

SUPERVISED LEARNING FOR HUMAN ACTIVITY RECOGNITION USING SENSOR DATA

Project submitted

In partial fulfilment of the requirements for the degree of

MASTER OF COMPUTER APPLICATION

By

SOUMIK TARAFDER

CLASS ROLL NO: 002110503025

EXAM ROLL NO: MCA2340046

REGISTRATION NO: 160130 of 2021-22

Under the supervision of

PROF. SUSMITA GHOSH

Department of Computer Science & Engineering

Faculty of Engineering and Technology

Jadavpur University, Kolkata-700032, India

May, 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

TO WHOM IT MAY CONCERN

I hereby recommend that the project entitled "**SUPERVISED LEARNING FOR HUMAN ACTIVITY RECOGNITION USING SENSOR DATA**" prepared under my supervision by SOUMIK TARAFDER (Reg. No. 160130 of 2021-22, Roll No: 002110503025) may be accepted in partial fulfilment for the degree of Master of Computer Application in the Department of Computer Science and Engineering, Jadavpur University.

.....
Professor Susmita Ghosh
Project Supervisor
Department of Computer Science and Engineering
Jadavpur University

.....
Professor Nandini Mukhopadhyay
Head
Computer Science and Engineering
Jadavpur University

.....
Professor Ardhendu Ghosal
Dean
Faculty of Engineering and Technology
Jadavpur University

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

CERTIFICATE OF APPROVAL

The foregoing project is hereby accepted as a credible study of an engineering subject carried out and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein, but approve the project only for the purpose for which it is submitted.

Supervisor

External Examiner

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING AND TECHNOLOGY
JADAVPUR UNIVERSITY

DECLARATION OF ORIGINALITY
AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this project entitled “**SUPERVISED LEARNING FOR HUMAN ACTIVITY RECOGNITION USING SENSOR DATA**” contains literature survey and original research work by the undersigned candidate, as part of his Master of Computer Application studies.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Candidate's Name: Soumik Tarafder

Roll No: 002110503025

Project Title: Supervised Learning for Human Activity Recognition Using Sensor Data

Signature with Date:

ACKNOWLEDGEMENT

I wish to record my heartfelt gratitude towards all the people who have helped me in completion of this project.

I offer sincere thanks to my guide Prof. Susmita Ghosh for her kind support, inspiration and thoughtful insights.

I offer sincere thanks to Professor Nandini Mukhopadhyay, Head of the Department Computer Science and Engineering, Jadavpur University, for his continuous support and guidance throughout the project duration.

I am extremely thankful to Department of Computer Science and Engineering, Jadavpur University for providing all the facilities and necessary help towards the development of this project.

I am grateful to my parents, family members and friends for their continuous support, inspiration, encouragement and blessings without which this project would not have been success.

Soumik Tarafder

Registration No.: 160130 of 2021-22

Roll No.: 002110503025

Department of Computer Science & Engineering

Jadavpur University

Index

1. Introduction.....	9
1.1. Problem Definition.....	9
1.2. Scope of Report.....	9
1.3. Organization of Report.....	9
2. Literature Survey.....	11
2.1. Background.....	11
2.2. Types of Learning.....	11
2.2.1. Supervised Learning.....	11
2.2.2. Unsupervised Learning.....	12
2.3. Machine Learning.....	13
2.3.1. Supervised Machine Learning Techniques.....	13
2.3.1.1. KNN.....	13
2.3.1.2. Neural Networks.....	13
2.3.2. Unsupervised Machine Learning Techniques.....	13
2.3.2.1. K-Means Clustering.....	13
2.3.2.2. Anomaly detection.....	14
2.4. Existing Techniques.....	14
3. Classifiers for Human Activity Recognition.....	15
3.1. Workflow.....	15
3.1.1. Data Collection.....	15
3.1.2. Data Preprocessing.....	15
3.1.3. Labeling.....	15
3.1.4. Model Training.....	15

3.1.5. Model Evaluation.....	15
3.2. Flowchart.....	16
3.2.1. K-Nearest Neighboring.....	16
3.2.2. Random Forest.....	17
3.2.3. Linear Support Vector Machine.....	18
3.3. Classifier Used.....	18
3.3.1. K-Nearest Neighboring.....	18
3.3.2. Random Forest.....	19
3.3.3. Linear Support Vector Machine.....	20
4. Result and Analysis.....	22
4.1. Introduction.....	22
4.2. Dataset Used.....	23
4.2.1. Activities.....	23
4.2.2. Understanding the dataset.....	24
4.2.3. Exploratory Data Analysis.....	24
4.3. Performance Metrics.....	27
4.4. Analysis.....	31
5. Conclusion and Future Direction	
5.1. Conclusion.....	32
5.2. Future Direction.....	33
6. References.....	35

List of Table

1. Table 4.1--Performance of Classifiers Used.....27

List of Figures

1. Fig 4.1--Number of times each activity done.....25
2. Fig 4.2--Number of times each activity done by each subject.....25
3. Fig 4.3--Box Plot of Mean Acceleration.....26
4. Fig 4.4—Frequency Density of Mean Body Acceleration.....26
5. Fig 4.5—Confusion Matrix for KNN.....28
6. Fig 4.6—Classification Report for KNN.....28
7. Fig 4.7—Confusion Matrix for Linear SVM.....29
8. Fig 4.8—Classification Report for Linear SVM.....29
9. Fig 4.9—Confusion Matrix for Random Forest.....30
10. Fig 4.10—Classification Report for Random Forest.....30

List of Algorithms

1. Algorithm 3.1—K-Nearest Neighboring.....18
2. Algorithm 3.2—Random Forest.....19
3. Algorithm 3.3—Linear Support Vector Machine.....20

1. Introduction

1.1. Problem Definition

Human activity recognition is the task of automatically identifying and classifying the different activities that a person is performing based on sensor data collected from wearable devices. The problem involves analyzing data from various sensors such as accelerometers and gyroscopes, and then using machine learning algorithms to classify the activities [6]. The goal is to accurately recognize activities such as walking, running, sitting, standing, and other common activities that humans engage in.

The challenge in human activity recognition is to develop a robust and accurate algorithm that can deal with the variability and complexity of human movements, as well as the noise and inaccuracies of sensor data. Furthermore, the algorithm must be able to adapt to individual differences in how people perform activities and generalize well to new environments.

The solution to this problem has many potential applications, including health monitoring, fitness tracking, and activity recognition in smart homes and other intelligent environments.

1.2. Scope of Report

Machine Learning techniques have been used throughout this project and so this project contains the results of Machine Learning algorithms such as K-Nearest Neighborhood, Linear Support Vector Machine, Random Forest. Report contains analysis of the results of the classifiers based on these algorithms.

1.3. Organization of Report

1.3.1. Literature Survey

Earlier Methodology, Machine Learning techniques, Types of Learning (Supervised and Unsupervised), Existing Techniques are mentioned in this section.

1.3.2. Proposed Techniques for Human Activity Recognition

Workflow, Classifiers that are used in this project, Flowcharts are stated in this section as the proposed techniques to recognize Human Activity.

1.3.3. Result and Analysis

Results that are found after completion of this project and analysis of the results are stated in this section of the report.

1.3.4. Conclusion and Future Direction

The ultimate conclusion that is got after the completion of the project and what can be the future directions are mentioned in this section.

2. Literature Survey

2.1. Background

Human Activity Recognition (HAR) using sensor data is a popular research area in the field of machine learning and artificial intelligence. HAR involves the use of sensor data from devices such as accelerometers, gyroscopes, and magnetometers to identify and classify human activities, such as walking, running, sitting, and standing.

