The Automatic detection of human activity using accelerators sensors

[*Add contact details*]

[Add Date]

Table of contents

Abstract

Building human monitoring systems is so critical in many aspects regarding manufacturing, traffic, home safety, and many other fields. So, the accelerators' sensor readings have been utilized in building human monitoring systems that monitor different activities of humans like walking, standing, going upstairs/downstairs, etc. The readings of the sensors used to be recorded with their associated activity manually. So, in this research, we are working on the automation of human activity detection using different machine learning algorithms.

Keywords: accelerators, sensors, monitoring, machine learning

Introduction

The goal in this project is the detection of the different body activities in humans using only the measurement of the accelerators' sensors readings. This would be a great help for building the monitoring system that would help in finding if the person is walking, standing, talking with someone, working on the computer, going upstairs, going downstairs, etc. So, this monitoring system would be able to get all the different activities the human is doing which could be used in the hospitals to minor activities of the patients' bodies to know if any something abnormal, Also, it can be used in homes, streets, public places, etc. that would save a lot of resources either money or humans resources. So, we will try to build a robust machine learning model that is able to detect the activity type of the body based on the accelerators readings we got from the sensors. Currently, the scientists used to manually record the activity type that noticed from the 15 participation in this experiment. We will use the machine learning model in automating the process of detecting the activity typed that is done by the human.
The dataset used in this project is data generated from the experiment that uses accelerators sensors to monitor the activity types generated from five participants in these experiments who are holding these sensors in their chests. The scientist who is doing this experiment manually records the sensor readings and the noticed activity type and gives an identifier for each measurement and its related activity and all the measurements for each participant kept in the CSV file. So, the dataset contains from 15 CSV files each file include:

1- Sequential Number: a unique identifier for each measurement done for each participant in the experiment.
2- X acceleration: The value of the rate at which the human body change its speed with respect to time in the X direction
3- Y acceleration: The value of the rate at which the human body change its speed with respect to time in the Y direction
4- Z acceleration: The value of the rate at which the human body change its speed with respect to time in the Z direction
5- label: that contains 7 classes that range from 1 to 7 which each number represents the type of the activity.
This data like any other data have errors and problems so, we need to clean it and here are some of the steps that we will explain earlier:

1- Cleaning the missing values
2- Balancing the label
3- Scaling the features
4- Splitting the data

Methodology

Now, the different methods and techniques for the modeling and the analysis would be discussed and we will split this into two subsections the first section would discuss the analysis of the data and the hypothesis about it and the second section would discuss the methods and the techniques applied in the modeling:

A- Data Analysis

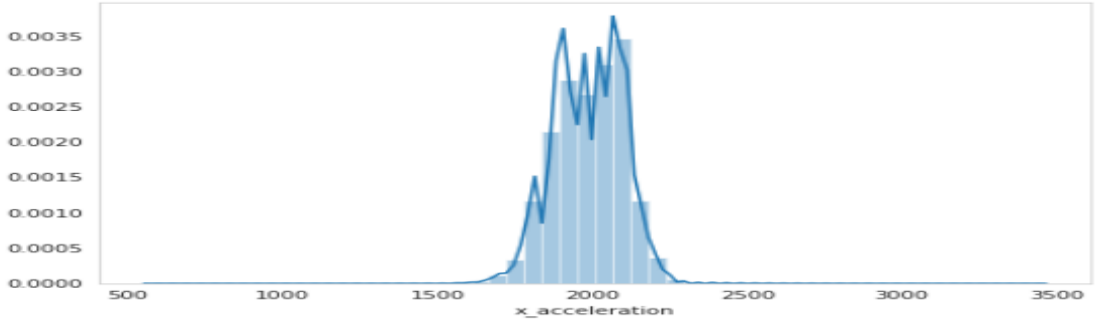
First, we have worked on each feature on the dataset independently:

1- We have got the basic statistics of the whole numerical features we have using the table below, in this table show us that; The three dimensions of the acceleration have different scale as can be noticed the x_acceleration ranges from 516 (minimum value) to 3222 (maximum value) with average value equals 1987.5 plus or minus 111.4 (standard deviation), while the y_acceleration have value ranges from 146 (minimum) to 3665 (maximum) with average 2382.7 plus or minus 100.1 (standard deviation), and the _acceleration have values range from 2 (minimum) to 3685 (maximum) with average 1970.48 plus or minus 94.34 (standard deviation).

