project was provided courtesy of Attila Reiss and accessed via the UC Irvine Machine Learning Repository [6]. Data collection and primary data analysis was conducted by Attila Reiss of the Technical University of Kaiserslautern.

The dataset was recorded with 9 subjects, wearing 3 IMUs and a heart rate monitor, and performing 12 different activities.

Each IMU contains a 3-axis MEMS (micro-electro-mechanical system) accelerometer, a 3-axisMEMS gyroscope, and a 3-axismagneto-inductive magnetic sensor, all sampled at 100Hz. The HR-monitor provided heart rate values with approximately 9Hz. The sensors were placed onto 3 different body positions. A chest sensor fixation includes one IMU and the heart rate chest strap. The second IMU is attached over the wrist on the dominant arm, and the third IMU on the dominant side's ankle3, both are fixed with sensor straps.

The raw PAMAP2 dataset contains a total 2,872,533 records and 55 features with 54 features are numerical and 1 feature is categorical.

3.1 Data Pre-processing

3.1.1 Create New Features

To account for the potential impact of orientation changes on sensor recognition performance [11], a fourth dimension called 'magnitude' is introduced to each sensor's existing dimensions [3]. The magnitude is computed using the following formula:

$$magnitude = \sqrt{x^2 + y^2 + z^2}$$

3.1.2 Imbalanced Sample Size

Due to the imbalanced sample sizes within the labeled activities in the dataset, I focused on data generated by subjects 101, 102, 105, and 108, as they performed all activities. Additionally, I included data from subject 109, who only performed one activity, to supplement the sample size for the 'rope jumping' activity. This approach was taken to ensure a more representative and balanced dataset for model training and evaluation.

3.1.3 Data Cleaning

Irrelevant features. Irrelevant features, such as those related to orientation and acceleration (±6g), were excluded from the dataset due to imprecise calibration and saturation issues, respectively.

Missing value. For the 'heart rate' feature, missing values were filled using the last non-null value because the sensor data is provided every 0.01s (100Hz), while the HR-monitor's sampling frequency was approximately 9Hz. Any missing values in other features were directly removed.

Wrong data. Data related to transient activities, specifically data labeled 0 of 'activityID', was eliminated. Mislabeling can occur during transitions between activities, so data from the initial and final 10 seconds of each labeled activity were deleted to ensure data consistency and accuracy.

Outliers. Outliers for each subject were detected using boxplots and subsequently removed from the dataset. However, some data points appeared to be outliers after the cleaning process. I chose to retain these data points in the dataset because factors like heart rate range can vary significantly among different individuals [7] and removing them could lead to potential loss of important information.

After cleaning, the data set contains 932,335 records with 36 features, and the target feature (activityID) contains 12 labels.

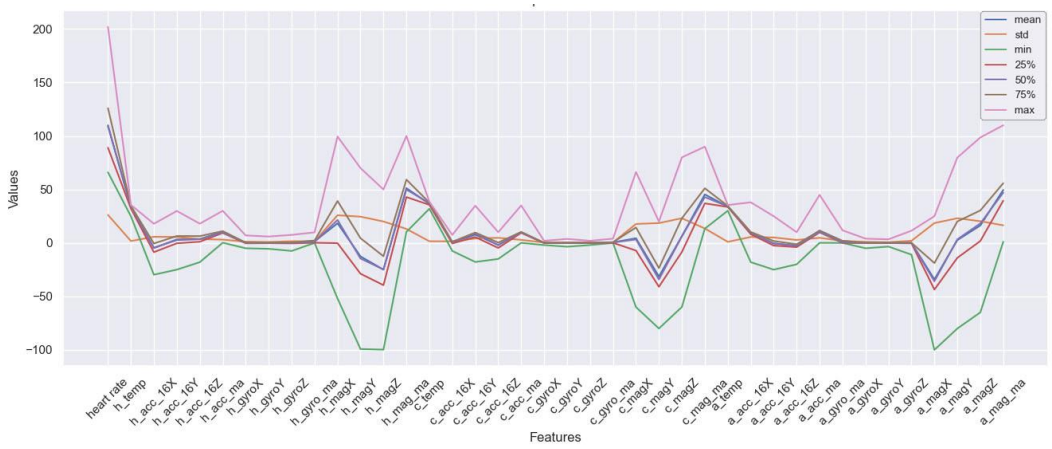
3.2 Data Exploration

3.2.1 Descriptive statistics

The dataset comprises a total of 932,335 records with 36 features, and the target feature (activityID) contains 12 labels. Among the features, heart rate exhibits the highest mean value, whereas features related to the gyroscope have the lowest mean and standard deviation. Additionally, the features