Supervised learning is a popular approach for HAR using sensor data, where labeled data is used to train a machine learning model that can then be used to classify new, unlabeled data. A variety of supervised learning algorithms have been used for HAR, including decision trees, support vector machines, random forests, and neural networks [6].

In recent years, deep learning approaches, particularly convolutional neural networks (CNNs) [10] and recurrent neural networks (RNNs), have shown promising results for HAR using sensor data. CNNs are well-suited for processing sensor data, as they can learn spatial features from the data, while RNNs can capture temporal dependencies in the data.

Several benchmark datasets are available for HAR using sensor data, including the UCI Human Activity Recognition Using Smartphones dataset [9], the WISDM dataset [4], and the Opportunity dataset. These datasets have been used to evaluate the performance of various supervised learning algorithms and deep learning approaches for HAR.

Overall, the literature on supervised learning for HAR using sensor data is extensive, with a wide range of approaches and techniques being explored. The field continues to evolve as researchers work to develop more accurate and robust models for HAR using sensor data.

2.2. Types of Learning

2.2.1. Supervised

Supervised learning is a type of machine learning in which an algorithm is trained on labeled data to make predictions or decisions about new, unseen data. In supervised learning, the algorithm is provided with a set of inputs (also known

as features or predictors) and corresponding outputs (also known as labels or targets) and learns to map the inputs to the outputs. The goal of supervised learning is to develop a model that can accurately predict the output for new inputs.

The process of supervised learning involves several steps, including:

1. Data collection: Collecting data that includes both the input features and corresponding output labels.
2. Data preparation: Preprocessing and cleaning the data to remove any noise or outliers, and splitting the data into training and testing sets.
3. Model selection: Choosing a suitable model architecture and algorithm for the problem at hand.
4. Training: Feeding the labeled training data into the model to learn the relationship between the input features and output labels.

2.2.2. Unsupervised

Unsupervised learning is a type of machine learning in which the algorithm learns patterns or relationships from unlabeled data without explicit guidance or supervision. In unsupervised learning, the algorithm is given a set of inputs (also known as features or predictors) without any corresponding output labels, and it must discover the underlying structure or patterns in the data.

The goal of unsupervised learning is to identify interesting or meaningful patterns or relationships in the data. This can include clustering similar data points together, identifying anomalies or outliers in the data, or reducing the dimensionality of the data to simplify further analysis.

The process of unsupervised learning involves several steps, including:

1. Data collection: Collecting data that includes only the input features without any corresponding output labels.
2. Data preparation: Preprocessing and cleaning the data to remove any noise or outliers.
3. Model selection: Choosing a suitable model architecture and algorithm for the problem at hand.
4. Evaluation: Evaluating the model's performance by analyzing the output and comparing it to known patterns or relationships in the data.

Unsupervised learning is useful when there is no labeled data available, or when the underlying patterns or structure in the data are not well understood.

Unsupervised learning algorithms are widely used in fields such as image processing [14], natural language processing, and anomaly detection.

2.3. Machine Learning

Machine learning is a field of artificial intelligence that focuses on the development of algorithms and models that allow computers to learn from data without being explicitly programmed. The aim of machine learning is to develop algorithms that can learn from data and make predictions or decisions based on that data. There are several types of machine learning, including supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning.

2.3.1. Supervised Machine Learning Techniques

2.3.1.1. k-Nearest Neighbors (k-NN)

k-NN is a simple and intuitive classification algorithm [6]. It classifies new instances based on the class labels of their nearest neighbors in the feature space. In HAR, k-NN can be used to find similar instances based on sensor data patterns and classify activities accordingly.

2.3.1.2. Neural Networks

Deep learning techniques, such as Convolutional Neural Networks (CNNs) [10] [12] and Recurrent Neural Networks (RNNs) [15], have shown promising results in HAR. CNNs [13] are effective for analyzing sensor data with a spatial structure, such as accelerometer data, by capturing local patterns through convolutional layers. RNNs [5] [7], on the other hand, are suitable for sequential data, such as time series data from sensors [11], as they can capture temporal dependencies.

2.3.2. Unsupervised Machine Learning Techniques

2.3.2.1. K-Means Clustering

K-Means Clustering is a popular unsupervised learning algorithm used for clustering. It aims to group similar data points into K clusters based on their similarity in terms of their features.

2.3.2.2. Anomaly Detection

Anomaly detection is an unsupervised learning algorithm used to identify unusual or unexpected data points in a dataset that do not conform to the general pattern.

In this project, we will be working on some supervised machine learning algorithm to predict several Human Activities like walking, standing, walking downstairs, laying, walking upstairs etc.

2.4. Existing Techniques

There are various existing techniques for HAR that utilize different sensor modalities and machine learning approaches. Here are some of the commonly used techniques:

1. **Wearable sensors:** One of the most widely used techniques for HAR is utilizing wearable sensors, such as accelerometers, gyroscopes, and magnetometers, attached to the body or clothing of the user. These sensors capture the motion of the body and provide information about the orientation and movement of the user [3] [8].

2. **Smartphone sensors:** Smartphones have become a popular platform for HAR due to their widespread availability and built-in sensors, such as accelerometers and GPS. Smartphone-based HAR systems can collect data continuously and provide insights into users' daily activities [1] [2].

3. **Computer vision:** Computer vision techniques, such as image and video analysis, have been used to recognize human activities based on visual information. These techniques can use cameras or RGB-D sensors to capture the motion of the user and identify activities such as walking, running, or sitting.

Overall, the choice of technique for HAR depends on the specific application, the available sensors, and the required accuracy and robustness of the system. Recent studies have shown that deep learning techniques, in combination with wearable or smartphone sensors, can significantly improve the accuracy and applicability of HAR systems.

3. Classifiers for Human Activity Recognition

3.1. Workflow

The workflow for Human Activity Recognition (HAR) typically involves several steps, including data collection, preprocessing, feature extraction, model training, and evaluation. Here is a general outline of the workflow:

3.1.1. Dataset Collection

After doing a lot of survey over various dataset, the dataset is chosen on which the project work has been done.

3.1.2. Data Preprocessing

Data was preprocessed and all the values were normalized. To check whether the data is balanced or not plotting has been done.

3.1.3. Labeling

In supervised learning scenarios, the preprocessed data needs to be labeled with corresponding activity labels. This labeling has been done in this stage.

3.1.4. Model Training

With the labeled data and extracted features, a machine learning model is trained to recognize activities. The choice of the model depends on the specific problem, available data, and desired performance. In this project we have worked with KNN, Linear SVM and Random Forest Classifiers.