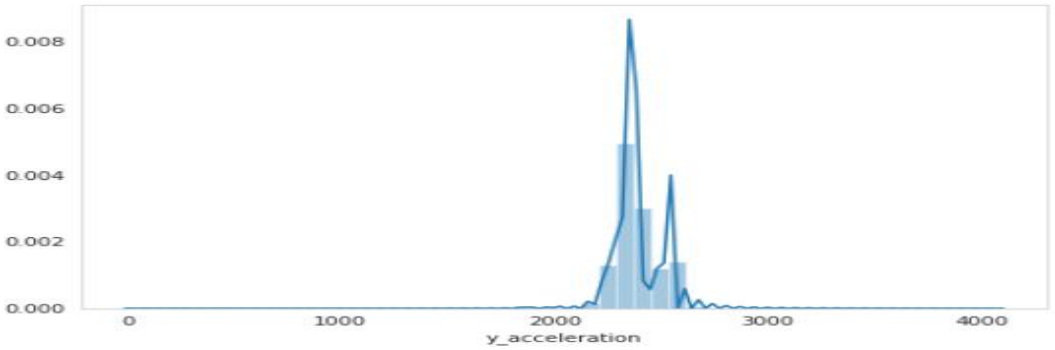
	sequential_number	x_acceleration	y_acceleration	z_acceleration
count	192690.000000	192690.000000	192690.000000	192690.000000
mean	67062.039104	1987.383035	2381.843349	1970.424168
std	41466.126876	111.361268	100.767572	94.176214
min	0.000000	583.000000	2.000000	2.000000
25%	32026.000000	1903.000000	2337.000000	1918.000000
50%	64194.000000	1991.000000	2366.000000	1988.000000
75%	98397.750000	2076.000000	2411.000000	2032.000000
max	166730.000000	3435.000000	4087.000000	3387.000000

“Figure 1: The basic statistics of the features”

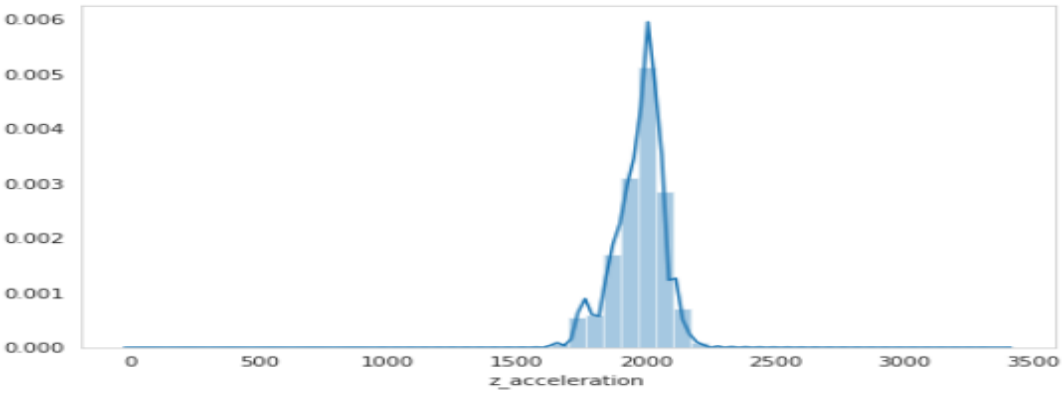
2- For elaborating more on each feature we have plotted the graph for the distribution for each numerical feature. As noticed from the distribution of the x-acceleration; the frequency of the values of the x-axis acceleration combat at values from 1700 to 2300, while the frequency of the values of the y-axis compact on the interval from almost 2200 to 2600, and for the z-acceleration the frequency of the distribution compact on the interval from 1600 to 2300. Which asserts that the three acceleration features have different scales of values.



