Project Report

Activity Monitoring

Shivani Soman, 304946159

Prachi Kumar, 104944750

Maithili Bhide, 104943331

Introduction

Activity Monitoring is the process of predicting a person's activity based on certain gathered information. In most cases, activity monitoring for activities like walking, jogging, sitting, standing, etc. is done with the help of a smartwatch or fitness band. Various examples in the market include Fitbit, Misfit, Apple Watch, etc. These devices exploit the data obtained from various sensors like accelerometer, gyroscope or magnetometer in order to determine the activity a person is performing. In the majority of applications, this information is used for tracking fitness, calories and number of steps walked but can also be used in other domains like healthcare. This information could be used in certain situations where a doctor would like to monitor a patient's activity remotely. It could also be useful to track the activities of older patients or even in certain applications involving fall detection.

In this project, we use sensor information obtained via a smartwatch and build a model that successfully determines the type of user activity. We used information from multiple sensors like accelerometer and gyroscope and also used various different classifiers and classification techniques to determine the best one. We have considered four different activities as follows - Laying Down, Sitting, Standing and Walking. The following sections contain a literature review as well the methodology of our project and the obtained results and analysis.

Literature Review

[1] Activity Recognition in Youth Using Single Accelerometer Placed at Wrist or Ankle - Andrea Mannini , Mary Rosenberger , William L. Haskell, Angelo M. Sabatini and Stephen S. Intille :

This paper from 2016 aimed to test an activity recognition algorithm previously designed for adults on data from youth performing similar activities. They collected data from 20 youth around the age of 13 performing a set of 23 activities classified among 4 general classes. Accelerometers placed at the wrist as well as the ankle were used in the data collection process. The accelerometer data was preprocessed to get the signal magnitude vector from the three axes of the accelerometer and the data was divided into 12.8 secs of non-overlapping windows. The classifier used was a Support Vector Machine with a radial basis function kernel.

They performed four different types of experiments in order to evaluate the model performance. The experiments included the following steps - Testing the algorithm originally developed for adults on youth data, extending the feature set and performing LOSO cross-validation separately on adult and youth data, testing the algorithm with both datasets combined, etc. They observed that in addition to features related to basic signal structure (mean, standard deviation, acceleration range), activity fragmentation features appeared to be an important source of information. Overall, their results showed that activity recognition algorithm used previously for adults could also be used in the case of youth data. When validating methods with a LOSO approach starting from a merged dataset, it was confirmed that the accuracy can be improved by using data from both groups, even if the tested subject was not included in the training set.

[2] Machine Learning Methods for Classifying Human Physical Activity from On-Body Accelerometers - Andrea Mannini and Angelo Maria Sabatini :

This paper focuses on the different computational algorithms dedicated to classifying human physical activity using accelerometers placed on the body. They also focus on special classifiers based on Hidden Markov Models (HMM). They discuss the most commonly used methods for automatic classification of human activity with special emphasis on Markov modelling. The authors also exploit an annotated dataset of signals from on-body accelerometers in order to test several classification algorithms, including HMMs with supervised learning.

They derived features from the coefficients of Short Time Frequency Transform (STFT) to calculate the frequency domain entropy. They also consider other features like the correlation coefficients between each pair of accelerometer signals. For feature selection they perform Principal Component Analysis (PCA) followed by a data whitening transformation. They also compare various supervised and unsupervised classification approaches as well as provide extensive background oh Hidden Markov Models.

They propose to build a sequential classifier composed of a Gaussian cHMM and provide details of a validation study on it. Acceleration data, sampled at 76.25 Hz, was acquired from five bi-axial accelerometers, located at the hip, wrist, arm, ankle, and thigh. The original protocol was based on testing 20 subjects, who were requested to perform 20 activities but 13 subjects were randomly selected for further analysis, in order to ease the development work. They achieved relatively high accuracy for Gaussian MM and cHMM based sequential classifier with values of 92.2% accuracy and 95% accuracy. A major contribution of the paper lies in pursuing a Markov modelling approach to the design of human physical activity classification.

[3] Estimation of Physical Activity Energy Expenditure during Free-Living from Wrist Accelerometry in UK Adults, Tom White, Kate Westgate, Nicholas J. Wareham, Soren Brage

They show that there is a strong relationship between wrist acceleration and physical activity energy expenditure. Physical activity causes an increase in energy expenditure, and this can be used to describe the behavioral profile (in terms of activity levels) of a person. They built predictive models of physical activity measures from wrist accelerometers, using acceleration of the torso and physical activity energy expenditure.

1695 people wore two devices continuously for 6 days - a combined heart rate and movement sensor, which measured heart rate and acceleration of the torso in 15 second intervals, and a wrist accelerometer worn on the non-dominant wrist. People were also asked to run on a treadmill to to get their individual heart rate response, which helped get calibration parameters for a PAEE(Physical activity energy expenditure) equation. The acceleration in the three directions was used to get the vector magnitude per sample (square root of the sum of squares of readings on all three axes). They used multilevel linear regression models to predict PAEE and torso acceleration from wrist acceleration. They used 60% of the people (their data) for training, and 40% of the people (their data) for testing. They saw that although their models used only the magnitude of wrist acceleration for prediction, they still got very good results.

[4] Estimating Activity and Sedentary Behavior From an Accelerometer on the Hip or Wrist
Mary E. Rosenberger, William L. Haskell, Fahd Albinali, Selene Mota, Jason Nawyn, and Stephen Intille

This paper talks about comparing the effectiveness of physical activity measurements based on accelerometers on the hip versus the wrist, and using these accelerometers to classify whether a person in sedentary, walking, cycling or doing work like folding clothes. Previously an accelerometer worn on the hip was used to measure physical activity, but now one on the wrist is becoming more popular.