3.1.5. Model Evaluation

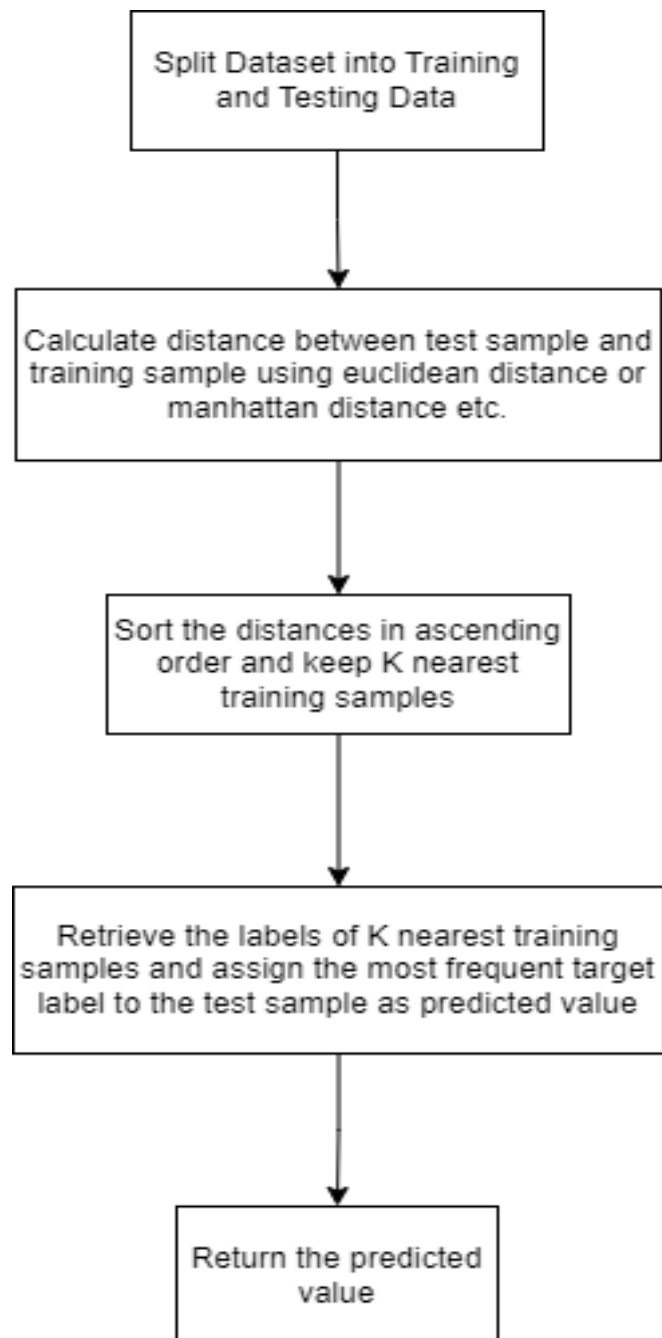
Once the model is trained, it needs to be evaluated to assess its performance. This is typically done by splitting the labeled data into training and testing sets. The model trained using KNN, Linear SVM and Random Forest algorithm are then evaluated on the testing set, and metrics such as accuracy, precision, recall, or F1-score are computed to measure its performance. Cross-validation or other evaluation techniques like GridSearchCV are used to ensure robustness of the model.

It's important to note that the workflow may vary depending on the specific requirements and constraints of the HAR problem. Different techniques and

algorithms may be used at each step, and iterations between steps may be necessary to refine the models and improve performance.

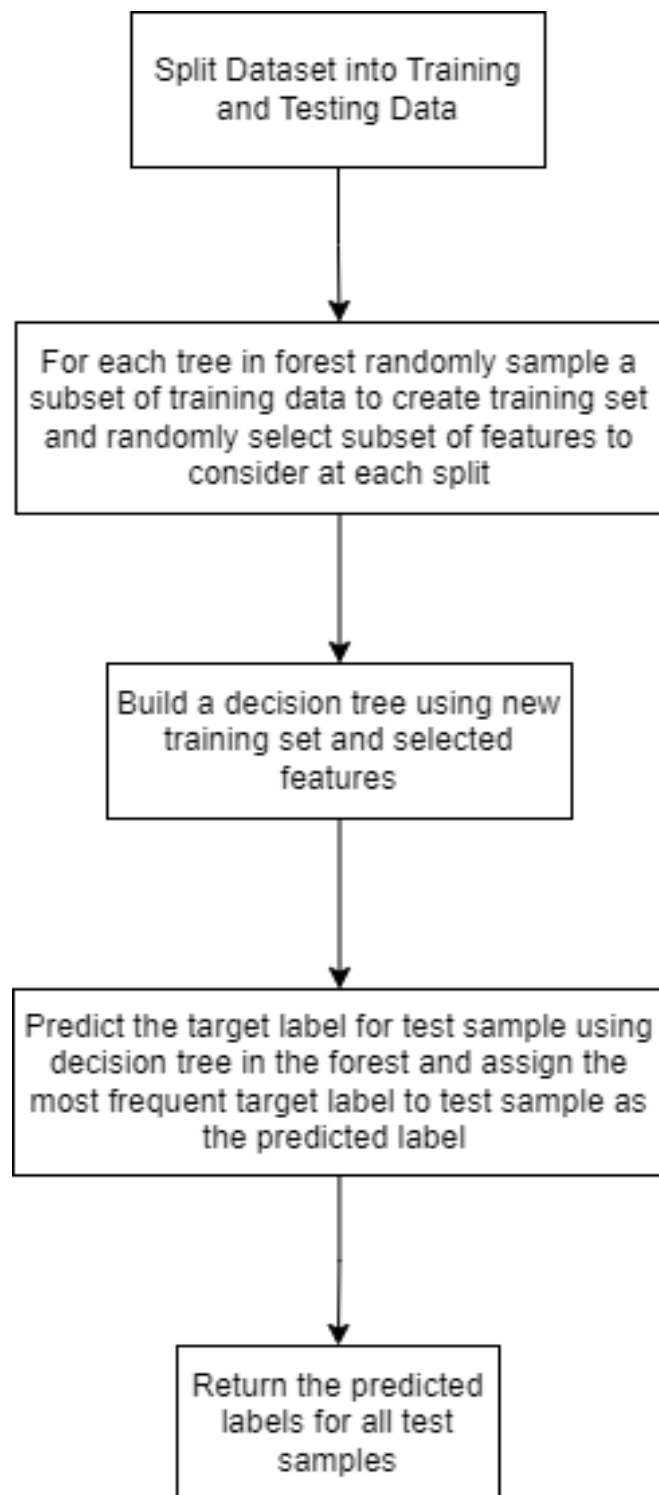
3.2. Flowchart

3.2.1. KNN



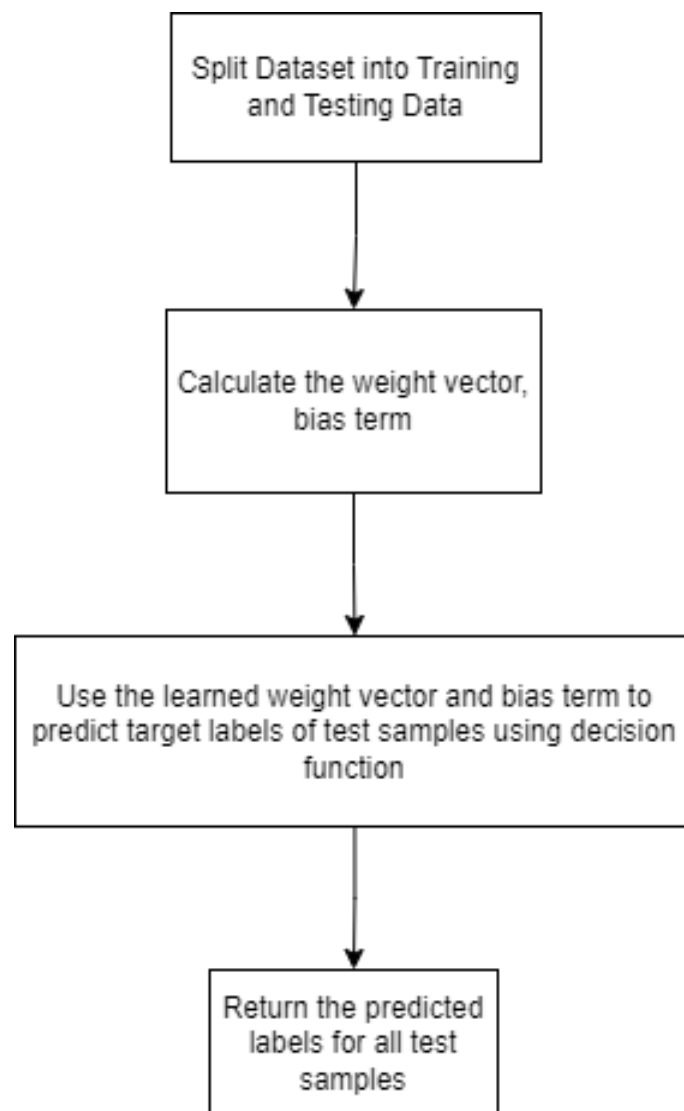
Flowchart For KNN Algorithm

3.2.2. Random Forest



Flowchart For Random Forest Algorithm

3.2.3. Linear SVM



Flowchart For Linear SVM Algorithm

3.3. Classifiers Used

3.3.1. KNN

Inputs:

- Training dataset: a matrix containing the feature values of the training data
- Target labels of the training dataset: a vector containing the target labels of the training data

- Test dataset: a matrix containing the feature values of the test data
- Number of nearest neighbours to consider (K)

Output:

- Predicted target labels for the test data: a vector containing the predicted target labels for the test data

Algorithm:

1. For each test sample:

- Calculate the distance between the test sample and each training sample using a distance metric (e.g. Euclidean distance).
The Euclidean distance between p and q is given by:
$$d(p, q) = \sqrt{(p - q)^2}$$
- Sort the distances in ascending order and keep the indices of the K nearest training samples.
- Retrieve the target labels of the K nearest training samples.
- Assign the most frequent target label to the test sample as the predicted label.

2. Return the vector of predicted labels for all test samples.

3.3.2. Random Forest

Inputs:

- Training dataset: a matrix containing the feature values of the training data
- Target labels of the training dataset: a vector containing the target labels of the training data
- Test dataset: a matrix containing the feature values of the test data
- Number of trees in the forest
- Maximum depth of each tree
- Number of features to consider at each split

Output:

- Predicted target labels for the test data: a vector containing the predicted target labels for the test data

Algorithm:

1. For each tree in the forest:
 - Randomly sample a subset of the training data (with replacement) to create a new training set.
 - Randomly select a subset of the features (without replacement) to consider at each split.
 - Build a decision tree using the new training set and the selected features.
2. For each test sample:
 - Predict the target label for the test sample using each decision tree in the forest.
 - Assign the most frequent target label to the test sample as the predicted label.
3. Return the vector of predicted labels for all test samples.

3.3.3. Linear SVM

Inputs:

- Training dataset: a matrix containing the feature values of the training data
- Target labels of the training dataset: a vector containing the target labels of the training data
- Test dataset: a matrix containing the feature values of the test data
- Regularization parameter (C)

Output:

- Predicted target labels for the test data: a vector containing the predicted target labels for the test data

Algorithm:

1. Normalize the feature values of the training dataset to have zero mean and unit variance.

2. Solve the primal form of the linear SVM optimization problem:

$$\min_{w, b} 0.5 * ||w||^2 + C * \sum(\max(0, 1 - y_i * (w.T * x_i + b))) \text{ for } i = 1 \text{ to } n$$
 subject to $y_i * (w.T * x_i + b) \geq 1$ for $i = 1$ to n ,

where w is the weight vector, b is the bias term, x_i is the i -th training sample, y_i is its corresponding target label, n is the number of samples, and C is the regularization parameter.

3. Use the learned weight vector and bias term to predict the target labels of the test samples using the decision function:

$$f(x) = w.T * x + b$$

If $f(x) \geq 0$, predict the positive class ($y = 1$); otherwise, predict the negative class ($y = -1$).

4. Return the vector of predicted labels for all test samples.

4. Results and Analysis

4.1. Introduction

Human Activity Recognition (HAR) is a field of study in computer science and artificial intelligence that focuses on developing techniques and models to automatically identify, classify, and understand human activities based on sensor data. HAR has gained significant attention due to its applications in various domains, including healthcare, sports, smart homes, surveillance, and assisted living.

The goal of HAR is to create systems that can recognize and interpret human activities in real-time, typically using data from sensors such as accelerometers, gyroscopes, magnetometers, and other wearable or ambient sensors. These sensors capture information about movements, orientations, and environmental factors that can be analysed to infer the underlying human activities.

The process of HAR involves several stages, including data collection, preprocessing, feature extraction, model training, and activity recognition. Data collection involves acquiring sensor data from individuals performing different activities in various settings. Preprocessing techniques are applied to clean, filter, and normalize the data to ensure its quality and consistency.

Feature extraction is a crucial step in HAR, where relevant information is extracted from the sensor data to represent each activity. These features can include statistical measures, frequency domain features, time-domain features, or other domain-specific features that capture the characteristics of the activities.

Machine learning techniques play a vital role in HAR. Once the models are trained, they can be used to recognize and classify human activities in real-time. The models take sensor data as input and predict the corresponding activity based on the patterns and knowledge learned during the training phase. The recognized activities can be further analysed and used to make decisions, provide feedback, or trigger specific actions in various applications.

The advancements in sensor technology, machine learning algorithms, and computing power have significantly improved the accuracy and efficiency of HAR systems. Researchers and practitioners continue to explore new techniques, models, and sensor modalities to enhance HAR performance, enable more complex activity recognition, and expand its applications to diverse domains.

HAR has the potential to revolutionize various industries by enabling personalized healthcare monitoring, smart environments, activity tracking, fall detection, and behaviour analysis. It can contribute to improving healthcare outcomes, optimizing sports performance, enhancing safety and security, and providing assistance to individuals with disabilities or special needs.

Overall, Human Activity Recognition is a multidisciplinary field that combines aspects of computer science, signal processing, pattern recognition, and machine learning to enable machines to understand and interpret human activities based on sensor data. It holds great promise for creating intelligent systems that can better understand and interact with humans in various contexts.

4.2. Dataset Used

The dataset can be downloaded from the following link

<https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones#>

Human Activity Recognition database is built from the recordings of 30 persons performing activities of daily living while carrying a waist-mounted smartphone with embedded inertial sensors (accelerometer and Gyroscope).

4.2.1. Activities

1. Walking
2. Walking Upstairs
3. Walking Downstairs
4. Sitting
5. Standing
6. Laying

Accelerometers detect magnitude and direction of the proper acceleration, as a vector quantity, and can be used to sense orientation (because direction of weight changes)

Gyroscope maintains orientation along a axis so that the orientation is unaffected by tilting or rotation of the mounting, according to the conservation of angular momentum.

Accelerometer measures the directional movement of a device but will not be able to resolve its lateral orientation or tilt during that movement accurately unless a gyro is there to fill in that info.

With an accelerometer you can either get a really "noisy" info output that is responsive, or you can get a "clean" output that's sluggish. But when you combine the 3-axis accelerometer with a 3-axis gyro, you get an output that is both clean and responsive in the same time.

