DETECTION OF POSTURAL TRANSITIONS USING MACHINE LEARNING

GARIMELLA KRISHNA SANDILYA – 590100 044010151 6

NAME OF HOME INSTITUTION	AMRITA VISHWA VIDYAPEETHAM
NAME OF HOST INSTITUITION	UPC-ETSEIB
DEGREE	B.TECH ELECTRICAL AND ELECTRONIC ENGINEERING
TYPE OF DOCUMENT	STUDENT EXCHANGE – RESEARCH ASSIGNMENT
NAME	KRISHNA SANDILYA GARIMELLA
GUIDE	PROF. ANDREU CATALA MALLOFRE
DATE OF SUBMISSION	03-07-2019

Abstract

The purpose of this project is to study the nature of human activity recognition and prepare a dataset from volunteers doing various activities which can be used for constructing the various parts of a machine learning model which is used to identify each volunteers posture transitions accurately.

This report presents the problem definition, equipment used, previous work in this area of human activity recognition and the resolution of the problem along with results.

Also this report sheds light on the process and the steps taken to undertake this endeavour of human activity recognition such as building of a dataset, pre-processing the data by applying filters and various windowing length techniques, splitting the data into training and testing data, performance of feature selection and feature extraction and finally selecting the model for training and testing which provides maximum accuracy and least misclassification rates.

The tools used for this project includes a laptop equipped with MATLAB and EXCEL and MEDIA PLAYER CLASSIC respectively which have been used for data processing, model training and feature selection and Labelling respectively.

The data has been collected using an Inertial Measurement Unit contains 3 tri-axial Accelerometers, 1 Gyroscope, 1 Magnetometer and 1 Pressure sensor. For this project only the Accelerometers, Gyroscope and the Pressure sensor is used. The sensor is made by the members of the lab named 'The Technical Research Centre for Dependency Care and Autonomous Living (CETpD) at the UPC-ETSEIB campus.

The results obtained have been satisfactory, and the objectives set have been fulfilled. There is room for possible improvements through expanding the scope of the project such as detection of chronic disorders or providing posture based statistics to the end user or even just achieving a higher rate of sensitivity of transitions of posture by using better features and increasing the dataset size by increasing the number of volunteers.

Table of Contents

Table of Figures

List of Abbreviations :

1. CETpD : The Technical Research Centre for Dependency Care and Autonomous Living

2. SiSt : Sit to Stand Transition

3. StSi : Stand to Sit Transition

4.SVM : Support Vector Machine

5. GI : Gini Index

6. IG : Information Gain

1. Preface

Below mentioned are the reasons for pursuing this project as a part of the bachelor level final project and its developments and motivations along with personal interests that have led into the deep pursuit of knowledge into this topic. Also this section enumerates the objectives and motivations of this bachelor's project.

1.1 Origin

The idea of pursuing this project arose from the desire to understand working of sensors placed in/on the human body and to gain deeper insight into the data processing and machine learning techniques that can be used to identify various conditions associated with the human body. This project was proposed by Professor Andreu Catala Mallofre as a part of final year bachelors project over the necessity of a greater purpose which is using postural transition data to eventually detect Parkinson's disease onset.

1.2. Motivation

The motivation to pursue this project arises from personal interest in exploring the various machine learning algorithms and signal processing tools and the whole machine learning pipeline in general to achieve the required outputs in the field of human body data processing. It is to be noted that a properly functioning postural transition detection technique can be beneficial to patients by detecting their frailty which is the greater purpose of this project and also a motivating factor. Also the need enforce such a system on a wearable device which is low cost and yet provides effective monitoring is a motivation.

1.3. Previous requirements

The software knowledge requirements in order to work for this project is the knowledge of the scientific software MATLAB and data processing MS-EXCEL respectively. Apart from this the knowledge base of sensors including accelerometer, gyroscopes and pressure sensors would be beneficial.

Basic machine learning concepts such as weights, cost function, sensitivity, specificity and methods to improve accuracy would also be of great help.

Also, basic classification algorithmic knowledge such as Logistic Regression, Trees, Support Vector Machines for classification and K-nearest neighbours based ReleiFF algorithms for feature selection would be of use.

Basic Signal processing knowledge such as Fast Fourier Transform and Group delay and harmonics and signal filtering methods knowledge is mandatory.

2. Introduction

This report serves the purpose to explain the process and methodology observed in completion of this project work whose primary objective is to detect the postural transition of the human body using a wearable sensor.

2.1. Problem Objectives

The objective of this project is to detect posture transitions of the human body. Namely Sit-Stand(SiSt) and Stand-Sit(StSi) transitions using a laboratory built inertial measurement unit which can be tested on patients/volunteers and data is collected. This collected data needs to be preprocessed and features need to be extracted following which machine learning classification algorithms are employed which are used to classify these postural transitions.

2.2. Project Scope

The study conducted in this project serves the purpose of further work in this field of Human Activity Recognition and Chronic/Geriatric Disorder detection and prevention schemes. An average human grows old and becomes frail. He might fall/face injury during a postural transition such as StSi and an SiSt or even other transitions. Thus study of the nature of these transitions, the time taken to make such transitions and even predicting the accuracy of these transitions by wearable sensor units have become center-piece in monitoring the frailty of patients namely the old patients.

Also, some of the techniques used in this project during the recording of data and labelling phase such as ‘Sensor Flip’ technique can be used for accurate ground truth annotation during the labelling phase, thus enabling a better supervised learning model.

2.3. Problem Approach

Based on the purpose and scope of the project the following approach has been formulated to achieve the main goal of classification. It is tough to focus on the output without clearly defining the approach and with a defined approach it is easy to be organized about the results and plots and output data and draw appropriate comparisons. The approach is as proposed.