“Figure 2: The distribution of x-acceleration values”

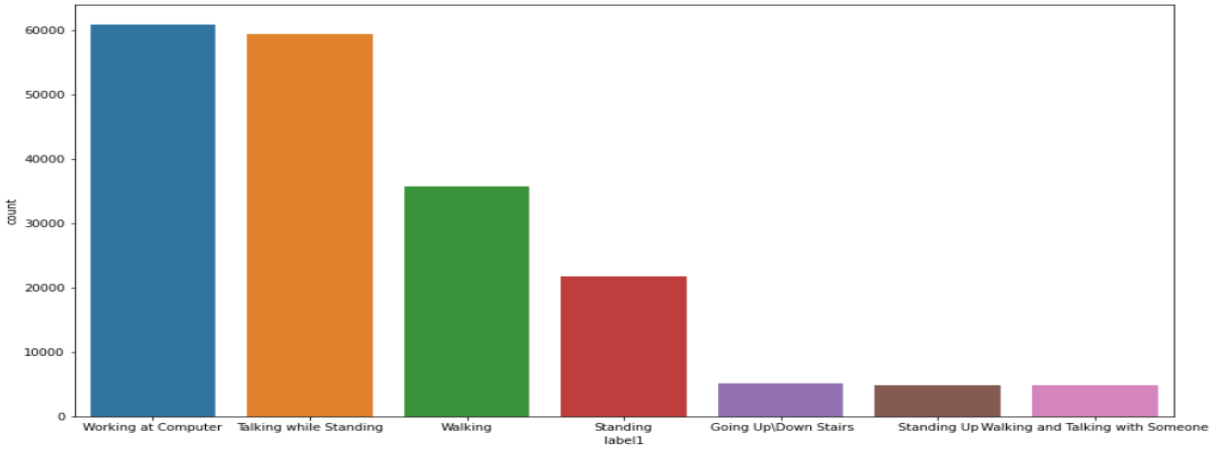


“Figure 3: The distribution of y-acceleration values”



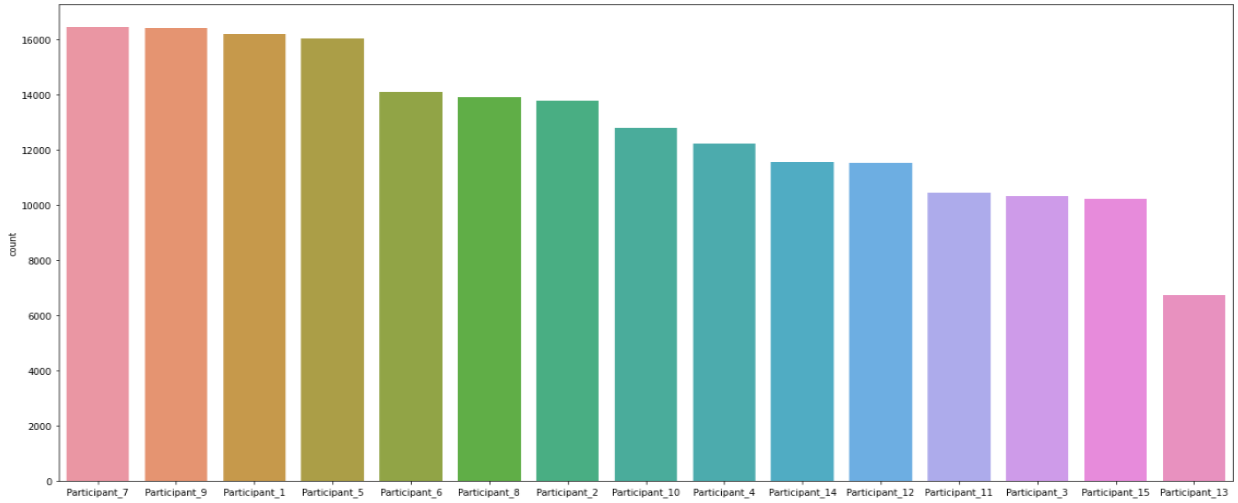
“Figure 4: The distribution of z-acceleration values”

3- According to the categorical features as seen in the below barplot the majority of activities done (label) was "working on the computer" and "talking while standing" activities while the activities with the lowest occurrence are "going up and downstairs", "Standing up" and "walking and talking with someone"



“Figure 5: The count plot of the label”

4- For another categorical variable that we have added on the data which related to participants, and as seen in the bar plot below the participant number 9, 7, 5 and 1 are the participants with the most participation in the activities. While the participants 11, 5, 3, and 13 are the lowest regarding participation in the activities.