associated with the magnetic sensor demonstrate a smaller mean compared to heart rate but a larger standard deviation, as depicted in Figure 1.

Figure 1. Overview of Descriptive Statistics



3.2.2 Data Visualization

The following plots display the distributions of various features across different activities. The two box plots depicted in Figure 2 indicate that as the intensity of activities increases (from lying to rope jumping), the heart rate values (left plot) tend to increase, while the hand temperature values (right plot) tend to decrease.

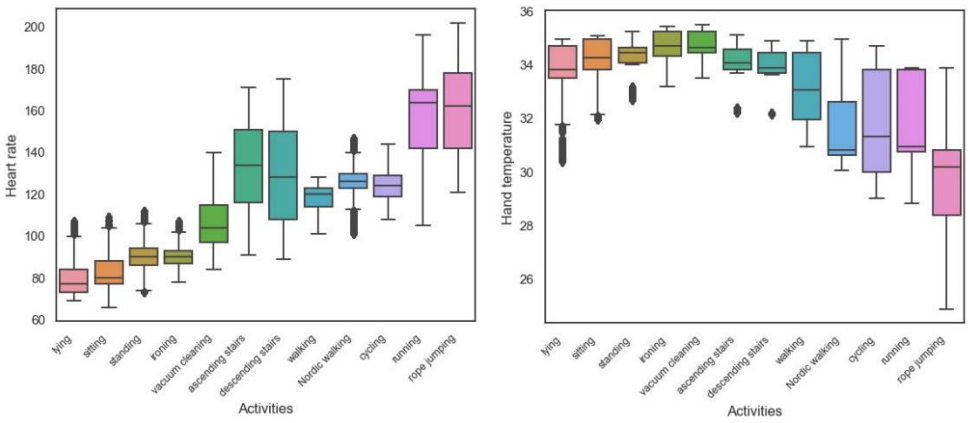


Figure 2. The Boxplot of Heart Rate (left) and Hand Temperature (right) by Activity

The data distribution of acceleration and magnetometer features varies across the three dimensions, while the distribution of gyroscope data remains similar within different dimensions (Figure 3). Understanding these differences can aid in feature selection and resampling the dataset to achieve better balance in sample sizes.

The correlation map reveals that most features have low correlation with each other, except for chest temperature and hand temperature, which exhibit a relatively high correlation of 0.83.

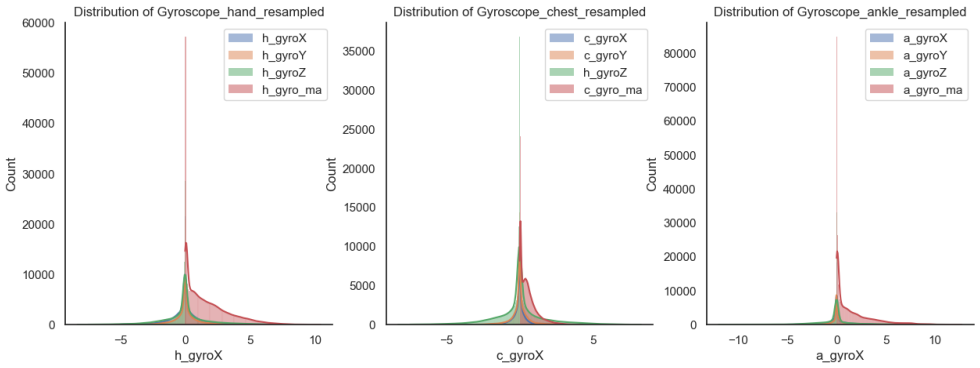


Figure 3. The Distribution of Gyroscope

In Figure 4, the imbalanced sample sizes among the activities are depicted, highlighting a substantial gap of 75,735 samples between rope jumping and ironing. This imbalance can negatively impact model performance and lead to misrepresentation of rare activities [8]. To mitigate the risks associated with imbalanced samples, the controlled under-sampling technique is employed to address this issue. After under-sampling, total 362,832 records with 36 features were keeping in data set. Each activity contains 30,236 data points.