The paper talks about a study they conducted in which 37 adults in the ages of 18 to 74 wore accelerometers on the hip and wrist, and a portable metabolic unit to measure energy expenditure. They then performed 20 different activities - both sedentary and physical. The activities were of four types - sedentary (like laying down, sitting, etc.), cycling, ambulation (walking) and lifestyle (folding laundry, sweeping floor, etc.). Their accelerometer measured at a rate of 90 Hz. The raw accelerometer data was converted to a motion summary count using the area under the accelerometer curve. They then did statistical analysis to determine the ability of the accelerometer based data to be able to predict the activity type (like sedentary, ambulation, lifestyle or cycling) and classify activity intensity. All models used data for a class obtained during the steady state - so there were no transitions between walking and sitting, etc. They saw that an accelerometer on the hip performs better than an accelerometer on the wrist for classifying these activities.
The paper also talks about the challenges of using an accelerometer on the wrist - for example, if walking or jogging and the arms are swinging freely, the data would characterize that activity quite well. However, if the wrist does not have free movement during those activities (like holding a cup of coffee while walking), then the accelerometer data will have very different characteristics. They suggest the need for adding multiple features to be able to handle these type of challenges.

[5] Computational Methods for Estimating Energy Expenditure in Human Physical Activities - Shaopeng Liu, Robert Gao and Patty Freedson :

This paper focuses on three interrelated areas in the estimation of Physical Activity Energy Expenditure (PAEE) 1) types of sensors worn by human subjects, 2) features extracted from the measured sensor signals, and 3) modeling techniques to estimate the PAEE using these features. Physical activity is critically linked to overall health, lowering risk of cardiovascular disease, weight control, diabetes, obesity and many other diseases making the estimation of physical activity a critical task to targeted health improvement. In this paper, 2 types of body worn sensors were used - 1) Accelerometer and foot pressure sensors 2) Physiological sensors.

The raw signals obtained from body worn sensors contain many data points and were sampled to create time domain features, frequency domain features and demographic and anthropometric features in the feature generation step. Linear regressions, non-linear regressions and hybrid approaches were used for modeling PA Energy Expenditure. This paper presented an overview of recent advances in the estimation of PAEE and concluded by delineating the shift in research to the three areas of sensors, features and methodology.

[6] Orientation sensing for gesture-based interaction with smart artifacts - Alois Ferscha, Stefan Resmerita, Clemens Holzmann, Martin Reichor :

This paper focuses on orientation sensing and gesture recognition. The authors of this paper focused on 3 types of artifacts subject to manipulation by the user - 1) hand worn 2) hand carried and 3) hand graspable real world objects. The proposed framework specification is independent of sensor technology and classification methods, and elaborates an application-independent set of gestures. Orientation information is very important for gesture detection and this paper describes a method of using Euler's angles to generate an orientation matrix.

The authors are considering a large set of application independent gestures corresponding to the 3 types of artifacts under consideration in the gesture library since a fairly large set of gestures can be employed for controlling fairly large classes of applications. A framework for gesture recognition is developed using the use case of a smartphone. This framework consists of sensor, classifier, trained and composition modules. The central idea behind the framework proposed in this paper is application-independent gesture recognition using orientation data.

Our Takeaways from the Literature Survey

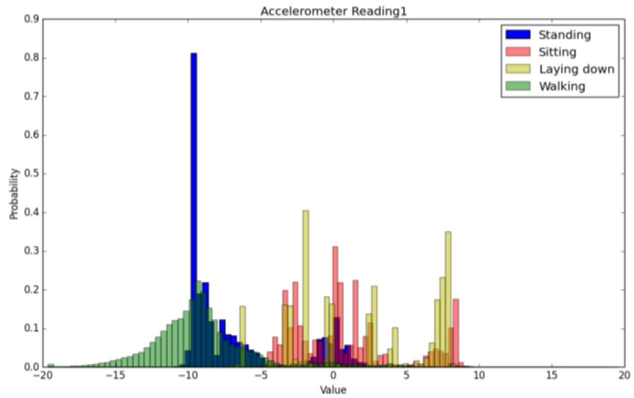
Overall, based on reading these papers, we realized that accelerometer sensor data is very important for our task of classifying different activities. We also decided to experiment by adding more sensor data like orientation and gyroscope to complement the accelerometer readings.

We also realized that there may be a difference in readings of these sensors depending on whether we wear the smartwatch on our dominant or non-dominant hand. Wearing the smart watch on the dominant hand could lead to more hand movements during any activity. For instance, while walking, we may point to something with our dominant hand. Another example may be if we are drinking coffee while sitting. People would most likely hold the coffee mug with their dominant hand to drink from it. In our data collection, we have worn the smart-watch on our non-dominant hand to minimize the unnecessary movements (not related to the activity which we are doing) which would occur more frequently with the dominant hand.

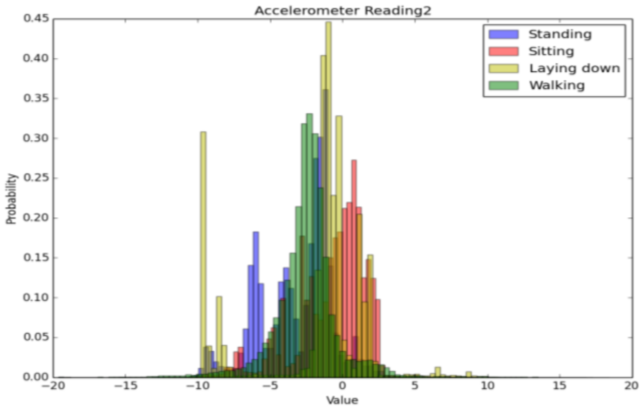
Collected Data

We plotted our collected data sensor readings for different class labels to get a better picture of the data distribution. The plots for accelerometer readings x, y and z are shown. For each figure, the x-axis shows the value of the sensor reading, and the y-axis shows the probability of being at that value. Different classes - standing, sitting, laying down and walking are shown in colors.

The figure to the right show the distribution of the data for the x-reading of the accelerometer. We can see that for the standing class (blue in the figure), the data has a very high probability of having acceleration -10 m/s2, as we may have moved our arms suddenly from at rest by our sides to check the time or fold our arms. Sitting and laying down data shows a quite high probability of having acceleration close to zero.