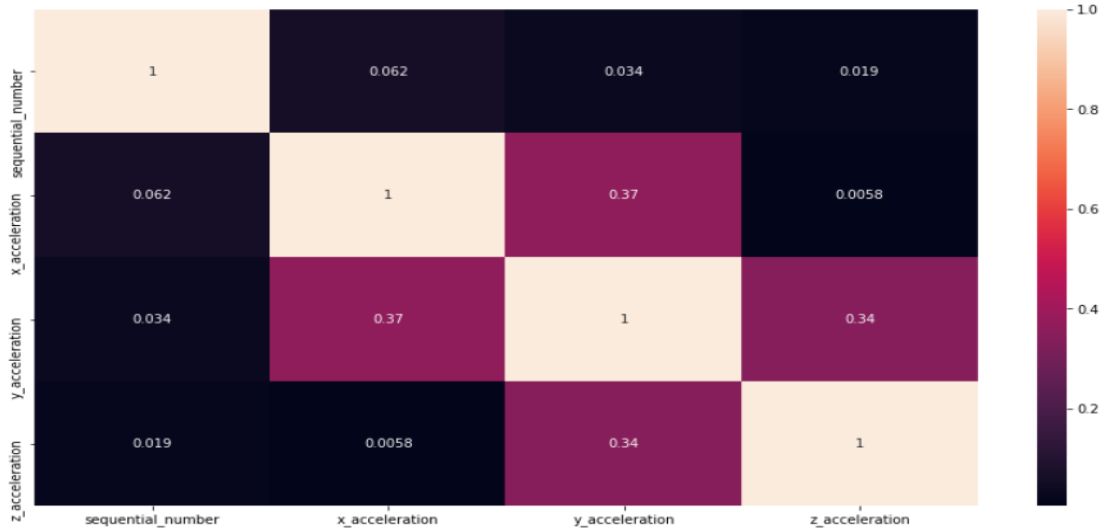
"Figure 6: The count plot of the participtiants"

Second, we have investigated the relationships between all the pairs of the features:

1- We have constructed the correlation matrix that shows how much each pair in the dataset are correlated;

The Hypothesis is that the three accelerations of the three axes; x, y, and z are dependent on each other and there is a somehow strong positive correlation between all the pairs of the three accelerations.

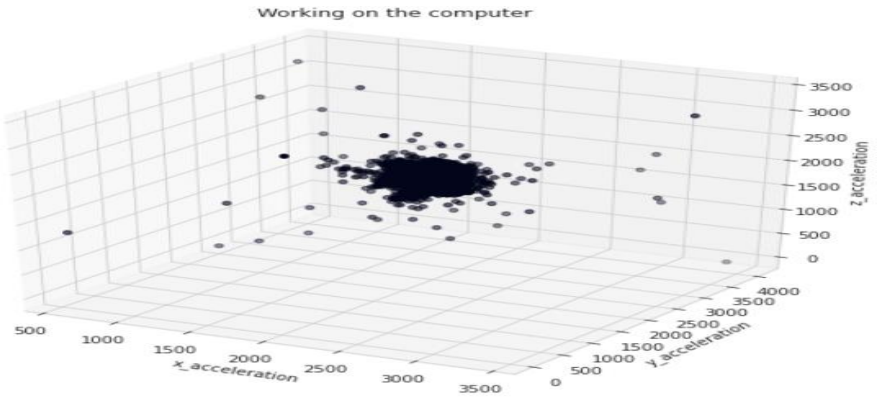
And, as seen in the correlation matrix below the correlation value between x-acceleration and y_acceleration is 0.36 which is below 0.5 that means it is a weak correlation between those two accelerators, while the correlation value between x-acceleration and z-acceleration is 0.018 which is so weak positive correlation while it also less than 0.5, and the correlation between y-accelerators and the x-accelerators is 0.36 which is relatively a weak positive correlation as it also less than 0.5. That means that our hypothesis is invalid as the correlation between the three pairs of the accelerators.



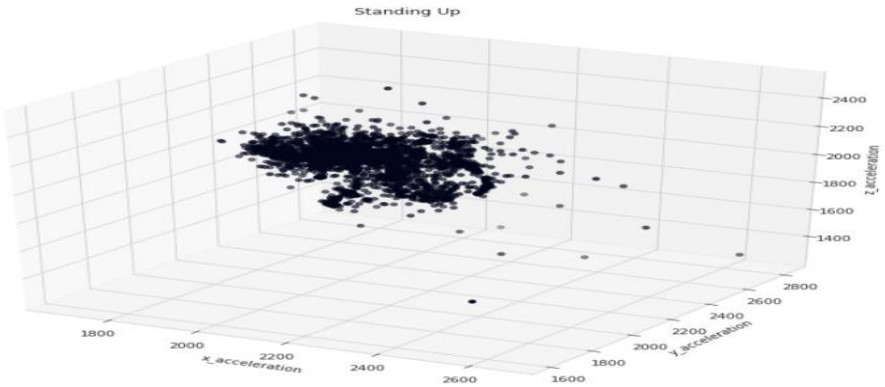
"Figure 7: The correlation matix"

2- We created seven 3D scatter plots to investigate the relationship between the x-acceleration, y-acceleration, and z-acceleration readings for each activity type. We were able to combine the seven plots in one plot but we preferred to make it clearer.