The main problem is to detect the postural transitions of the human body provided by volunteers wearing sensor modules. This needs knowledge of previous attempts in order to reduce error and get a clear picture of the path to be followed. Also the scheme of data collection must be such that we get almost equal samples of postural and non-postural transitions in order to achieve a balanced data set.

It must also keep in mind that while collecting data from the volunteers the delay caused due to sensor and video camera delays needs to be taken into account and a technique to reduce that delay needs to be devised and synchronize both sensor and video camera recordings.

The collected data is then accessed using MATLAB and collected in an MS-Word document which will be used for calculating the delay time for synchronizing the video and sensor readings and also to label the signals.

The raw sensor data simultaneously is processed in a MATLAB environment using a filtering technique and split into windows of a fixed time interval and made to overlap. These windows count as samples and for each window the relevant features need to be identified and extracted. Using these extracted features which are grouped together in a feature vector for each window sample. A dataset can be constructed which is ready for feature selection.

The feature selection is done in order to prevent overfitting of the classification algorithm output thereby reducing accuracy and thus giving a wrong output. Thus feature selection gives the top features which can be used to classify the data and the rest of the features need to be discarded.
Finally, this optimized dataset is split into testing and training data and the training data is used to train a classifier along with its output labels. The results are checked on the testing data and the accuracy is analyzed on a confusion matrix to determine classifier accuracy.

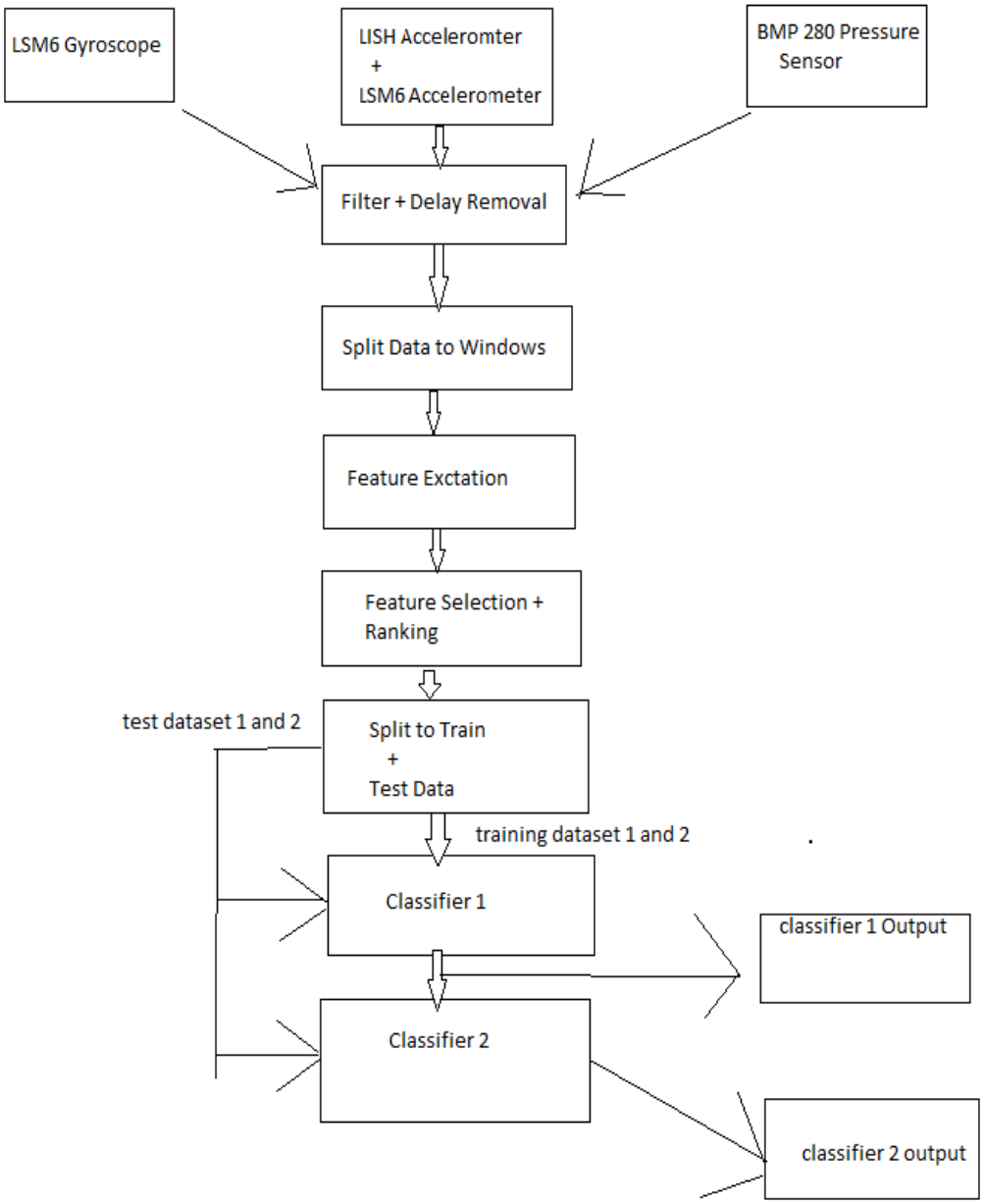

Figure 1 : The Machine Learning pipeline for this project

3. Literature Survey on Human Activity Recognition

Human Activity Recognition is picking up among computer science and electronic professionals as a means of determining various facets of the human body such as posture, gait, frailty, stability and is used to detect diseases onset and enable preventive healthcare or provide statistics to the end user for the purpose of personal monitoring or even to study the movement of humans and try to emulate that to robots thereby mimicking natural movements.

One of the aspects of human activity recognition is the study and detection of postural transitions as it has a direct impact on determining the body frailty of humans especially old people, whose falls and accidents can be prevented on accurately studying the fall and postural transitions patters.