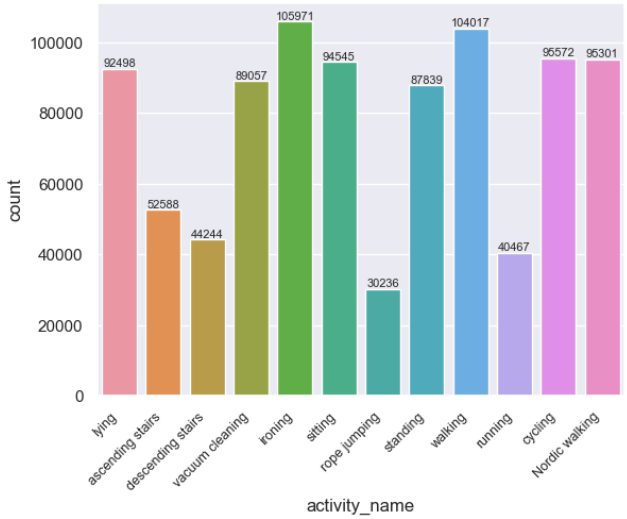


Figure 4. The Number of Samples for Each Activity before Under-sampling

Figure 5 presents a comparison between the distribution plots of the original raw data and the resampled data. The purpose is to verify that the balancing process maintains data integrity without causing substantial information loss.

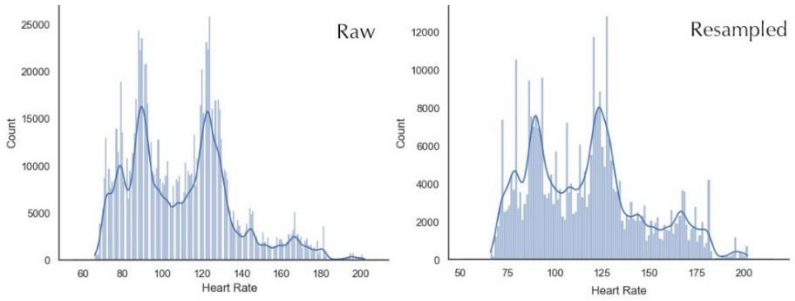


Figure 5. The Comparison of Distribution of Heart Rate between Raw and Resampled Data

4 Methodology

The activity classification task is a multi-class classification problem, which aims to predict one of 12 activities a subject is performing based on 33 features, including heart rate, temperature, and sensor records.

Due to limited computational resources, a subset of 5,000 records was randomly selected from the original dataset for the subsequent modeling process.

As feature sets may exhibit varying performances in different models, I conducted modeling with feature selection. Subsequently, I utilized the filtered feature sets from each model in other models and compared their respective performances. This approach allowed me to assess the impact of feature selection on different models and identify the most effective feature sets for improved overall performance.

Given that this is a categorical classification task all models were optimized with respect to accuracy as the evaluation metric.

$$accuracy = \frac{True\ positive + True\ negative}{Total\ Number\ of\ Predictions}$$

In addition to assessing model performance through other evaluation metrics, the learning curve was employed as a valuable tool to gain insights into the models' performance. By plotting the learning curve, we could observe how the model's performance evolves with varying amounts of training data. This allowed us to identify potential issues such as overfitting or underfitting and make informed decisions about the model's overall effectiveness and generalization capabilities.

4.1 Data Transformation

4.1.1 Scaling

The features in the dataset have different scales. If the features have vastly different magnitudes, some algorithms may give more importance to the features with larger values, regardless of their actual predictive power [9]. Therefore, the standardization approach was used to scale the features. This approach can ensure that features have similar scales while retaining the distribution information of the original data. The formula for standardization:

$$z = \frac{x - \mu}{\sigma}$$

where x is the original value, μ is the mean of the data, σ is the standard deviation of data.