The Hypothesis is while the person working on the computer the x-acceleration and z-acceleration would be so small (<1000) compared with the y-acceleration values as the person while working can move his head vertically consistently. And, as can we see in the below first 3D plot that our hypothesis is a valid hypothesis as the x-acceleration and z-acceleration < 1000 and y-acceleration > 1000 so, we can conclude that our hypothesis is true.

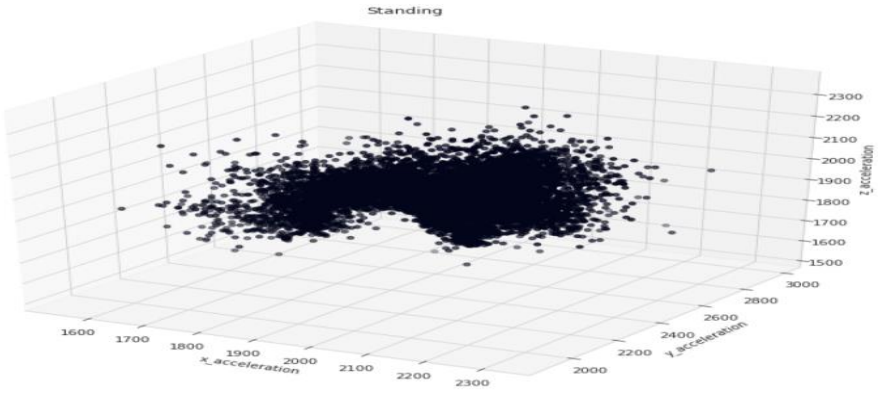


"Figure 8: The 3D scatter plot that show the relationship of the 3-axis accelerations with label working on computer"
The second Hypothesis is that while the person standing up the most readings of y-acceleration and z-acceleration be high compared with the most readings of x-acceleration as the person moves vertically and diagonally but still constant horizontally. And, as seen in the graph below, our hypothesis is a valid hypothesis as the most readings of the x-acceleration < 1000, y-acceleration >= 1000, and z-acceleration > 1500.



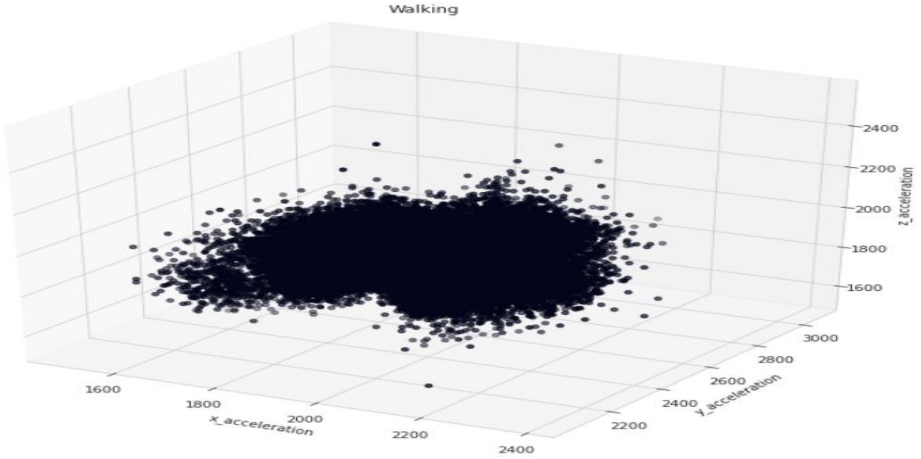
"Figure 9: The 3D scatter plot that show the relationship of the 3-axis accelerations with label sitting up "

The third Hypothesis is that when the person standing the three readings of the accelerations should be small as he doesn't move, vertically, diagonally, or horizontally. But as we can see in the graph below for the standing activity type the most of the readings of the three accelerators show medium values for the three accelerations which indicate the invalidity of our hypothesis.