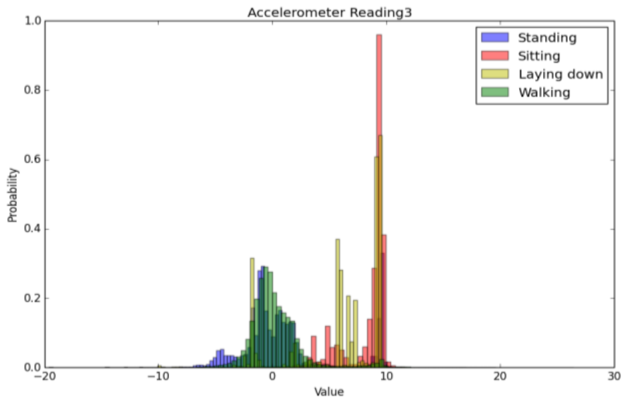


In the figure to the right, we can see the distribution of data for the y-reading of the accelerometer. We can see that laying down and sitting again have a very high probability of having zero acceleration in the y direction.



We also see that the overlap of the different classes for values of y accelerometer readings is quite high, which suggests the need for more sensor inputs or features to be able to distinguish between classes.

In the figure to the right, we now see the data distribution for the z-reading of the accelerometer. We see that sitting data has a very high probability of having acceleration 10 m/s2 (eg. because of sudden movements from rest to keeping the hand under the chin). We also see that walking has a good chance of having zero z-acceleration, which makes sense as our arms swing with quite a uniform speed by our sides when we walk.



Sensors that We Used

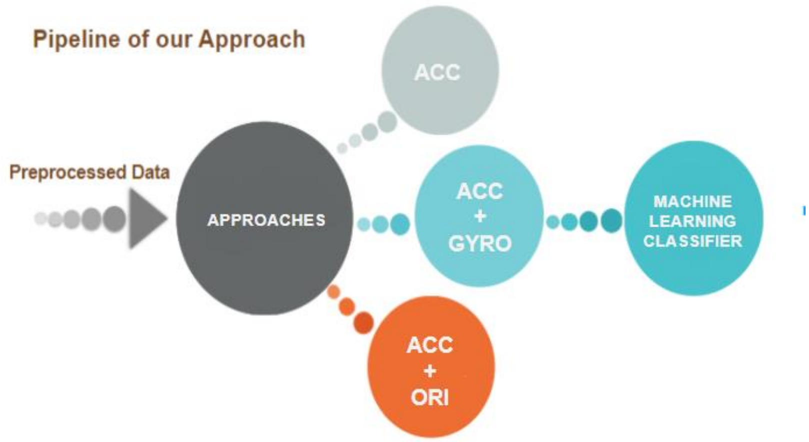
The android sensor readings that were available were accelerometer, gyroscope, magnetic field, orientation, light, gravity, linear acceleration, rotation vector and step count. Based on the papers that we read and our understanding of these sensors, we decided to use the accelerometer, gyroscope and orientation sensor readings. By using the accelerometer sensor, we have included the influence of gravity as well, because accelerometer readings include the force of gravity. Magnetic field sensors (which are used mainly in applications like a compass) and light sensors were seen as unnecessary for the task of activity monitoring.

Pipeline of Approach and Implementation

Our project is divided into two separate phases - one where we split the dataset into sections of 10 ms and another where we split it into sections of 4 secs. The following sections explain our methodology in the two different phases.

Phase 1

In phase 1 of our project, we did not keep our training and testing data separate (in terms of training data and testing data belonging to different individuals). We merged our entire dataset together and then split it into training and testing sets. We considered three different sensor readings - accelerometer, gyroscope and orientation readings. The pipeline of our approach is shown in the diagram below.



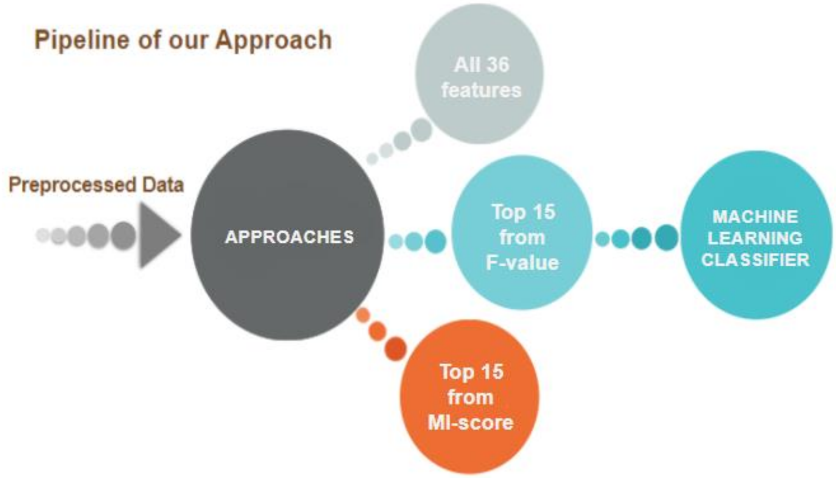
We first preprocessed the data into sections of 10 ms. The preprocessing would be explained in detail in the following section. We then used the preprocessed dataset and performed activity monitoring classification on it using features from three different set of sensor readings -

1. Accelerometer only
2. Accelerometer and Gyroscope
3. Accelerometer and Orientation

We then took each of these sets of readings and performed various classification techniques on it to find the best one. We did this in order to compare the different readings obtained and to find the ones that provide the most information.

Phase 2

We realised that our time window for the first phase was too short and hence increased it to 4 secs. In this section, we kept the training and testing data completely separate. The testing set was based on data from users which were not included in the training. We also extracted 36 features from the sensor readings of accelerometer, gyroscope and orientation data. The details of the preprocessing and feature extraction are provided in the following section. The pipeline of phase 2 was as follows -



For phase 2 of our project we also considered three different sets of features -

1. All 36 features
2. Top 15 features based on F-value
3. Top 15 features based on Mutual Information Score

We first considered all the 36 features extracted in order to train our models, then performed feature selection in order to find the top 15 features. The metrics we used were the ANOVA

(Analysis of Variance) F-value and Mutual Information Score. The top features obtained from these two metrics were then used to train classifiers.