By 2030, it is determined that 11.09% of Spain's population is going to cross the age of 75 [1]. This presents a challenge of risk of falls and accidents which if sustained and not prevented can pose a threat to their health and make them dependent [3]. A sample of people off which 32% of the people were 75years or older fell minimum once per year of which serious injuries were sustained by 24% [2]. This happens due to various reasons as a result of a combined effort from multiple small disabilities which directly depend on gait, postural transitions [4].

Human activity recognition relies heavily on sensor based data which are worn either on the wrist, waist, toe, neck, thigh and sometimes even the chest. In some experimental setups more than one sensing units ae used in different parts of the body for more accurate results [5]. The sensors used typically are accelerometers, gyroscopes. A device which contains both accelerometers and gyroscopes and sometimes even pressure sensors is called an Inertial Measurement Unit (IMU).

For postural transition detection even pressure sensors can be used. The accelerometers are typically tri-axial accelerometers here each posture or posture transition and changes can be seen as a corresponding change in one or two or all the accelerometer axes. This feature makes an accelerometer inside an IMU an important part of the posture detection process.[6]

This sensor data is filtered typically using a Low Pass Butterworth filter of a specific cutoff frequency to reduce noise. In the case of postural transitions, it is determined that most of the transitions occur below a set frequency which can be determined by the frequency response of the signal [7]. If the trained machine learning model is going to be in an embedded device, then the signal is resampled as well by the required number of samples to make it easier for hardware to process.

Typically the features selected from this sensor data is a mixture of time dependent data and frequency domain data [8] such as Minimum, Mean, Standard Deviation, Crest Factor and Kurtosis, Spectrogram, Harmonics and Signal Power respectively.

The features also are very high in number and need to be reduced using various dimensionality reduction techniques most popular of which is Principal Component analysis [9] using either Eigen values and Eigen vectors or Singular Value Decomposition respectively.

Some studies also employ the feature selection technique to pick the best features by ranking them using ranker algorithms such as Wrapper[10], ReleifF[11] etc. Feature ranking for selection can be employed alongside with dimensionality reduction techniques but that depends on the application specific approaches. In this study the PCA algorithm was reducing the accuracy as the groups of features were offering similar variances and the Wrapper algorithm though accurate is more computationally intense and thus only feature ranking algorithm ReleifF[16] suggested by Kiera and Rendell has been used to select the best features that can be used for classification.

The majority of the machine learning approaches for human activity recognition and detection of postural transitions use supervised learning methods[6,10,11] which include manually labelling sensor data and training a supervised model with it. This model before training is split into training code and testing code and training is performed for a holdout of a small sample of data on which this trained model is tested. The mean of accuracies of all the testing of the trained models on the holdout data gives the final training accuracy and testing of the model on the test data gives the final testing accuracy which otherwise is called ‘Model accuracy’.

The labelling which is done manually using sensor data is a very tedious process and to make the process easier the most popular technique is to use a video camera to record the volunteer activity and outline corresponding time activities in the video to the sensor.

However, this can be only done if the sensor and the video camera are of the same recording frequency which is not the case most of the times [12]. Thus a ‘delay parameter’ calculation is a method proposed in this project based on experiments performed in CETpD-UPC . However, the most popular synchronizing technique includes making the volunteer do a choreographed set of moves before the main activity. A method proposed is sudden transitions for effective labelling [13]. Labelling using this process is called ‘Ground Truth Annotation’

An obsolete technique for labelling has been the use of stop watches[14] for each activity and each transition. However, such an approach can lead to errors if the person recording is not fast enough and experienced enough. Also, it doesn’t have the comfort of accurately determine the body position visually as done in the video camera recording technique.

The most popular classifiers for this purpose are the Support Vector Machines and Trees because of their simple use and good amount of documentation available and generally high accuracy. Also, since IMU data is generally nonlinear in nature, it is better to use the above mentioned methods which train well with nonlinear datasets. However, they are the best for binary classification in supervised learning thereby requiring the use of sequential classifiers for various classes of the output.

4. Human Activity Recognition

4.1. Postures and its Transitions

It refers to a way a person does an activity such as sitting/standing/walking/running and other similar activities. The major part of this study concentrates only on transitional postures. The posture of a person can tell about the general health of a person. A simple analogy of observance between the way a healthy man stands and the way a sick man stands can tell the importance of posture recognition.

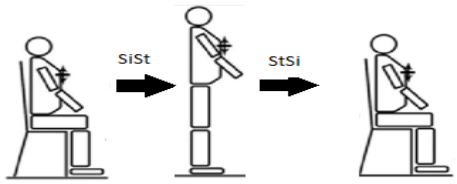

Figure 3SiSt and StSi Transition

Similarly, postural transitions are important because majority of accidents associated with old people with weak musculoskeletal systems happen during a postural transition. Also, the basic transition of StSi and SiSt is important perquisite for other more mechanically demanding physical activities such as walking, standing, running. Thus it is important to accurately classify them and this data can be used to predict a variety of physical conditions such as Parkinson's disease, arthritis and fall prevention.

4.2. Typical Sensors used

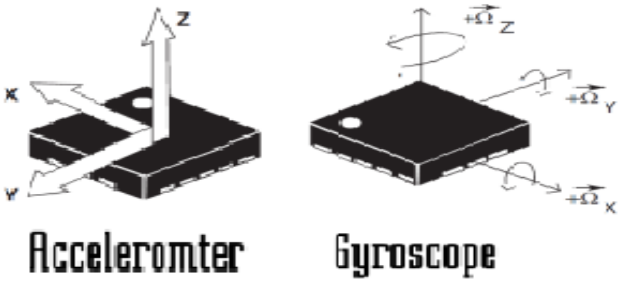

Figure 2Accelerometer and Gyroscope Axes

Sensors used for postural transition recognition does include accelerometers and gyroscopes as they provide data such as acceleration, rotation and translational movement components. Also, pressure sensors have been used which can show difference in levels with respect to atmospheric pressure. This can be used to detect difference in level activities such as sitting and standing. The Accelerometers can differentiate between the running activity and walking activity because they occur at different frequencies. An Accelerometer sensor detects changes in acceleration which stresses the Piezoelectric sensor and a voltage is induced through the leads proportional to the acceleration.