4.1.2 Shuffling

Since the data collection is chronological, I shuffled the dataset to avoid the model learning the sequential correlation. Shuffling data also improves the generalization ability of the model.

4.2 Data Splitting

The dataset was randomly divided into a training set and a test set for the modeling process, comprising 80% and 20% of the selected samples, respectively. The main objective of the model is to predict the unknown activity labels in the test data accurately. To achieve the highest possible overall accuracy on the test part, standard k-fold cross-validation (CV) is used on the training set to select the most promising methods. This approach allowed us to select the most promising methods for further evaluation on the test data.

4.3 Feature – Extracted Models

4.3.1 Principal Component Analysis (PCA)

To determine the significance of these features and see if there was any possibility for dimensionality reduction, I processed to perform Principal Component Analysis (PCA) on dataset. The PCA results show that 23 features are needed to explain 90% of variance. Due to the sensitivity of the PCA algorithm to outliers and its assumption of linear relationships in the data, I decided to retain all 33 features for analysis. However, I set a practical upper limit of 23 features for each model to prevent potential issues related to high-dimensionality and improve model interpretability.

4.3.2 Multinomial Logistic Regression

The multinomial logistic regression also known as softmax regression which fits model by minimize the cross-entropy loss function:

$$L = -\frac{1}{N}\sum_{k=0}^{N} y_k \log(\hat{y}_k)$$

where $\hat{y}_k$ is the estimated probability for label k.

I employed both forward and backward stepwise methods for feature selection. I then evaluated their performance by comparing their cross-validated accuracy and log-loss using a 5-fold cross-validation. Additionally, I assessed the suitability of the selected models using the scores of Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC).

I additionally used Recursive Feature Elimination (RFE) method to identify the optimal feature set by compare 5-fold cross-validated accuracy.

After selecting the best feature set using the above methods, we proceeded with parameter tuning and model evaluation, resulting in the final optimized parameters: maximum number of iterations is 1000, solver is saga, penalty is L1 regularization with strength C= 0.6158.

4.3.3 Trees - Based Classifier

The Trees-Based Classifier utilizes decision tree structures for classification tasks. These classifiers allow us to compute feature importance, which is valuable for feature selection. Consequently, I initially trained the full model using a tree-based algorithm and subsequently eliminated features with zero importance. Additionally, I conducted Recursive Feature Elimination (RFE) for feature selection on the tree-based model (cv=5). Entropy is used to determine how well a particular feature can split the data into classes or categories.

$$Entropy = -\sum_{i=1}^{c} p_i \log_2(p_i)$$

where c is the number of classes in the target variable, and p_i is the proportion of samples belonging to class i at a specific node.

Decision Trees. Decision Trees algorithm recursively splits the data based on feature values to create a tree-like structure. Each leaf node corresponds to a class label, and the path from the root to a leaf represents a decision path based on the features. The decision trees parameters are minimum samples leaf = 5, minimum samples split=20, criterion=entropy, maximum depth =11.

Random Forest. Random Forest algorithm combines multiple decision-trees, resulting in a forest of trees. It uses bagging or bootstrap aggregating and feature randomness when building each decision tree. The random forest parameters are criterion=entropy, number of estimators = 50, maximum depth =16, minimum samples leaf = 5, minimum samples split =15.

4.3.4 K Nearest Neighbor (KNN)

The K Nearest Neighbor (KNN) classifier is known as a ‘lazy’ learning approach. It is a non-parametric algorithm that estimates the hypothesis function by the mean value of the ground truth

labels over the k nearest neighbor. The similarity between data points is measured by Manhattan distance:

$$Manhattan\ Distance = \sum_{i=0}^{n} \left| x_{1i} - x_{2i} \right|$$

where $(x_1, x_{1i}), (x_2, x_{2i})$ are the coordinates of the first and second point in the n-dimensional space, respectively.