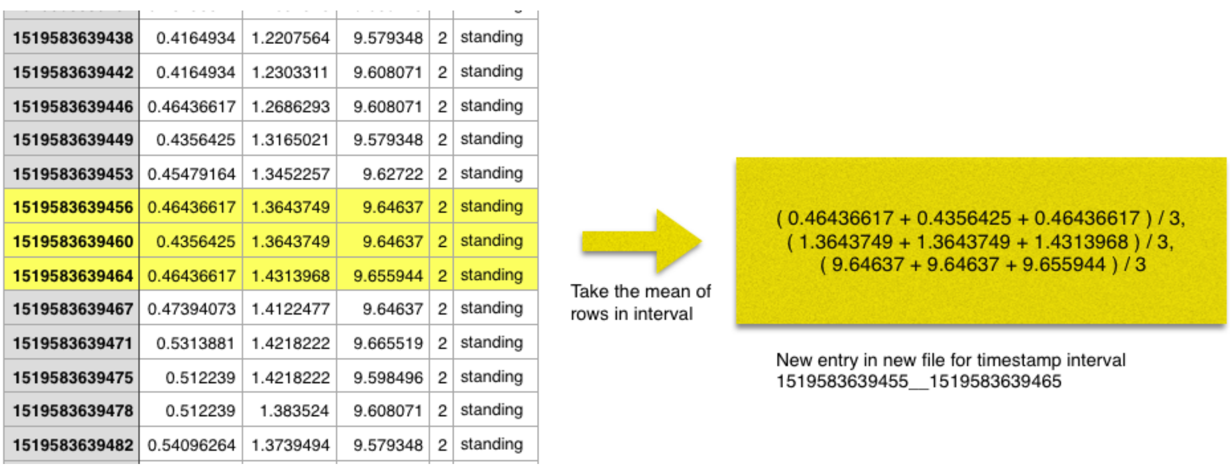
In both phases, we coded in Python and used the Python Scikit-Learn library along with Pandas, Numpy and TensorFlow. For each model, we chose the parameters that gave the best results, and those results have been reported. For instance, in the Random Forest Classifier, we changed the maximum tree depth parameter across different values (10, 50, 100, unlimited) to get the best possible results.

Data Preprocessing

For data preprocessing, we took groups of readings at intervals of 't' seconds (In Phase 1, 't' is 10 milliseconds, and in Phase 2, 't' is 4 seconds). These readings were taken from the accelerometer sensor, gyroscope sensor and orientation sensor data. As these sensors measurements occur at different frequencies, we compared the timestamp intervals across sensors and merged the readings at a common timestamp interval to form a single new reading. The features that were extracted differ in Phase 1 and Phase 2, as described below.

Phase 1 - Window Size = 10 milliseconds

In Phase 1, we only used the mean of the data from sensors in the timestamp window size of 10 milliseconds. For example, as the figure below shows, we took the mean of the accelerometer x, y and z readings in each interval of 10 ms.



1519583639438	0.4164934	1.2207564	9.579348	2	standing
1519583639442	0.4164934	1.2303311	9.608071	2	standing
1519583639446	0.46436617	1.2686293	9.608071	2	standing
1519583639449	0.4356425	1.3165021	9.579348	2	standing
1519583639453	0.45479164	1.3452257	9.62722	2	standing
1519583639456	0.46436617	1.3643749	9.64637	2	standing
1519583639460	0.4356425	1.3643749	9.64637	2	standing
1519583639464	0.46436617	1.4313968	9.655944	2	standing
1519583639467	0.47394073	1.4122477	9.64637	2	standing
1519583639471	0.5313881	1.4218222	9.665519	2	standing
1519583639475	0.512239	1.4218222	9.598496	2	standing
1519583639478	0.512239	1.383524	9.608071	2	standing
1519583639482	0.54096264	1.3739494	9.579348	2	standing

Features Used:

Case 1: Only Accelerometer Sensor Data - We used the x_mean, y_mean, z_mean of the accelerometer data. This gave us 3 total features.

Case 2: Accelerometer + Gyroscope Sensor Data - We used the x_mean, y_mean, z_mean of the accelerometer and gyroscope data. This gave us 6 total features.

Case 3: Accelerometer + Orientation Sensor Data - We used the x_mean, y_mean and z_mean of the accelerometer and orientation data. This gave 6 total features.

Phase 2 - Window Size = 4 seconds

In this phase, we extracted more features from the data. For all the readings in a window size of 4 seconds, we found the mean of the x, y and z readings, the standard deviation of the x, y and z readings, the maximum x value, the maximum y value, the maximum z value, the minimum x value, the minimum y value and the minimum z value. We extracted these features for accelerometer, gyroscope and orientation sensor data.

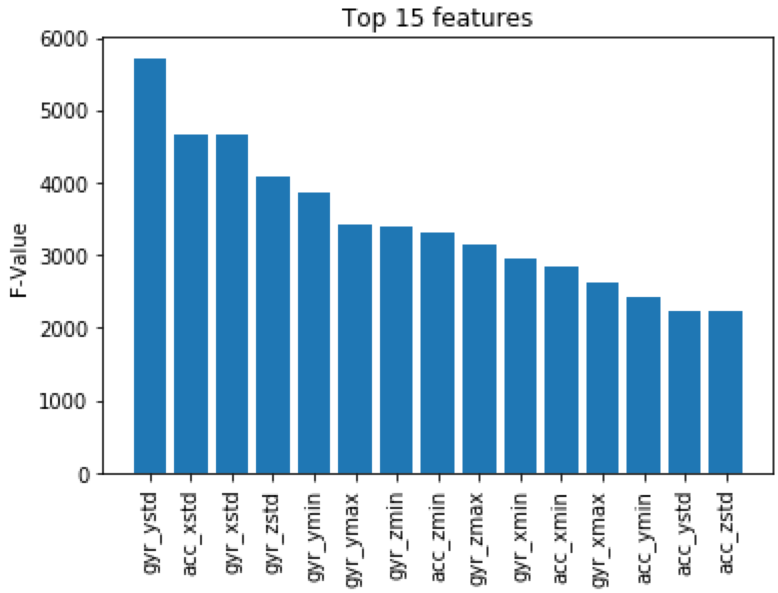
Features Used :

Case 1: We used all features from the 3 sensor readings - x_mean, y_mean, z_mean, x_std_dev, y_std_dev, z_std_dev, x_max, y_max, z_max, x_min, y_min, and z_min for accelerometer sensor data, gyroscope sensor data and orientation sensor data. This gave us 36 features for every instance.