Thus, to detect various postures and transitions all these sensor combinations need to be used which gives way to the Inertial Measurement unit which is typically a combination of accelerometers and gyroscopes fitted on a PCB board. As the name suggests it measures the movement and can be used for various embedded applications paving way for low cost and efficient detection of posture and its transitions.

4.3. Problems in Classification

While classifying the data for human activity recognition there are various variables which affect the accuracy of the trained algorithm and of the test cases. Some of them are:

- Un-synchronized sensor and ground truth annotation scheme readings.
- Wrong labelling which is mostly done manually if supervised learning is used.
- Feature selection can sometimes generate features with negative weights and use of those weights while training the classifier.
- Not removing the delay between the original data and the filtered data.
- Use of features which are linearly overlapping and fail to highlight differences between samples.

5. Sensors Used in this Project

Sensors used in this project are 2 accelerometers, one gyroscope and one pressure sensor. They are named as LISH Accelerometer, LSM6 Accelerometers, LSM6 Gyroscope, BMP280 pressure sensors. The actual model names used are

- LSM6DSM Accelerometer: Manufactured by ST Microelectronics. Supply Voltage of maximum 3.6V. It consumes a maximum of 0.65mA in high performance mode. 4kB first in first out (FIFO) transmission. Full scales includes +2g to +16g and -2g to -16g.

- LSM6DSM Gyroscope: Manufactured by ST Microelectronics and comes inside the LSM6DSM IMU along with the LSM6DSM Accelerometer. It also offers a full scale of -16g to +16g. It has interrupts for freefall and can also function as a step detection and tilt sensor.

- LIS2HH12 Accelerometer: Manufactured by ST Microelectronics. It can detect ranges of -8g to +8g. It needs a supply voltage of 3.6V and has a 16bit data output which can transfer data up to 800Hz.

- BMP280 Pressure Sensor: Manufactured by Bosch Sensor Tech. It can detect temperature and pressure up to 1100 hPa and 85C. Supply voltage of 3.6 V is needed and data resolution is 0.01hPa and 0.01C.It uses a minimum current of 0.1e-6 A

6. Sensor Data Collection Scheme

The Sensor data has been collected in non-laboratory real world conditions in Barcelona, Spain. 9 volunteers were used for this purpose of which 8 were male and 1 female.

The volunteers were all healthy and physically fit and were between the ages 20-25 years with the median age being 22 years. They were all students and reported no history of bone or muscle based weakness.

The Inertial Measurement Unit which was attached to the belt was used for collecting the data. There was also video camera based evidence of sensor recording of volunteer postures using ground truth annotation. The Video camera was switched on and the sensor was switched on 10 seconds later. At the 20th second of the camera recording the sensor was flipped along the y axis and 10 seconds wait period was granted following which the volunteer would wear the belt. The volunteer would perform a SiSt/StSi motion every 10 seconds after that a total of around 300 seconds. The volunteer would then remove the sensor and place it such that y-axis of the sensor would face upwards and after a 10second wait period the sensor would be flipped. This flipping of the sensor follows for another 10second wait period where the video camera and the sensor are switched off together.

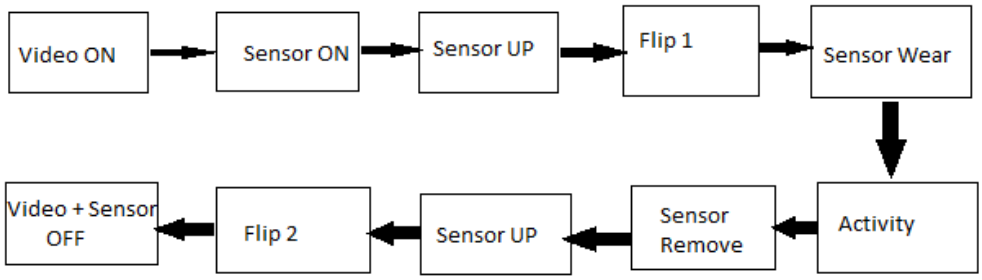

Figure 4Sensor Data Recording Scheme from each Volunteer

The video camera and sensor recording are read together for labelling process. The sensor data is opened in Matlab and arranged in the form of windows explained later in this report. The video camera is used to note down the initial flip time and final flip time of the sensor and the ON-time of the sensor. This flip time corresponds to a change in the y-axis in the accelerometer data vs time graph which can be plotted using Matlab.

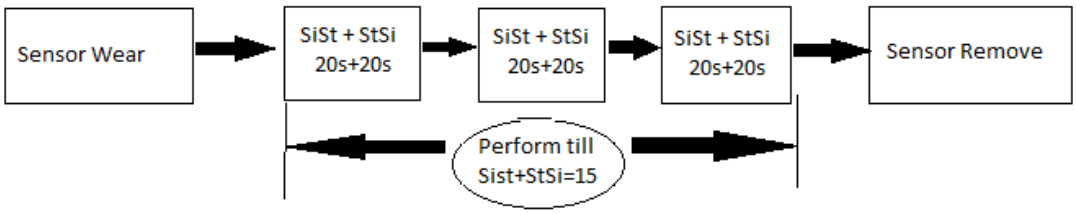

Figure 5 Activities Performed by each volunteer

The derivation of the delay time is as follows:

S_{ONV} = Sensor ON time in the Video

T_{FlipIV} = Time of Initial Flip in Video

T_{FlipFV} = Time of Final Flip in Video

T_{flipIG} = Initial Graph Time when y-axis value suddenly dips(Max Value) and other axes cross.

T_{flipFG} = Final Graph Time when y-axis value suddenly dips(Max Value) and other axes cross.