I trained the K Nearest Neighbor (KNN) model by sequentially adding features based on their descending rank of feature importance from the Random Forest model. Subsequently, I performed 3-fold cross-validation to identify the feature set that achieves the highest accuracy. The parameters of KNN model are the number of neighbors is 5, metric is Manhattan, and the weights is distance.

4.3.5 Support Vector Machine (SVM)

The Support Vector Machine (SVM) aims to find a hyperplane or set of hyperplanes in an N-dimensional space (N – the number of features) that distinctly classifies the data points. RFE was employed to determine the best feature set. Following hyperparameter tuning and model evaluation, the final parameters for SVM are as follows: kernel=poly with degree=2, gamma=0.1, and regularization C=1.

4.4 Model Comparison

Using the feature sets selected in the previous section as the base, I introduced three additional features to assess the influence of heart rate and temperature on the model. Consequently, a total of 8 feature sets were considered in the comparison of 9 different models. The feature sets are shown in Table 1. Finally, I selected the feature set with the highest accuracy and conducted hyperparameter tuning on three algorithms: Random Forest, Gradient Boosting, and Multilayer Perceptron, to identify the best-performing model. The parameters are defined as:

1) **Random Forest:** criterion = entropy, maximum depth=10, minimum samples leaf =5, minimum samples split = 8, number of estimators= 50.
2) **Gradient Boosting:** maximum depth= 5, number of estimators=50, minimum samples leaf=8, minimum samples split=20, learning rate= 0.05.
3) **Multilayer Perceptron:** activation= relu, maximum interactions =500, alpha= 0.1, hidden layer sizes= (80, 50, 25), solver=sgd, learning rate =constant, initial learning rate =0.01.

Feature Set	Number of Features	Notation
Heart rate	1	heart
Temperatures	2	temp
Heart rate & hand temperature	2	heart_h
Heart rate & ankle temperature	2	heart_a
Heart rate & Temperatures	3	fea_knn
Selected by Random Forest	5	fea_rf
Selected by SVM	10	fea_svm
Selected by Logistic Regression	15	fea_lgr

Table 1. Feature Sets Selected by Models

5 Results

In the initial phase of the last section, I embarked on finding the most suitable feature set for each algorithm, with a focus on evaluating the models based on their test accuracy, as presented in Table

2. The results clearly indicate that the K Nearest Neighbor classifier outperformed the other four methods, achieving the highest test accuracy of 97.2% using only 3 features. On the other hand, Logistic Regression exhibited the weakest performance, with a test accuracy of 83.4% using all of features. After conducting further comparisons of the complexity and learning curves of Logistic Regression models using three different feature sets, I have ultimately determined that the feature set comprising 15 features is the most optimal choice.

Models	**Train Accuracy %**	**Test Accuracy %**
Logistic Regression	83.48	80.70
Decision Trees	95.45	93.60
Random Forest	98.60	96.90
K Nearest Neighbor	98.75	97.20
Support Vector Machine	86.4	81.50

Table 2. Results of Feature Selection

During the comparison step, all nine algorithms were trained using eight different feature sets. Figure 6 illustrates the test accuracy of each algorithm. It is evident that the feature set containing only 'heart rate' exhibits the lowest test accuracy across all models, and the accuracy increases as one more temperature-related feature (hand or ankle temperature) is added.

The feature sets, fea_lgr, fea_rf, and fea_svm, exhibit similar performance across all algorithms except for Logistic Regression and Gaussian Naïve Bayes. After careful consideration, I ultimately selected the fea_knn feature set to train the final model. This feature set contains only 3 features, significantly reducing the training time of classifiers and requiring fewer computations during online classification.