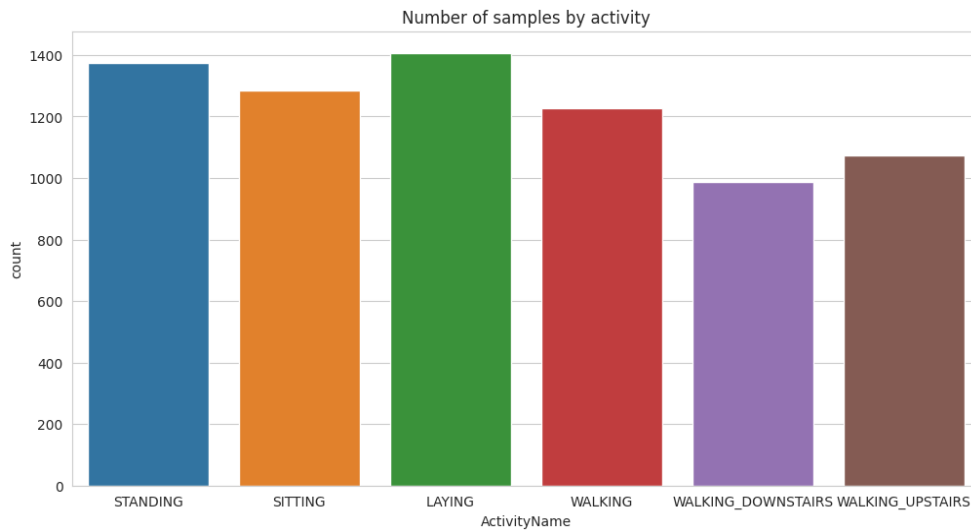
4.2.2. Understanding the dataset

- Both sensors generate data in 3 Dimensional space over time. Hence the data captured are '3-axial linear acceleration'(tAcc-XYZ) from accelerometer and '3-axial angular velocity' (tGyro-XYZ) from Gyroscope with several variations.
- prefix 't' in those metrics denotes time.
- suffix 'XYZ' represents 3-axial signals in X , Y, and Z directions.
- The available data is pre-processed by applying noise filters and then sampled in fixed-width windows(sliding windows) of 2.56 seconds each with 50% overlap. i.e., each window has 128 readings.
- The dataset was preprocessed previously. So all the features were normalized.
- Also the dataset was split into training and testing data in a ratio of 71 : 29 i.e. 71% data is used for training and 29 % data is used for testing

4.2.3. Exploratory Data Analysis

Exploratory Data Analysis (EDA) is the process of examining and analyzing data to summarize its main characteristics and uncover insights that may be hidden in the data. It involves visualizing and summarizing the data using various statistical and graphical techniques, without making any assumptions about the underlying distribution or relationships between variables. EDA is an important first step in data analysis, as it helps to identify data quality issues, outliers, and patterns that may be useful for developing models and making predictions.

4.2.3.1. Number of times activities has been done

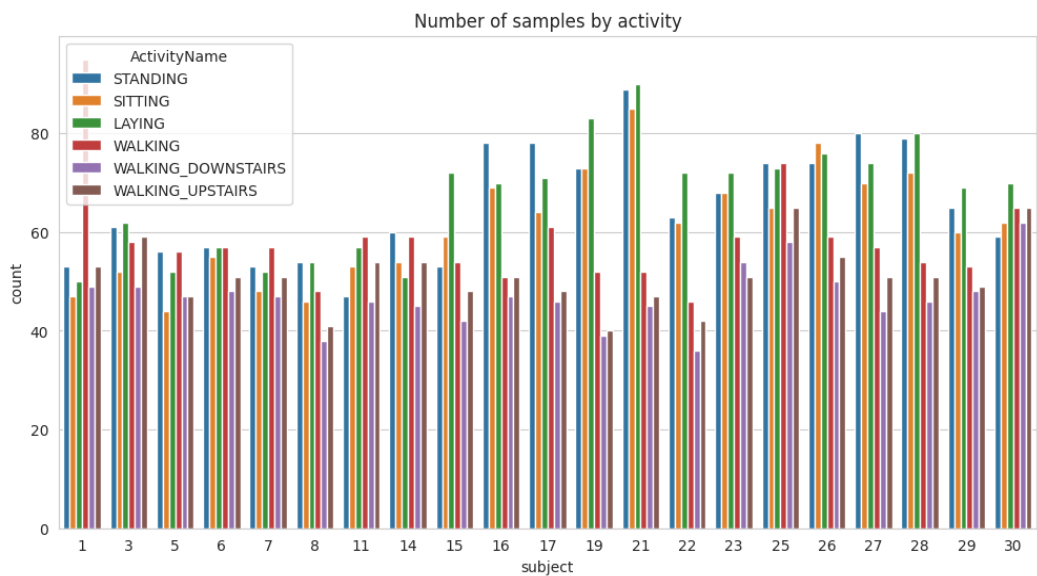


Bar Diagram for Count of each Activity (FIG 4.1)

Observations:

- All the activities have been done almost equal number of times
- This [FIG 4.1] implies that the data is almost balanced.

4.2.3.2. Activities done by each person

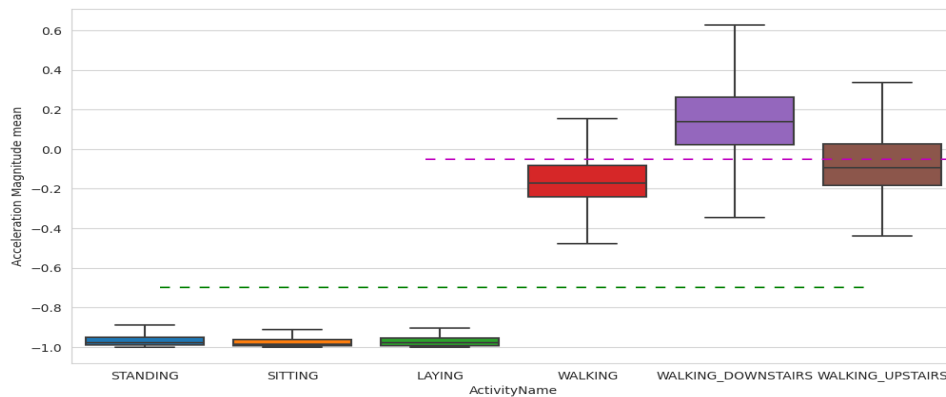


Bar Diagram for Activities done by each Subject (FIG 4.2)

Observations:

- We have got almost equal number of readings from all the subjects
- This indicates all the subjects has done almost equal no of activities for each activity
- This [FIG 4.2] implies the dataset is well balanced

4.2.3.3. Box plot of Acceleration Magnitude can separate Moving and Stationary activities

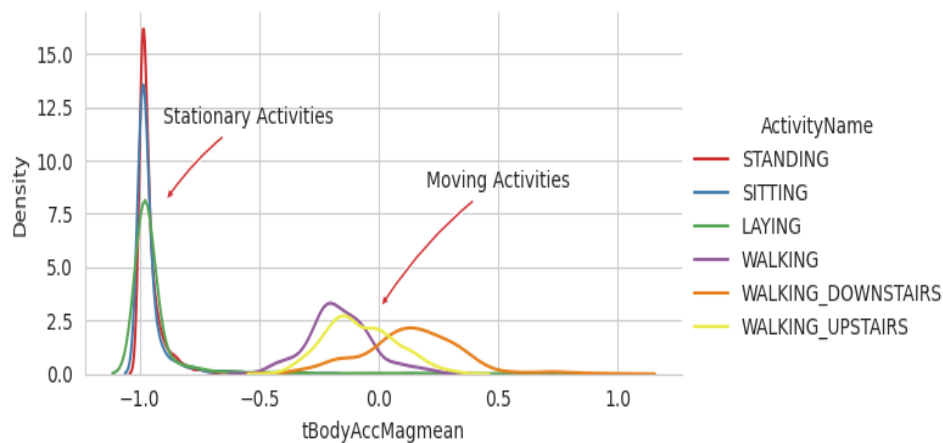


Box Plot for Mean Acceleration (FIG 4.3)

Observations:

- If Mean Acceleration is < -0.8 then the Activities are either Standing or Sitting or Laying [FIG 4.3].
- If Mean Acceleration is > -0.6 then the Activities are either Walking or Walking Downstairs or Walking Upstairs [FIG 4.3].
- If Mean Acceleration > 0.0 then the Activity is Walking Downstairs.