Actual First flip time is $T_{initialFlip} = T_{FlipIV} - S_{ONV}$

Actual Second flip time is $T_{finalFlip} = T_{FlipFV} - S_{ONV}$

First it must be verified that that the difference between $T_{initialFlip}$ & $T_{finalFlip}$ are the same or almost same as the difference between T_{flipIG} & T_{flipFG}. This proves that the correct points in the graph have been considered and the video times taken have also been accurate.

Now for the Delay Time calculation,

$$T_{difference1} = T_{initialFlip} - T_{flipIG}$$

$$T_{difference2} = T_{finalFlip} - T_{flipFG}$$

It must be observed that the differences between $T_{difference1}$&$T_{difference2}$ is very less and thus they can be considered almost equal. However a better method would be to take their difference and use that as the delay time.

$$T_{delay} = \frac{T_{difference1} + T_{difference2}}{2}$$

For more accurate value this delay time calculations can be performed a few more times and the mean of all the delay times can be used as the final T_{delay}

Figure 6: IMU Upright(Left) and IMU Flipped(right)

Figure 7: Location of the IMU belt. Just above the pelvic bone.

7. Signal Pre Processing

The filtered signal which is used is filled with all sorts of noise and unwanted components. For this purpose, one must filter the signal. Also, the classifier for machine learning will take only datasets arranged in a tabular fashion with the columns being the feature vector elements/features and the rows being the samples or windows.

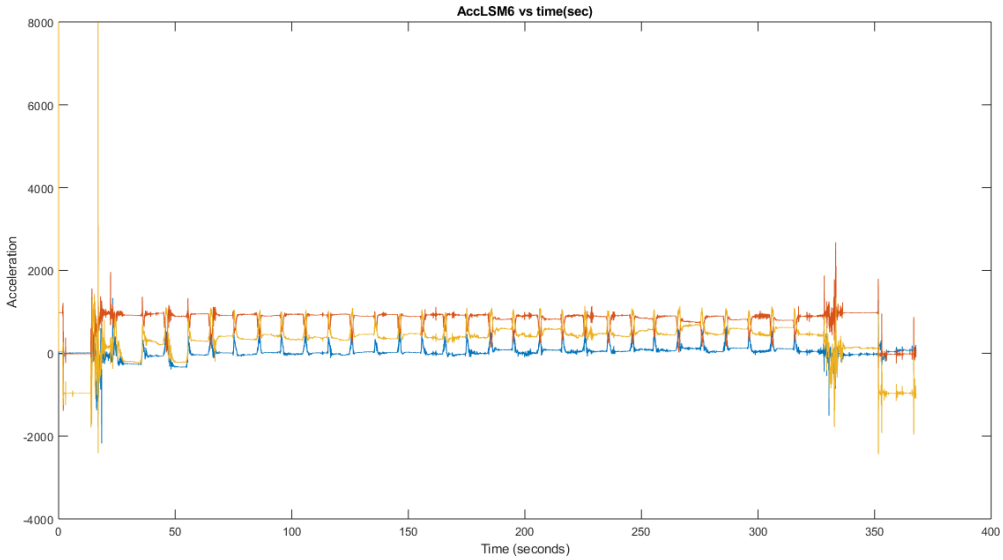

Figure 8: LSM6 Accelerometer vs Time . Unfiltered and Noisy

7.1. Filtering

The first step in preprocessing of the IMU signals is the filtering technique which employs a Low Pass Butterworth signal in order to filter out the high frequency noise. During the recording of the sensor data the volunteer might have suddenly changed his position or may have jerked which are actions which induce noise into the system. This is undesirable and thus needs to be removed. The filter is given an order of 10, cutoff frequency 10 Hz and sampling frequency of 200Hz which is also the sampling frequency of the IMU.

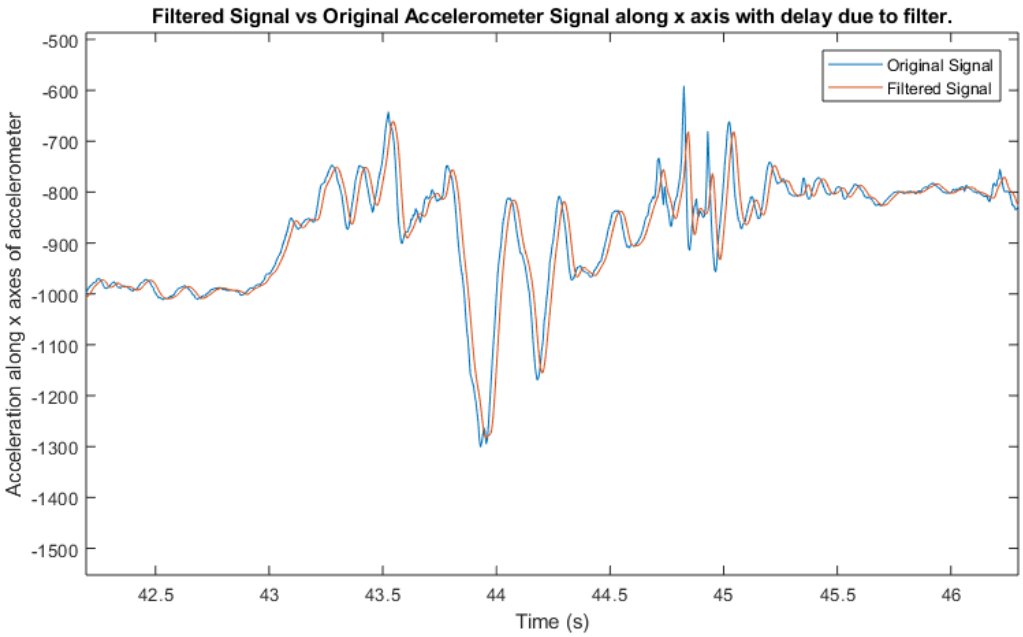

Figure 9: Original LISH Accelerometer Signal(x) vs Filtered signal(x) without adjusting for delay caused due to Filter.