Case 2: We performed feature selection to choose the top 15 best features from our list of 36 features for accelerometer sensor data, gyroscope sensor data and orientation sensor data. Training our classifiers on this subset of best features increased our model accuracy.

Top 15 features using F-value -

Analysis of variance (ANOVA) uses F-tests to statistically assess the equality of means when you have three or more groups, in this case we have 4 - sitting, standing, walking and laying down. F-test is a way of comparing the significance of the improvement of a model, with respect to the addition of new variables. This test helps to capture linear dependency between features. In this first method of selection, we use an F-statistic, i.e the ratio of the between-group variance to within-group variance to identify important features from the point of view of classification. Based on our implementation, after selecting Kbest (K=15) features, we got the following features with the highest F-value. We trained our classifiers using just these 15 most significant features instead of all 36 and observed better accuracy.

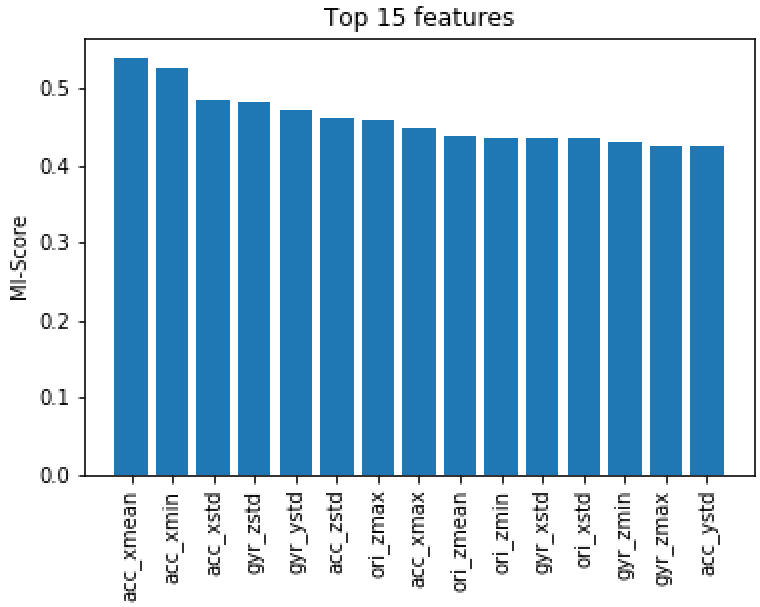


Case 3:

Top 15 features using MI-Score -

Mutual information between two random variables is a non-negative value, which measures the dependency between the variables. It is equal to zero if and only if two random variables are independent, and higher values mean higher dependency. In the first method of feature selection using F-test, we captured the linear dependency between two random variables. Mutual information methods can capture any kind of statistical dependency, but being nonparametric, they require more samples for accurate estimation. After selecting the top 15 features with the highest mutual information, we trained our classifiers using just these 15 features instead of all 36 and observed improved accuracy.

By identifying the top 15 features with the best F-value and MI-Score we were able to see that although the rankings of the features varied with F-value and MI-Score, there was a significant overlap in the features that these methods returned. Only two of our features were common in Phase 1 and Phase 2 - acc_xmean and ori_zmean. This indicated that our feature extraction in Phase 2 was more robust than in Phase 1.


Top 15 features
MI-Score
0.0
0.1
0.2
0.3
0.4
0.5
acc_xmean
acc_xmin
acc_xstd
gyr_zstd
gyr_ystd
acc_zstd
ori_zmax
acc_xmax
ori_zmean
ori_zmin
gyr_xstd
ori_xstd
gyr_zmin
gyr_zmax
acc_ystd

Results and Discussion - Phase 1

We trained on 80% of our data, and tested on 20% of our data (unseen). The results are shown below for the different models that we tried out, with their accuracy and F1-score. Three different cases are shown - only training on accelerometer data, training on accelerometer and gyroscope data, and training on accelerometer and orientation data. The F1-score reported is the average F-score over all four classes.

Classifier	**Accelerometer**		**Accelerometer, Gyroscope**		**Accelerometer, Orientation**	
	Accuracy	F1-Score	Accuracy	F1-Score	Accuracy	F1-Score
Random Forest	0.9377	0.9376	0.9702	0.9703	0.9944	0.9944
Naive Bayes	0.5623	0.5165	0.6819	0.6852	0.6716	0.6682
Decision Tree (CART)	0.9203	0.9203	0.9573	0.9573	0.9900	0.9900
Logistic Regression	0.5572	0.5487	0.5542	0.5449	0.7597	0.7512
SVM	0.9251	0.9235	0.9389	0.9394	0.9965	0.9965
KNN	0.9445	0.9426	0.9759	0.9755	0.9944	0.9938
Neural Network	0.4967	0.4230	0.6668	0.6513	0.6079	0.5764
Neural Network (scaled data)	0.6345	0.6212	0.7364	0.7357	0.8760	0.8690v
Gradient Tree Boosting	0.9224	0.9223	0.9402	0.9401	0.9794	0.9794
LSTM	0.9142	0.9159	0.8755	0.8800	0.9974	0.9974

In the results from this phase, we can see that for the only accelerometer sensor data case and the accelerometer and gyroscope sensor data case, the best performance is seen in K-nearest neighbor (where k = 5). For the case with accelerometer and orientation sensor data, we see that LSTM works really well, giving an accuracy of 99.74%. (These are the cases highlighted in yellow above)

Other observations like seeing the accuracy and F1-score increase greatly after adding the orientation data in training suggest that orientation data might be very important to activity prediction. The orientation sensor measures the degrees of rotation that the device makes along the 3 axes.
We can also see that the neural network performance improves a lot after scaling the data so that it has 0 mean and unit variance.