4.2.3.4. Differentiating Moving and Stationary activities by Mean Body Acceleration



Frequency Density in Mean Acceleration (FIG 4.4)

Observations:

- Stationary Activities have overlapping data with respect to Mean Body Acceleration.
- This implies [FIG 4.4] that while predicting stationary activities will provide higher numbers of error

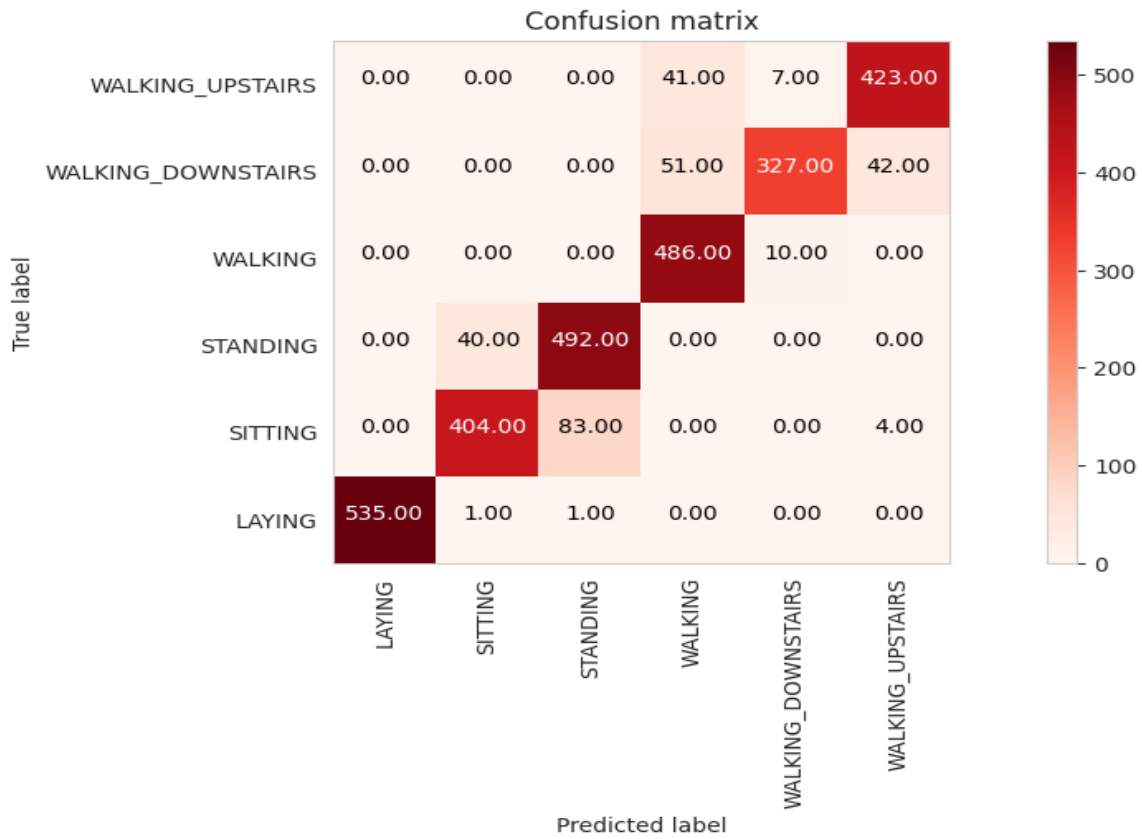
4.3. Performance Metrics

SUBJECT	KNN	LINEAR SVM	RANDOM FOREST
PARAMETER	neighbors = 14	C = 0.5	depth = 9 trees = 70
TRAINING TIME (HH:MM:SS.MS)	00:00:12.79	00:00:41.63	00:02:35.40
TESTING TIME (HH:MM:SS.MS)	00:00:00.76	00:00:00.01	00:00:00.03
MEAN ACCURACY	90.50%	96.62%	91.37%
STANDARD DEVIATION	0.000025	0.17	0.46
PRECISION	0.91	0.97	0.92
RECALL	0.90	0.97	0.92
F1-SCORE	0.90	0.97	0.92

Performance of Classifiers Used (Table 4.1)

(Mean Accuracy has been calculated as the average of accuracy of 20 iterations, standard deviation is calculated over the accuracies got over the 20 iterations. Parameter, training time, testing time, precision, recall and f1-score has been given for which best accuracy has been achieved)

4.3.1. KNN :



Confusion Matrix for Best Estimator in KNN (FIG 4.5)

```

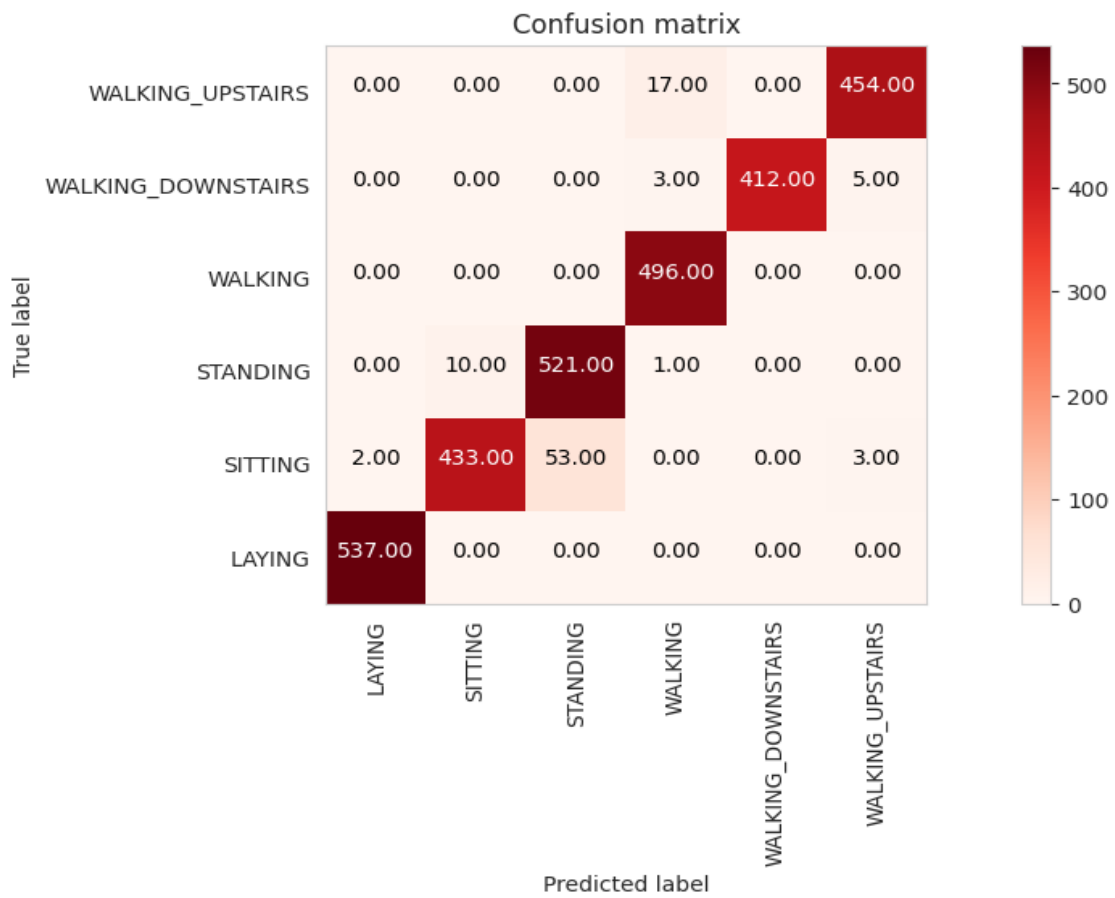
=====Classification Report=====
              precision    recall  f1-score   support

   LAYING           1.00      1.00      1.00         537
   SITTING           0.91      0.82      0.86         491
   STANDING          0.85      0.92      0.89         532
   WALKING           0.84      0.98      0.91         496
 WALKING_DOWNSTAIRS  0.95      0.78      0.86         420
 WALKING_UPSTAIRS   0.90      0.90      0.90         471

 accuracy              0.90         2947
 macro avg             0.91      0.90      0.90         2947
 weighted avg          0.91      0.90      0.90         2947
  