7.2. Removal of Filter Delay

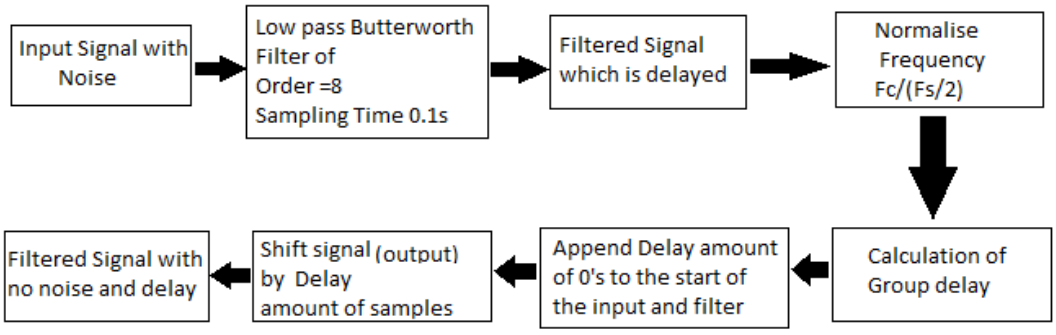

Figure 10: The Delay Removal Algorithm

The Filter induces a delay in the system which can cause inaccuracies during the labelling process and windowing techniques by shifting the sample by a small amount of samples forward. Since the sampling frequency is 200Hz (0.005 seconds) even a small delay of 0.01 second can cause wrong labelling and increase in misclassification. To counter this immediately after the filtering process the Group Delay is calculated.

The Group Delay is the measure of the time delay of the amplitude. It is a measure of how much two signals which are passed through a signal processing unit are delayed by in units of either time or samples.

Signals when passed through filters or other devices such as amplifiers undergo a delay which results in the filtered signal samples not being synchronized with the original signal samples with respect to the time.

Signals of multiple frequency components such as body signals in which dynamic motion occurs in a different frequency compared to static postures and different from postural transitions suffer distortion because the samples of various frequencies are not delayed by the same amount of time at the output of the device. This can result in scale change or even shape change.

The group delay is computed using the slope of the phase response at any given frequency of the device under test. Thus it can be calculated by differentiation the phase response of the filter with respect to the frequency.

Considering a Linear time invariant(LTI) signal which can be broken down into sine and cosine components is of the form

$$x(t) = b(t)\cos(wt + \theta)$$

where the amplitude $b(t)$ is changing slowly with respect to the frequency w.

Thus $\left|\frac{d}{dt}\log\big(b(t)\big) < w\right|$.

Thus the system frequency response which is the convolution is defined as

$$y(t) = |H(iw)|b\big(t - t_g\big)\cos(w(t - t_\emptyset) + \theta)$$

Where t_g and $t_\emptyset$ are the group delay and phase delay respectively.

We know that $\emptyset(w)$=arg{$|H(iw)|$}

$\emptyset(w)$ = phase shift and

$H(iw)$ = the transfer function.

Thus, phase delay is $t_\emptyset(w) = -\frac{\emptyset(w)}{dw}$

And Group delay $t_g = -\frac{d\emptyset(w)}{dw}$

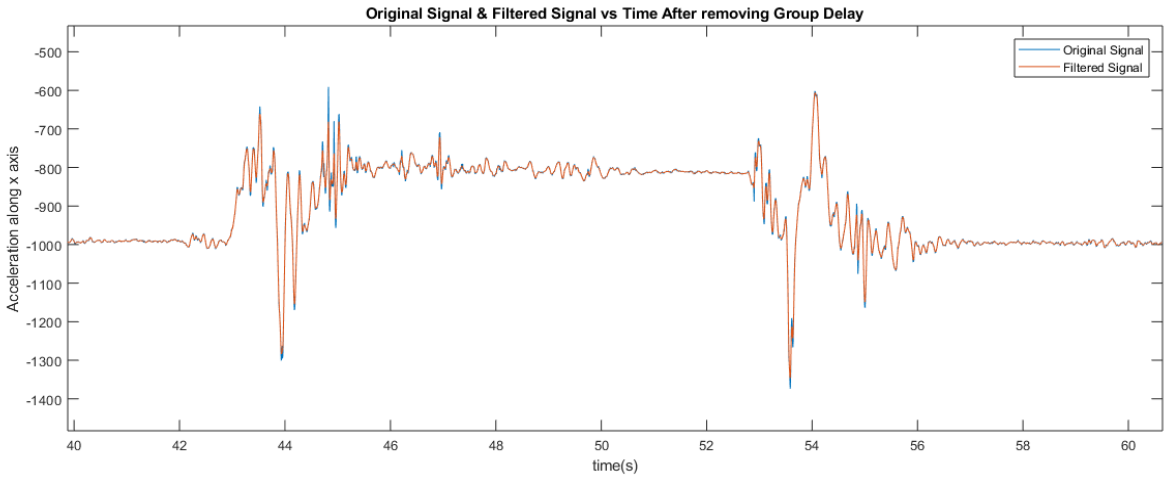

Figure 11: Filtered Signal vs Original Signal after removing Delay for one axis of LISH Accelerometer

7.3. Windowing Technique

The Filtered signal after removal of delays is grouped together into samples between a set region of time. This is called a window. Since this IMU signal is a time-series signal the most optimum way to process it would be to highlight its changes and try to give an input to the classifier which can enumerate these changes. If an individual sample approach is taken, then one cannot know the changes in the samples.

Thus a windowing method is used where a set of signal samples for a specified amount of time is taken as a window and multiple such windows are taken at various time intervals. These windows can overlap so that the changes are observed consistently over which features can be extracted. Typically, the best overlap ratio should be 50%. This method is known as 'Sliding Window Technique'.