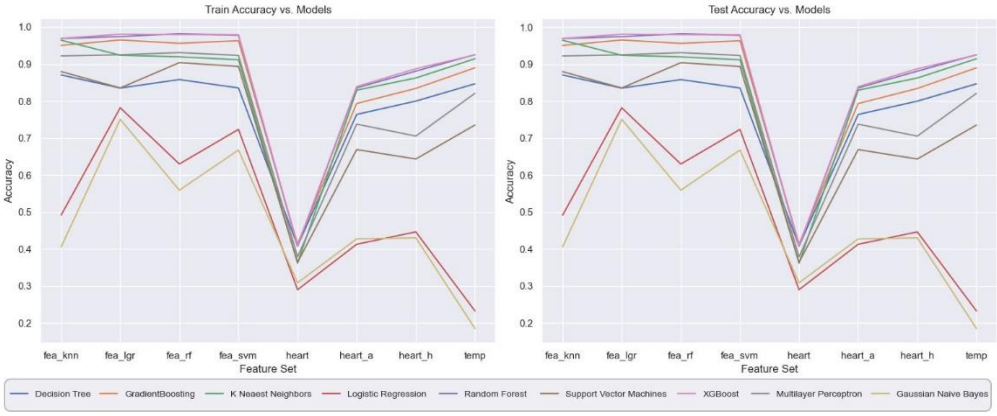


Figure 6. Train Accuracy (left) and Test Accuracy (right) of Models by Feature Sets

The algorithms include Logistic Regression, Gaussian Naïve Bayes, Support Vector Machine and Decision Trees consistently demonstrate lower accuracy compared to the other algorithms across all feature sets. Thus I excluded these four algorithms from further processing.

The best performance was achieved with XGBoost, Random Forest and K Nearest Neighbor using fea_knn feature set. While XGBoost and K Nearest Neighbor demonstrated commendable performance, Random Forest emerges as a standout option among the three, primarily due to its higher interpretability. Conversely, the KNN model and XGBoost model raise concerns about potential overfitting, as evidenced by its learning curve. Therefore, I selected Random Forest as one of the final classifiers.

In the final step, I utilized the fea_knn feature set to train three distinct models: Random Forest, Gradient Boosting, and Multilayer Perceptron, with the results summarized in Table 3. Among them, the Gradient Boosting model stands out by attaining the highest test accuracy and average F1 score.

Despite its superior performance, it is essential to consider the interpretability aspect of models in practical applications. In this regard, the Random Forest model demonstrates notable strengths, offering a more comprehensive and intuitive understanding of feature importance and model decisions.

Models	**Avg_Train Accuracy**	**Test Accuracy**	**Avg_F1 Score**
Random Forest	0.962	0.962	0.960
Gradient Boosting	0.965	0.965	0.970
Multilayer Perceptron	0.934	0.941	0.940
Feature Set: heart rate, hand temp. (h_temp), ankle temp.(a_temp)			

Table 3. Results of Final Models

Moreover, the Random Forest model excels in better generalization. The learning curves depicted in Figure 7 reveal that the Gradient Boosting model (right plot) exhibits signs of overfitting on the training data, as indicated by a slight decrease in training accuracy with an increased sample size. Conversely, the Random Forest model demonstrates superior performance in the learning curve, with both training accuracy and validation accuracy increasing until reaching a point of stability. Additionally, the generalization gap between the two accuracies diminishes over time, indicating improved generalization ability.

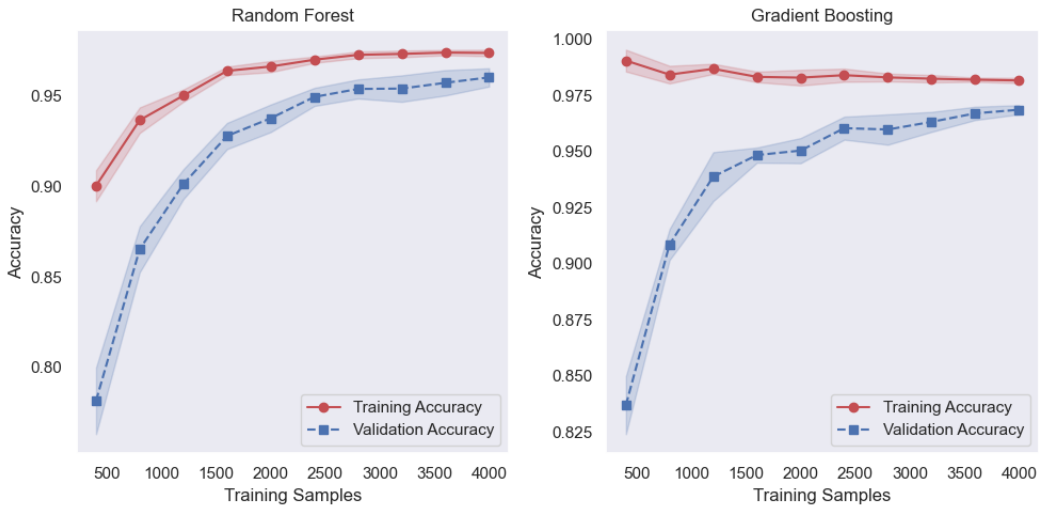


Figure 7. The Learning Curves of Random Forest (left) and Gradient Boosting (right)