```

Classification Report for Best Estimator in KNN (FIG 4.6)

4.3.2. LINEAR SVM :



Confusion Matrix for Best Estimator in Linear SVM (FIG 4.7)

```

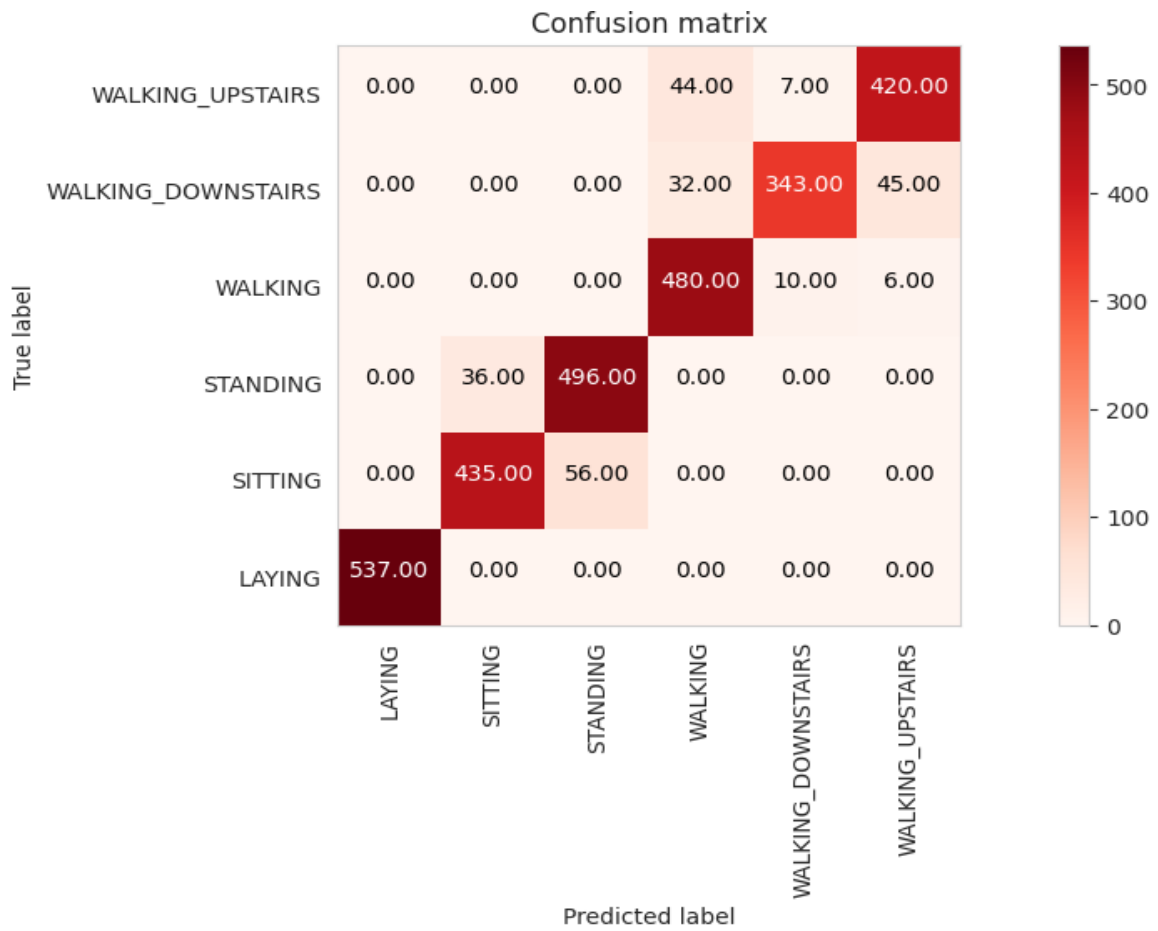
=====Classification Report=====
              precision    recall  f1-score   support

   LAYING           1.00      1.00      1.00         537
   SITTING           0.98      0.88      0.93         491
   STANDING          0.91      0.98      0.94         532
   WALKING           0.96      1.00      0.98         496
 WALKING_DOWNSTAIRS  1.00      0.98      0.99         420
 WALKING_UPSTAIRS   0.98      0.96      0.97         471

 accuracy                   0.97         2947
 macro avg                   0.97         2947
 weighted avg                 0.97         2947
  
```

Classification Report for Best Estimator in Linear SVM (FIG 4.8)

4.3.3. RANDOM FOREST :



Confusion Matrix for Best Estimator in Random Forest (FIG 4.9)

```

=====Classification Report=====
              precision    recall  f1-score   support

   LAYING              1.00      1.00      1.00         537
   SITTING              0.92      0.89      0.90         491
   STANDING              0.90      0.93      0.92         532
   WALKING              0.86      0.97      0.91         496
 WALKING_DOWNSTAIRS    0.95      0.82      0.88         420
 WALKING_UPSTAIRS     0.89      0.89      0.89         471

 accuracy                   0.92         2947
 macro avg                  0.92         2947
 weighted avg               0.92         2947
  
```

Classification Report for Best Estimator in Random Forest (FIG 4.10)

4.4. Analysis

In terms of Accuracy,

Linear Support Vector Machine is giving the best result with an accuracy of 96.62% and this implies that

- The activities can be distinguished best with a straight line with respect to the features
- Since the values of the features are accelerometer and gyroscopic data so different activities have different numeric values for all the features which eventually helped Linear Support Vector Machine to find the best result.

In terms of Standard Deviation,

Random Forest is having greatest value of Standard Deviation of 0.46 and this implies that

- Random Forest is the most vulnerable classifier to predict Human Activity as it tries different combinations of maximum depth and number of trees.
- Some operations in Random Forest such as calculating Impurity or entropy value for each node takes much time to complete the training.

K-nearest Neighborhood is providing least value of Standard Deviation as the accuracy of this classifier is almost same in every iteration.

Now

Precision = (True Positive) / (True Positive + False Positive)

Recall = (True Positive) / (True Positive + False Negative)

F1-Score = $2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$

Since Linear Support Vector Machine is providing best accuracy so Precision, Recall and F1-Score is best for Linear Support Vector Machine with a Precision, Recall and F1-score of 0.97 each.

5. Conclusion and Future Direction

5.1. Conclusion

In conclusion, supervised learning is a powerful and widely used approach in machine learning that has shown remarkable success in various applications. By providing labeled training data, supervised learning algorithms learn to generalize patterns and make predictions or classifications on new, unseen data. Here are some key points to summarize the benefits and implications of supervised learning

Supervised learning algorithms excel in tasks that require accurate predictions or classifications. They can leverage labeled data to learn complex relationships and make precise predictions on unseen data.