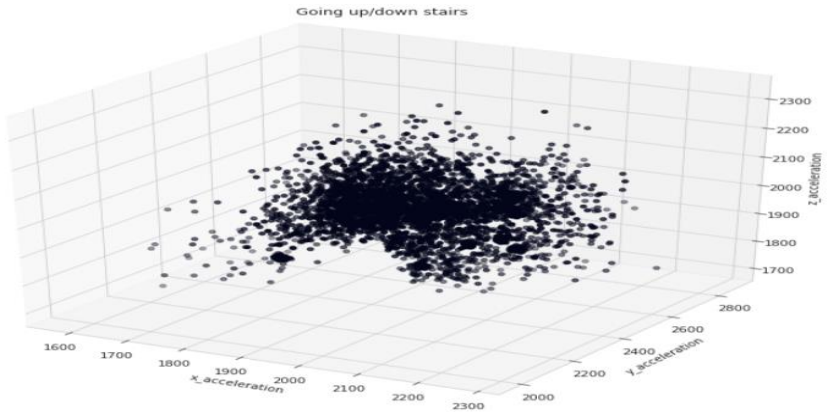
“Figure 10: The 3D scatter plot that show the relationship of the 3-axis accelerations with label standing”

The fourth Hypothesis is that when the person is walking the most of the x-acceleration readings should have a value from medium to high (1000<value<2000) while they-acceleration and the z-accelerations the most values of them should be low as the person walk which means he only move horizontally. But as seen in the below figure the three values of the accelerations have a relatively from medium to high readings (<1000). That indicates our hypothesis this time is not valid.



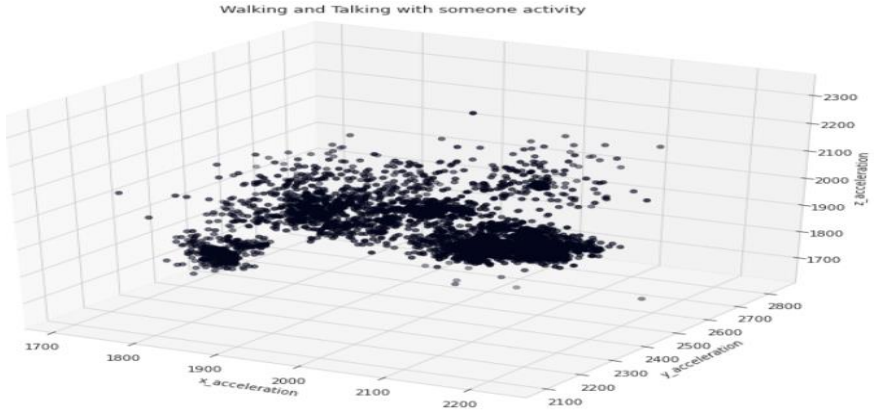
“Figure 11: The 3D scatter plot that show the relationship of the 3-axis accelerations with label walking”

The fifth Hypothesis is that when the person goes up or downstairs the z-accelerations should be high and also the x-accelerations and y-accelerations. And, from the figure below we can notice in most of the readings the three values of the readings are relatively high which means the validity of our hypothesis.



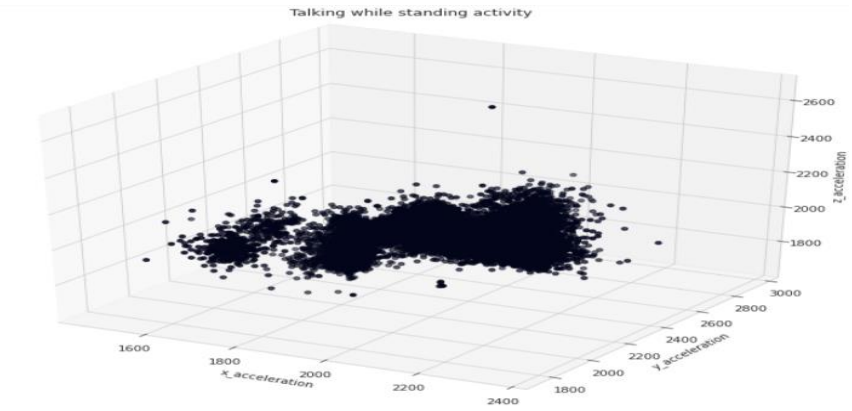
“Figure 12: The 3D scatter plot that show the relationship of the 3-axis accelerations with label going up/down sairs”

The sixth Hypothesis is that when the person walking and talking with someone it would be almost the same condition when the person only walking but he maybe moves vertically also as he is talking with someone so the x-accelerations and y-accelerations should be relatively higher than the z-accelerations. As noticed in the below graph our hypothesis is valid as most of the x-accelerations and y-acceleration readings are relatively high and higher than the z-accelerations values.