Seeing such a high accuracy (like 99%) for many models made us wonder whether the models had become overfit to our data specifically, and it turns out that they had. We asked another group for some of their data for testing purposes, and when we tested our models on their data, we saw that the accuracy dropped significantly. For instance, for the Random Forest Classifier, the accuracy dropped from 99.44% (tested on our data) to 41% (tested on another group's data). This told us that the models were overfitted to our data; to the specific way we walked, sat, stood and laid down. We tried to generalize the models a little better by changing the parameters to reduce overfitting. For instance, in our Random Forest Classifier, we changed the maximum depth of each tree to 10 instead of 50 and saw some slight improvement in the results on the other group's data; the accuracy increased slightly to 45%.

The main reason for this was that when we trained our models, we merged all of our collected data together and shuffled it. This led us to our next approach (Phase 2), where we trained on only 2 persons data and tested on the 3rd person's data. We also increased the window size to 4 seconds in the next approach to get a wider interval in which to make an assessment of a person's activity, and also increased the feature extraction.

Results and Discussion - Phase 2

In this phase we ensured that our training and test data was completely separate, i.e. training and test data belonged to different sets of individuals. The results shown below are for the 3 cases we tested - using all features, using best 15 features based on F-value and using best 15 features based on Mutual Information score. The F1-score reported is the average F-score over all four classes.

Classifier	**All Features**		**Top features with best F-value**		**Top features with best Mutual Information score**	
	Accuracy	F1-Score	Accuracy	F1-Score	Accuracy	F1-Score
Random Forest	0.6981	0.7135	0.7648	0.7771	0.7248	0.7400
Decision Tree (CART)	0.6149	0.6268	0.6889	0.7001	0.6124	0.6299
Logistic Regression	0.6057	0.6323	0.5929	0.6222	0.5944	0.6055
LSTM	0.7084	0.7227	0.7497	0.7679	0.7698	0.7815
Gradient Tree Boosting	0.6386	0.6529	0.6750	0.6872	0.6406	0.6527
Naive Bayes	0.6124	0.6012	0.4589	0.4018	0.6036	0.5799
SVM	0.2135	0.1088	0.6894	0.6992	0.6298	0.6298
KNN	0.5569	0.5518	0.6226	0.6374	0.5939	0.6048
Neural Network	0.2643	0.1060	0.6042	0.6114	0.6180	0.6245
Neural Network (scaled data)	0.5893	0.5842	0.6334	0.6163	0.4748	0.4392

In this phase we have used subsets of features from all three sensors in the 3 cases that we tried. Although the accuracy values in this phase are lower than the first phase, our classifiers are more robust to generalizations as we have ensured that the training and test data comes from completely different individuals.

Most classifiers were giving us an accuracy of about 60% when using all 36 features except SVM and Neural Network. This was because in the case of SVM the classifier was predicting only 3 classes - standing, sitting and walking while the Neural Network was only predicting 2 classes - sitting and walking when using all 36 features. This issue was resolved when we trained our classifiers using the top 15 features according to F-value and MI-Score. This made us realize that using too many features may not always be a good idea.

We can also see that the performance of the models improves slightly when we go from all 36 features to only the best 15 features. For instance, the accuracy of the LSTM jumps from 0.70 to 0.74 for the 15 features based on F-value, and to 0.76 for the 15 features based on mutual information score. Similarly, the Random Forest Classifier, the Decision Tree, Gradient Tree Boosting, SVM, KNN and the Neural Network show an improvement in their performance when we move to the best 15 features instead of using all 36 features. This suggests that some of the features may be irrelevant and too many features may be causing the model to become slightly more overfitted on our data thereby having a slightly lower accuracy on the unseen test data.

From the results we can see that our LSTM model was the most effective at correctly classifying the data. This is because the data that we are using is a sequential time series data. The LSTM model uses a Recurrent Neural Network and this helps to capture the temporal information in the data in addition to the spatial information making it more efficient than the simple Neural Network and even other models.