After seeing the signal samples, it is determined that that least time is taken by the postural transition actions which are between 1.2 seconds and 2 seconds. Thus, a window is needed which can take the whole transition between is time period.

Since the feature extraction will contain frequency domain features which shall make use of Fast Fourier Transform algorithm, it is better to choose a window length of a power of two to ensure no zeros are appended to the start or end of the window due to a Fourier transform.

It is seen that a 256 (8th power of 2) sample window takes 1.275 seconds and a 512 (9th power of 2) sample window takes 2.56 seconds. Since we have the minimum transition time of 1.2-2 seconds it is best to take a 256 second window with 50% overlap so that a window can capture most or whole of the postural transition.

Variables - ForLabelling

ForLabelling

543x4 double

	1	2	3	4
1	1	256	0	1.2750
2	129	384	0.6400	1.9150
3	257	512	1.2800	2.5550
4	385	640	1.9200	3.1950
5	513	768	2.5600	3.8350
6	641	896	3.2000	4.4750
7	769	1024	3.8400	5.1150
8	897	1152	4.4800	5.7550
9	1025	1280	5.1200	6.3950
10	1153	1408	5.7600	7.0350

Figure 12: Window Intial and Final Sample Values as shown in column 1 and 2 . Time initial and Final Values as shown in column 3 and 4

8. Feature Processing

Feature processing for posture analysis plays an important role as the features must be chosen such that they can highlight the difference between various postural transitions. The features which are decided upon are applied over the time windows extracted as applying features for each sample is tedious and does not provide much relevant information.

Popular features in time domain and frequency domain used in mean, standard deviation of gravitational acceleration and skewness and kurtosis of FFT of the IMU signals.

For posture classification, different features have been used in the literature. Popular choices are mean and standard deviation of the signal over the time window. The mean of the gravitational acceleration in a window can be used to distinguish between postures [38], where the standard deviation can be used to distinguish between static and dynamic activities. The correlation between different accelerometer axes can be used to distinguish activities that move in one dimension (such as walking) from activities that move in more directions (such as climbing the stairs) [8]. Other features include signal skewness and kurtosis [2], and shifted delta coefficients (slope of the signal in different points) [1]. Features are extracted not only from the time domain, but also the frequency domain, after performing an FFT. The frequency domain feature energy [20] can distinguish between periodic (such as walking) and non-periodic activities. Another frequency domain feature is entropy, which measures the complexity of the signal. This can distinguish between movements of different complexities, such as cycling and walking [20]. Park et al. [28] use spectral energy and FFT magnitudes as features.

8.1. Feature Extraction

A combination of time domain and frequency domain features are extracted as time domain features can help identify the variations in raw sensor signal windows with respect to time using statistical quantities such as Mean, Standard Deviation, Crest Factor etc. The frequency domain features help identify the components of the signals occurring in various frequencies such as harmonic values, spectrogram, signal power etc. The features extracted for each sensor are as shown next:

Feature Type	Feature Name	Number of Features
Time	Mean of all axes	3
Time	Maxima of each axis	3
Time	Standard Deviation of each axis	3
Time	RMS quantity of each axis	3
Time	Kurtosis of each axis	3
Time	Skew of each axis	3
Time	Range of each axis	3
Time	Crest Factor of each axis	3
Time	Integral Value of each axis	3
Time	Difference of current window from previous window	3
Time	Mean Difference between X&Y axis , Y&Z axis and X&Z axis	3
Time	Slope of Linear Regression of each axis	3
Time	Mean of Window Difference	3
Time	Standard Deviation of Window Difference	3
Frequency	Maximum of FFT of each axis	3
Frequency	Sum of Harmonics of each axis	3
Frequency	Maximum Harmonic for each axis	3
Frequency	Kurtosis and Skew of FFT	3
Frequency	Mean Harmonic Frequency	3
Frequency	Entropy of FFT signal	1
Frequency	Power Spectrum of Harmonics	3
Frequency	Mean Frequency of Harmonics	3
	Total	65

Table 1 : Features Extracted for each window of each sensor

8.2. Feature Selection

A lot of features mentioned in the above table might either be redundant or some of the features will provide noise. The problem with having too many features is that it confuses the classifier and may result in higher misclassification rat and lower accuracy. This project uses a Feature ranker to reduce the feature set.

8.2.1. Over Fitting

Each Feature is a dimension in the data. Thus the higher the dimensions, the higher there are chances that overfitting can occur during classification. Each classifier essentially tries drawing a curve through the feature points and tries to reduce the Euclidean distance between them. However, the problem with multiple features is that there are many points with a considerable spread which cannot be covered in a curve or straight line, thus creating an output response which is not accurate.

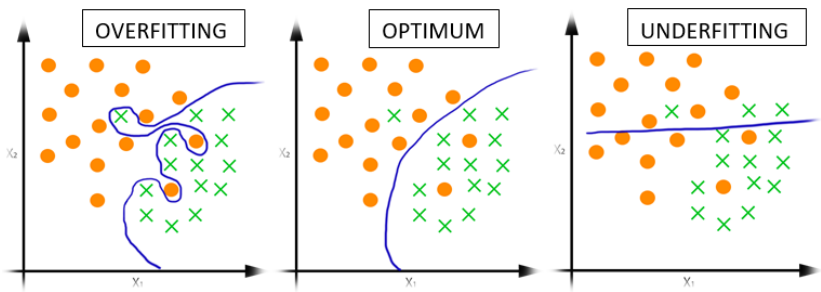

Figure 13 : Various types of curve fitting

8.2.2. Dimensionality Reduction

To avoid problems such as overfitting and to make the data more computationally efficient. The number of features(dimensions) are reduced by using a feature selection algorithm. The approach used in this project is to rank features and choose the most valuable features. There are multiple feature selection and ranking algorithms such as “Wrapper”, “Neighborhood component Analysis” etc. The ReleifF algorithm has been used over here for feature ranking and selection.