Supervised learning is applicable to a wide range of problem domains, including image and speech recognition, natural language processing, fraud detection, sentiment analysis, and medical diagnosis. It can be used for regression problems (predicting continuous values) and classification problems (assigning discrete labels).

Many supervised learning algorithms provide interpretability, allowing users to understand and interpret the reasoning behind the model's predictions. Linear regression, decision trees, and support vector machines are examples of algorithms that offer interpretable models.

Supervised learning provides a clear framework for evaluating the performance of models. Metrics such as accuracy, precision, recall, F1 score, and mean squared error are commonly used to assess model effectiveness and compare different algorithms.

Supervised learning heavily relies on labeled training data, which can be a limitation. Acquiring and annotating large amounts of labeled data can be time-consuming, costly, or even infeasible in certain scenarios. However, techniques like transfer learning, active learning, and semi-supervised learning can mitigate the data requirements to some extent.

Overfitting is a common challenge in supervised learning, where a model becomes overly complex and performs poorly on new, unseen data. Techniques like regularization, cross-validation, and early stopping are used to prevent overfitting and ensure good generalization.

Some supervised learning algorithms, particularly those based on deep neural networks, can be computationally intensive and require significant computational resources. Training large models on vast amounts of data may require distributed computing or specialized hardware.

Supervised learning models can inherit biases present in the labeled training data, leading to biased predictions and potentially discriminatory outcomes. Ensuring fairness, transparency, and accountability in the training data and model evaluation processes is crucial to address these ethical concerns.

In summary, supervised learning is a versatile and effective approach in machine learning that allows for accurate predictions and classifications in various domains. While it comes with challenges such as data requirements and potential biases, advances in algorithms and techniques continue to improve the capabilities and applicability of supervised learning in real-world scenarios.

5.2. Future Direction

The future direction for Human Activity Recognition (HAR) using supervised learning is promising, with ongoing advancements and research aimed at enhancing the accuracy, robustness, and applicability of HAR systems. Here are some potential future directions for HAR using supervised learning

1. **Deep Learning Architectures:** Deep learning, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs), has shown significant potential in HAR. Future research can focus on developing more advanced deep learning architectures that can effectively capture complex temporal and spatial dependencies in sensor data, leading to improved activity recognition performance.
2. **Transfer Learning and Domain Adaptation:** HAR systems often face challenges in generalizing to new environments or users due to variations in sensor data or activity patterns. Transfer learning and domain adaptation techniques can be explored to leverage knowledge from pre-trained models or source domains and adapt them to new domains or users with limited labeled data, enabling more robust and personalized HAR systems.
3. **Multimodal Sensor Fusion:** Integrating data from multiple sensors, such as accelerometers, gyroscopes, magnetometers, and ambient sensors, can provide a richer context for activity recognition. Future research can focus on developing supervised learning approaches that effectively fuse data from multiple modalities, enabling more accurate and comprehensive HAR systems.

4. Incremental and Online Learning: Traditional supervised learning approaches often assume fixed training and testing sets, limiting their adaptability to dynamic or evolving environments. Future HAR systems can explore incremental and online learning techniques that can continuously learn and update activity models as new data becomes available, allowing for adaptive and real-time recognition in changing scenarios.

5. Explainability and Interpretability: As supervised learning models become more complex and powerful, there is a growing need for explainability and interpretability. Future research can focus on developing techniques that provide insights into the decision-making process of HAR models, allowing users to understand and trust the system's predictions and aiding in debugging and fine-tuning the models.

6. Addressing Class Imbalance and Data Scarcity: In HAR, there can be class imbalance issues where certain activities are underrepresented in the training data. Future research can explore techniques to address class imbalance and data scarcity, such as data augmentation, active learning, and semi-supervised learning, to improve the performance of HAR models on minority classes and scenarios with limited labeled data.

7. Privacy and Security Considerations: HAR systems often involve collecting and analysing sensitive data, raising privacy and security concerns. Future research can focus on developing privacy-preserving techniques that allow for accurate activity recognition while ensuring the protection of personal information and mitigating the risks of unauthorized access or misuse of data.

Overall, the future of HAR using supervised learning lies in the development of advanced models and techniques that can handle complex temporal and spatial dependencies, adapt to new environments, integrate multimodal sensor data, provide explainability, and address practical challenges such as data scarcity and privacy concerns. These advancements will enable more accurate, personalized, and trustworthy HAR systems that can find applications in healthcare, sports, smart homes, and various other domains.

6. References

1. Kwon, Y., & Lee, K. (2019). Human activity recognition using smartphone sensors and deep neural networks. *Sensors*, 19(21), 4780.
2. Sathya, S., & Prasad, R. V. (2018). Human activity recognition using deep learning neural network based on smartphone sensor data. *Journal of Ambient Intelligence and Humanized Computing*, 9(6), 1885-1895.
3. Hossain, M. S., Muhammad, G., & Ahmed, M. M. (2019). A comprehensive survey on human activity recognition using wearable sensors. *Sensors*, 19(5), 1253.
4. Shoaib, M., Bosch, S., Incel, O. D., Scholten, J., & Havinga, P. J. (2016). A comprehensive study of activity recognition using accelerometers. *Information Fusion*, 33, 74-93.
5. Zhang, S., Zhai, J., & Zhang, W. (2019). Human activity recognition based on deep learning for healthcare. *Journal of healthcare engineering*, 2019, 1-15.
6. Zhang, X., Zhou, W., Wang, J., & Wang, Y. (2020). A comparative study of human activity recognition using smartphone sensors with machine learning algorithms. *Symmetry*, 12(1), 76.
7. Guo, X., Zhao, C., Zhao, Y., Song, H., & Wang, L. (2020). A human activity recognition method using an improved deep learning network based on sensor data. *Mobile Networks and Applications*, 25(2), 560-569.
8. Wang, C., Chen, Q., & Yan, W. Q. (2019). Human activity recognition using wearable sensors: a deep learning study. *Journal of Ambient Intelligence and Humanized Computing*, 10(4), 1353-1364.

9. Yang, G., Zhang, T., & Guo, F. (2019). Human activity recognition using smartphone sensors and ensemble learning. *Journal of Ambient Intelligence and Humanized Computing*, 10(2), 763-774.
10. Alsheikh, M. A., Nassef, M., Nait-Charif, H., & Al-Maadeed, S. (2017). Human activity recognition using wearable sensors by deep convolutional neural networks. *Sensors*, 17(6), 1304.
11. Panwar, R., Singh, S., & Dogra, D. P. (2019). Human activity recognition using convolutional neural networks and weighted sparse representation based on multi-sensor data. *Sensors*, 19(17), 3671.
12. Li, J., Liu, Z., Zhou, M., & Feng, G. (2020). Human activity recognition using a wearable sensor with an improved deep learning network. *Sensors*, 20(5), 1315.
13. Yang, S. H., & Kim, S. K. (2019). Human activity recognition using ensemble of multi-level deep neural networks with sensor fusion. *IEEE Access*, 7, 32606-32619.
14. Pham, T. H., & Kim, J. (2019). Human activity recognition using smartphone sensors and deep neural networks with dropout and stacked denoising autoencoders. *IEEE Access*, 7, 118638-118651.
15. He, L., & Zhang, S. (2020). Human activity recognition using a deep learning model based on smartphone sensor data. *Mobile Networks and Applications*, 25(1), 180-188.