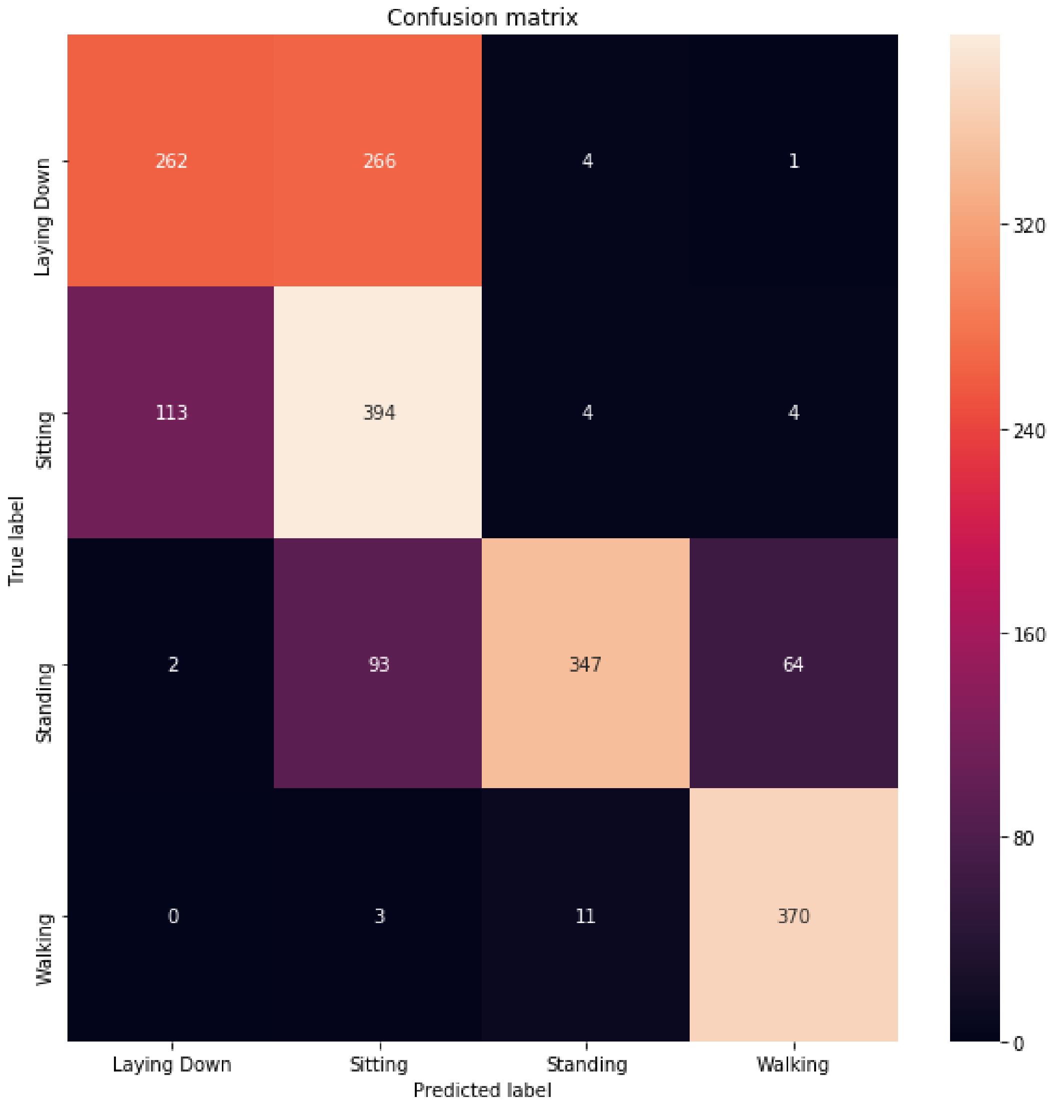
We can also see that the Random Forest classifier with 100 estimators and no maximum depth specified outperforms the two-layer LSTM trained on 20 epochs when we use the top 15 features according to the F-value.

Confusion Matrix

Below we have shown the confusion matrices obtained for the best performing model under case 1 (all 15 features), case 2 (best 15 features using F-value) and case 3 (best 15 features using MI score).

Case 1 : Training on All Features :

Best Performing Model : LSTM

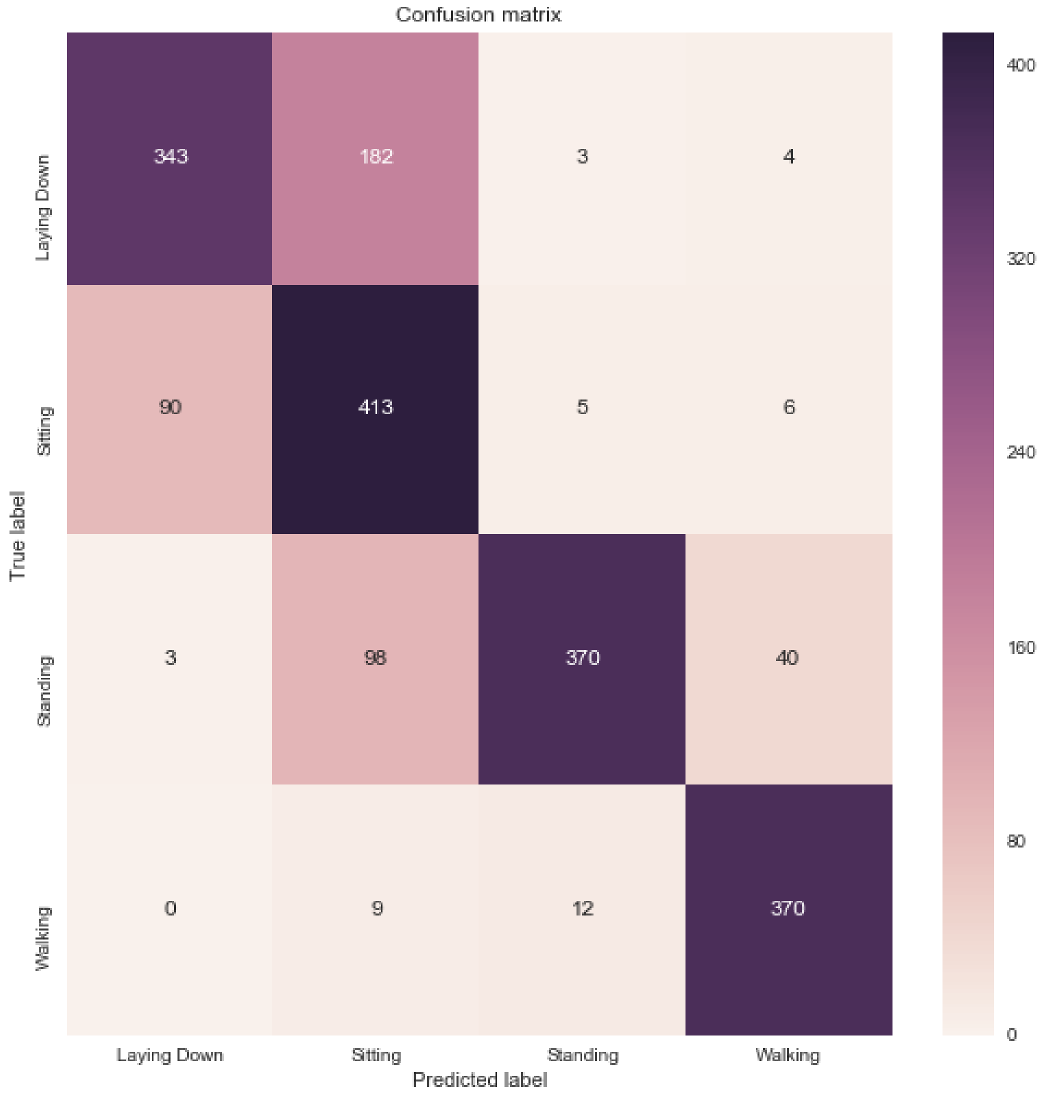


The above figure and its accompanying scale on the right shows that most of the classes were correctly predicted giving us a diagonal confusion matrix. We can also see here that some of the sitting examples were confused with laying down and conversely some laying down examples were confused with sitting when we used all 36 features.