The superiority of the Random Forest model is evident in both the confusion matrix (Figure 8) and the report table (Table 4). It consistently delivers high accuracy performance across different classes, with minimal variations. However, one notable point of confusion arises between the 'cycling' and 'rope jumping' activities. Specifically, the model misclassifies 11 data points labeled as 'rope jumping', with 9 of them being mistaken as 'cycling' and 2 as 'ascending stairs'.

This confusion highlights a common limitation in activity recognition systems, where certain postures or movements may be grouped into a single activity class due to their similarities.

Distinguishing these activities accurately would necessitate additional sensors, such as on the thigh, to capture more nuanced features [10].

Despite this challenge, the overall performance of the Random Forest model remains impressive, demonstrating its capability to generalize well to various activities.

Moreover, it is essential to take note of the fact that the final model utilized only three features: heart rate, hand temperature, and ankle temperature. This indicates that other features, such as acceleration, magnetometers, and gyroscopes, do not significantly contribute to physical activity prediction in this context.

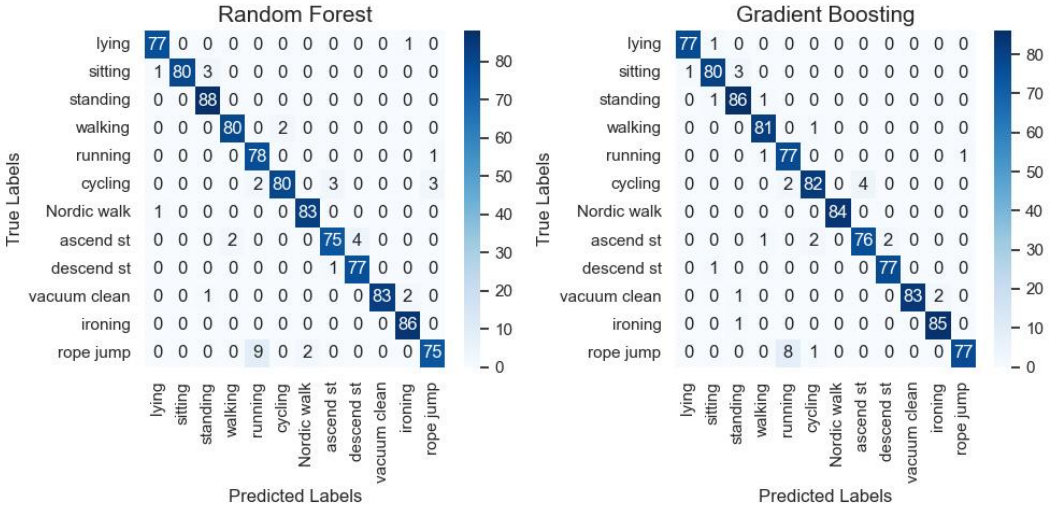


Figure 8. The Confusion Matrix of Gradient Boosting (left) and Random Forest (right)

Labels	Precision	Recall	F1-score	Support
Lying	0.97	0.99	0.98	78
Sitting	1.00	0.95	0.98	84
Standing	0.96	1.00	0.98	88
Walking	0.98	0.98	0.98	82
Running	0.88	0.99	0.93	79
Cycling	0.98	0.91	0.94	88
Nordic walking	0.98	0.99	0.98	84
Ascending stairs	0.95	0.93	0.94	81
Descending stairs	0.95	0.99	0.97	78
Vacuum cleaning	1.00	0.97	0.98	86
Ironing	0.97	1.00	0.98	86
Rope jumping	0.95	0.87	0.91	86

Table 4. The Report of Random Forest

6 Conclusions

This report discusses various supervised learning algorithms used for sensor-based human activity recognition. Model-based feature selection was conducted, revealing that only 3 features: heart rate, hand temperature, ankle temperature are necessary for achieving accurate predictions, with random forests exhibiting the highest performance among the models. In other words, even without utilizing sensor information such as acceleration, gyroscope, and magnetometer, the Random Forest model can still accurately predict physical activities.