“Figure 13: The 3D scatter plot that show the relationship of the 3-axis accelerations with label walking and talking with someone”

The seventh hypothesis is that when the person standing while talking the three readings of the accelerations should be small as he doesn't move, vertically, diagonally, or horizontally. But as we can see in the graph below for the standing activity type the most of the readings of the three accelerators show medium values for the three accelerations which indicate the invalidity of our hypothesis.



“Figure 14: The 3D scatter plot that show the relationship of the 3-axis accelerations with label Talking while standing”

2-modeling

For building a robust model that is able to correctly detect the activity type using only the three accelerations readings; we need to make strict steps for asserting this. So, we need to assure the data in their correct form which means it is 100% free from any errors or problems that can impact our machine learning model as said: "rubbish in rubbish out". So, let's investigate each step done and why we have implemented it:

1- Feature Engineering and reduction: As we have seen in data exploration we have not many features, all we have are 4 features (sequential_number, x-acceleration, y-acceleration, z-acceleration beside the added feature the participants). So, the feature engineering and feature reduction, not a big deal here but as we stated in the project goal we need to build a model or detector that take the values of the accelerations of the three-axis and detect the activity type made by a person based on that so, we need only the three features x-acceleration, y-acceleration, and z-acceleration not anything else. So, we have removed the sequential_number which is only an identifier for each reading from the sensor, and also we have removed the Participant feature as it indicates the person who participated in the experiment which is not needed while we will use the detector on a new person in real-life usage.

2- Cleaning the missing values: As an essential step, we need to ensure that our data is clean from any missings and if there are any missings we should fix it. In our case, our data was clean from missings but once we have returned the label from the codified number to the real label value we have detected that there was a wrong label which is 0 that converted to null after we have mapped the label from the codified numbers (1,2,3,4,etc.) to the real activities (standing, walking, working on the computer, etc.). So, we have removed all these nulls generated from the 0 labels.

3- Balancing the label: This step considered one of the most critical steps in this application as the data really suffer from the imbalance as seen in the figure that working on the computer and talking while standing activities have the majority in the data that could lead to biases in the machine learning model which means that the model gives a higher weight for this major classes that will surely lead in errors on detecting of the minor classes such that going up/down stair, walking, etc. This is known as a generalization problem that means the model fails in generalizing the whole class. For handling this we need to make the data balanced using either oversampling of the minor classes which mean increasing the number of samples of these minor classes or downsampling of the major classes which means decreasing the number of samples in the major classes. We have decided to go with the oversampling

technique especially because we have used only 10% of data so we won't decrease more samples. So, we used a technique called Synthetic Minority Over-sampling Technique (SMOTE) as its name tells the technique used to synthetic samples from the minor classes it takes a sample from the feature space that related to the minor class and find it k-nearest samples and take one vector in between the sample and its k-nearest neighbor and multiply it by a random number between 0 and 1 [1].

4- Scaling the features: As noticed in the exploration of the data the three accelerations x-acceleration, y-acceleration and z-acceleration have different scales and this also can lead to a problem for some of the machine learning algorithms that can consider the features with higher scale have a higher weight so we need to standardize all the features with the same scale by subtracting the mean of the feature from each value in this feature and divide by the standard deviation which makes the values of the whole features range from 0 to 1 [2].

5- Splitting the data: We have split the data into three partitions: The first partition is the training data that would be used for training the machine learning model, the second partition is the validation data which would be used to assess the model's performance once it finishes training we call this the initial testing phase, the final partition is the testing data that used to evaluate the model after finishing the whole process (usually after running of the most promising model).

6- Constructing and training/testing machine learning models: After making all the pre-modeling steps and the assertion of the quality of the data we have started building the machine learning models that would be used in detecting the activity type based on the accelerations values. We have used two of the most powerful and the widely known machine learning models which are XG-Boosting and Randomforest the two models are based on the ensembling methods which mean they combine different machine learning algorithms in deciding the final outcome like the case in Randomforest that uses multiple decision trees and combine them to take a robust decision about the activity type they detected. We have trained the two models on the training data then used the two models in making the prediction on the validation data.