This is consistent with our observations from the SVM and Neural Network results when using all 36 features. In the next confusion matrix we can see that this issue was resolved when we used a subset of the features instead of training our classifiers on all 36 features.

Case 2: Training on Top 15 Features (using F-value)

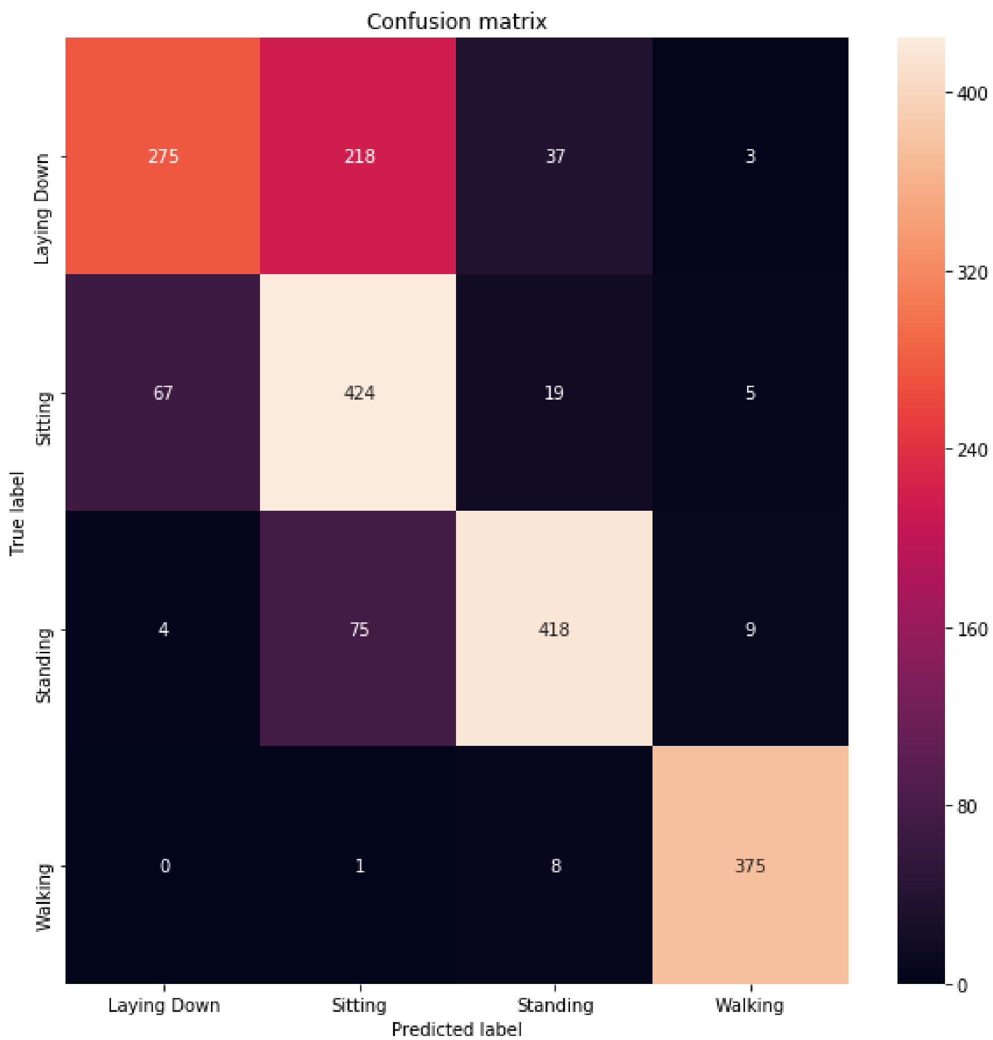
Best Performing Model : Random Forest Classifier



The above confusion matrix and its accompanying scale on the right shows a much better performance when we used the top 15 features using F-value for training our classifier. We have obtained a much more pronounced diagonal matrix in this case. Additionally, the confusion between the laying down and sitting classes has also decreased.

Case 3 : Training on Top 15 features (using MI-Score)

Best Performing Model : LSTM



This confusion matrix and its accompanying scale also show a similarly diagonal matrix as the ones we've seen in the first two cases. Comparing this LSTM confusion matrix with the LSTM confusion matrix in case 1 when we were using all 36 features, we can see decreased confusion between laying down and sitting examples when we used a subset of 15 features with the highest mutual information score. The number of correctly classified instances for each of the categories - standing, sitting, laying down and walking have gone up.

Conclusion

Overall, in this project we got a great insight into machine learning used in the health domain. We got a hands-on experience of working on machine learning right from the data collection stage (otherwise usually in an academic environment, the dataset is usually available or provided). We preprocessed this data and got it ready for model training, and tried out many different models to try to see the best performing one. We used two approaches (described as Phase 1 and Phase 2 above). Overall, in this project, we got to learn a lot. We realized the issues with our first approach and worked on the second approach to see better generalization of the models.

We also saw some challenges in using sensors on the wrist like a lot of wrist movement regardless of the current activity which can make it difficult to characterize a particular activity well with the data. Based on one of the papers we read, hip accelerometry may show a slight improvement in this respect. That may be an interesting future work of the project, to use some sensors on the hip (maybe using some kind of smart hip based pedometer/device which you can attach to your pant / belt / pocket) to measure the activity of a person. The only issue with a hip based sensor may come in the laying down stage when it may become uncomfortable for a person to lay down with a device on their hip.

Overall, we really enjoyed working on this project, and got to learn a lot.