The most valuable lesson I have gained from this project is the importance of feature selection. I have explored multiple methods, such as correlation analysis, Principal Component Analysis (PCA), feature importance, and Recursive Feature Elimination (RFE), to identify the most relevant features that significantly influence model performance. However, it is crucial to recognize that each method has its own assumptions and measurements, which may not be universally applicable in all situations. As a result, different methods may select different feature sets, leading to variations in model performance. As an example, when using the decision tree model for feature selection, the feature set obtained through the RFE method outperforms the feature set derived from feature importance. Understanding the strengths and limitations of each feature selection approach is essential to make informed decisions and achieve optimal results.

Future work on this project would likely to analyze the impact of imbalanced sample size of activities on model performance and feature selection.

Moreover, instead of individually predicting each activity, an alternative approach is to group activities based on posture. For example, activities like running, rope jumping, and ascending stairs can be grouped together, while activities like vacuum cleaning and ironing can be grouped separately. By first predicting the movement of each group as a whole and then predicting the movement within each group, the model can achieve better generalization ability and performance.

References

[1] Dempsey, Paddy C., et al. "Managing sedentary behavior to reduce the risk of diabetes and cardiovascular disease." Current diabetes reports 14 (2014): pp.1-11.

[2] Wu, Wanmin, et al. "Classification accuracies of physical activities using smartphone motion sensors." Journal of medical Internet research 14.5 (2012): e2208.

[3] Shoaib, Muhammad, et al. "Fusion of smartphone motion sensors for physical activity recognition." Sensors 14.6 (2014): pp. 10146-10176.

[4] Daghistani, T., & Alshammari, R. (2016). Improving Accelerometer-Based Activity Recognition by Using Ensemble of Classifiers. International journal of advanced computer science and applications, 7(5), pp.128-133.

[5] Chawla, J., & Wagner, M. Using Machine Learning Techniques for User Specific Activity Recognition. In Proceedings of the Eleventh International Network Conference (INC 2016): pp. 25.

[6] Reiss,Attila. (2012). PAMAP2 Physical Activity Monitoring. UCI Machine Learning Repository. https://doi.org/10.24432/C5NW2H.

[7] Jose, Anthony D., Frank Stitt, and D. Collison. "The effects of exercise and changes in body temperature on the intrinsic heart rate in man." American heart journal 79.4 (1970): 488-498.

[8] Sun, Yanmin, Andrew KC Wong, and Mohamed S. Kamel. "Classification of imbalanced data: A review." International journal of pattern recognition and artificial intelligence 23.04 (2009), pp. 687-719.

[9] Sharma, Vinod. "A Study on Data Scaling Methods for Machine Learning." International Journal for Global Academic & Scientific Research 1.1 (2022): pp. 23-33.

[10] Reiss, Attila, Gustaf Hendeby, and Didier Stricker. "A competitive approach for human activity recognition on smartphones." European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2013), 24-26 April, Bruges, Belgium. ESANN, 2013.

[11] Sun, L.; Zhang, D.; Li, B.; Guo, B.; Li, S. Activity Recognition on an Accelerometer Embedded Mobile Phone with Varying Positions and Orientations. In Ubiquitous Intelligence and Computing; Yu, Z., Liscano, R., Chen, G., Zhang, D., Zhou, X., Eds.; Springer: Berlin/Heidelberg, Germany,2010; Volume 6406, Lecture Notes in Computer Science, pp. 548–562.

[12] Reiss,Attila. Personalized mobile physical activity monitoring for everyday life. Diss. Ph. D. dissertation], Technical University of Kaiserslautern, (2014): pp. 39-40.