7-Models Evaluation: We have used some metrics to evaluate the performance of the two machine learning models. We have measured the accuracy of the two models which is the proportion of the correctly detected activity types from the whole predicated activity types. And, the second metric is the f1-score which is a harmonic mean of two important metrics which are the precision and the recall.

Results

As mentioned in the methodology we have used two powerful models which are the XG-Boosting and the Randomforest classifiers. Initially, we have used the default hyperparameters in the two models and postponed working on the hyperparameters till finding the winner or the most promising model between the two models to know that let’s investigate the table below that compares between the two models.

Metric	Randomforest	XGB
F1-score	0.68	0.47
Accuracy	0.68	0.48

“Table 1: The modeling results”

As seen in the above table the randomforest totally has beaten the XGB so, the Randomforest model is the most promising model that would be used in the hyperparameter tuning phase to improve its performance.

We have selected 4 essential hyperparameters of the randomforest to be tuned those four hyperparameters are:
1- n_estimators: This is the number of decision trees that would be applied in the Randomforest model. The default value is 100 trees. we will investigate various numbers for this hyperparameter like 10,100, etc.
2- Max_depth: This defines the highest depth each estimator (tree) can approach. The default amount for the Max_depth is None which intimates that each tree doesn't grow till all the nodes in the leaf become related to the corresponding model (pure). different values would be used for it like none, 5, 10, 15, etc.
3- min_samples_split: That defines the least sample required for an inner node to be split. The default value for this hyperparameter is 2 so different values would be used which are 2, 4, 8, 16, etc.
4- min_samples_leaf: The least number of samples required to give the leaf node. The default value for this hyperparameter is 1. different values would be tried like 1, 2, 3, 4, etc [3].

The best hyperparameter values after training the classifiers with those amount of combinations of hyperparameters:
max_depth is Nonemin_samples_leaf is 1
min_samples_split is 2
n_estimators is 100

Then we have used the Grid Search technique that used to train all the predefined combinations of the model to find the best combination of those hyperparameters. So, we have increased the randomforest classifier by almost 0.01 it is somehow a small increase but more combinations could be tried but those would consume a lot of time as all the possible combinations are tried.

Metric	Randomforest before tuning	Randomforest after tuning
F1-score	0.68	0.69
Accuracy	0.68	0.69

“Table 2: The model comparison before and after tuning”

Discussion

As seen in the result section we could reach 0.69 accuracies that surely is not high accuracy but we can say it is somehow decent as the number of the features are small compared with the amount of data. Of course, there is room for improvement of the model if the whole data is used but as we have currently a limited computational process we have selected to work on only 10% of the data which is still representative of all the seven activities. But, this 10% is almost 200,000 data points (samples) so, it is enough for the model to generalize but as we have mentioned the problem seems to be in the number of the features which let the model somehow underfit and not generalize the model well. So, we can say our final model is not robust enough to be published in production but initially, it is good enough for the investigation of the state of the art. As the model correctly detected the activity type of 20% of the data

which are almost 20000 samples of the testing data that means the model was able to correctly detect almost 14000 samples which are decent.

Conclusion

Finally, we could say we have done our goal in building a decent detector that is able to detect the activity type of persons using the accelerator sensors. As mentioned in the discussion above there is still room in the improvement of the model either with finding features like measure also the accelerations from other parts of the person bodies like his hands, legs, etc. and combine these features with the features that we have or use more data points and this the easier option that could be implemented as the data is available but more computational resources are needed. Overall the whole data science process from data cleaning, data exploration, modeling, evaluation, and automation has been implemented and utilized in this research.

References

[1] Blagus, R., Lusa, L. SMOTE for high-dimensional class-imbalanced data. BMC Bioinformatics 14, 106 (2013). https://doi.org/10.1186/1471-2105-14-106

[2] Raschka, Sebastian. About Feature Scaling and Normalization. (2014).https://sebastianraschka.com/Articles/2014_about_feature_scaling.html

[3] SAXENA, SHAROON. A Beginner's Guide to Random Forest Hyperparameter Tuning. (2020). https://www.analyticsvidhya.com/blog/2020/03/beginners-guide-random-forest-hyperparameter-tuning/