8.2.3. K-N-N Algorithm (ReleifF)

The nearest neighbor algorithm based ReleifF was suggested by. It is used for feature selection using the ranking of features. Every iteration compares the ‘k’ nearest samples to a chosen sample from the ones in the dataset and compares the labels between them. Thus it is a supervised approach. If a dataset of n windows/samples and f features are considered of which every instance ‘n’ belongs to a known class off 2 classes. The ReleifF algorithm first scales the database down to [0,1] such that they feature have similar scale Euclidian distance so that all features are contributing equally in the start. At each iteration a feature vector ‘f’ is chosen whose closest neighbouring samples ‘x’ is chosen for computation. The closest same class ‘x’ value is called ‘near hit’ and the closest different class ‘x’ value is called ‘rear miss’. Now the weight vector is derived where weight W_i is given as

$$W_i = W_i - (f_i - NH_i)^2 + (f_i - RM_i)^2$$

which is updated for every iteration.The score for each comparison is given and if the nearest neighbor is closer in the same class then the score is increased else the score is decreased. If another class neighbor exists in the nearest neighbours then depending on the closeness to the selected sample the score is reduced. Finally, at the end of each iteration, each feature is assigned a rank from high to low depending on the scores and finally the features are ranked from highest rank to lowest rank.

8.2.4. Choosing the Optimal K-Value

The value of ‘k’ must be chosen such that it is not too less or too much. A low ‘k’ fails to provide any experience as the scores and ranks diverge a lot with every iteration. A higher ‘k is more computationally intense and may fail to detect changes in the response variable. The in this project the ‘k’ value is selected using a graphical approach for each iteration. The score value stabilizes after a certain value is what has been observed. That is the value of k which is needed for optimum feature selection. The standard deviation of each iteration of each ‘k’ value is plotted against the value of ‘k’ itself and the point where the standard deviation becomes constant values is the best ‘k’ value. From this we can see that for C1 and C2 K-values are 311 and 321 respectively.

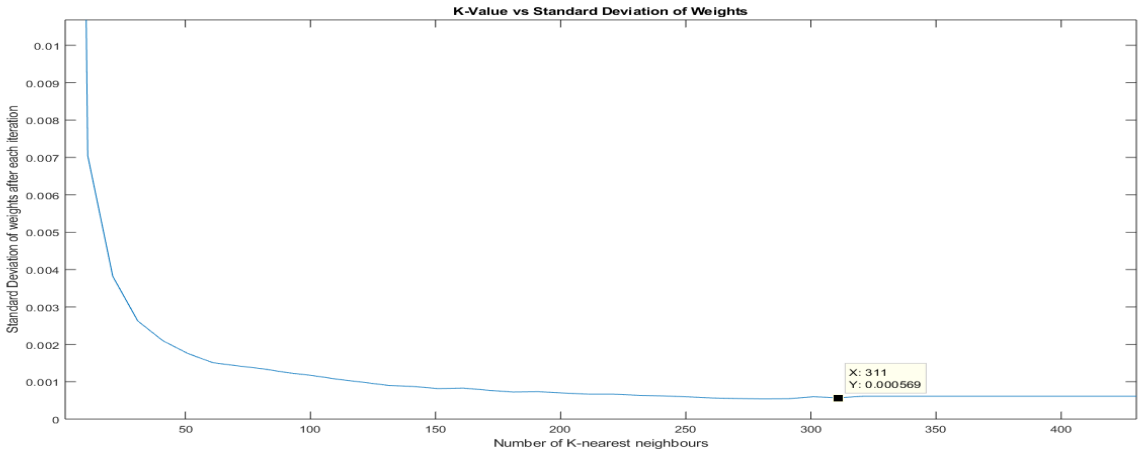

Figure 14 : Standard Deviation of weights vs k plot for Classifier 1

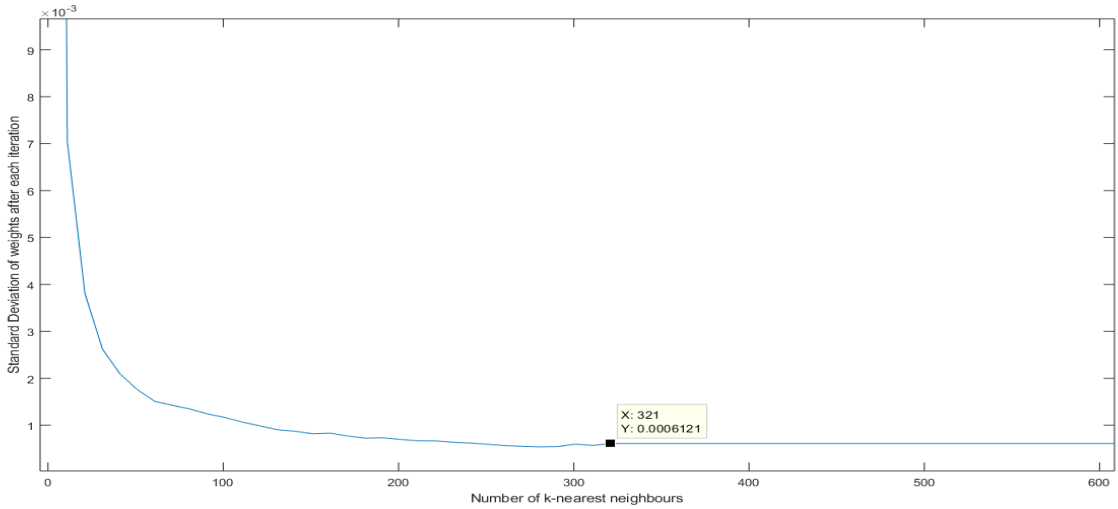

Figure 15: Standard Deviation of weights vs k plot for Classifier 2

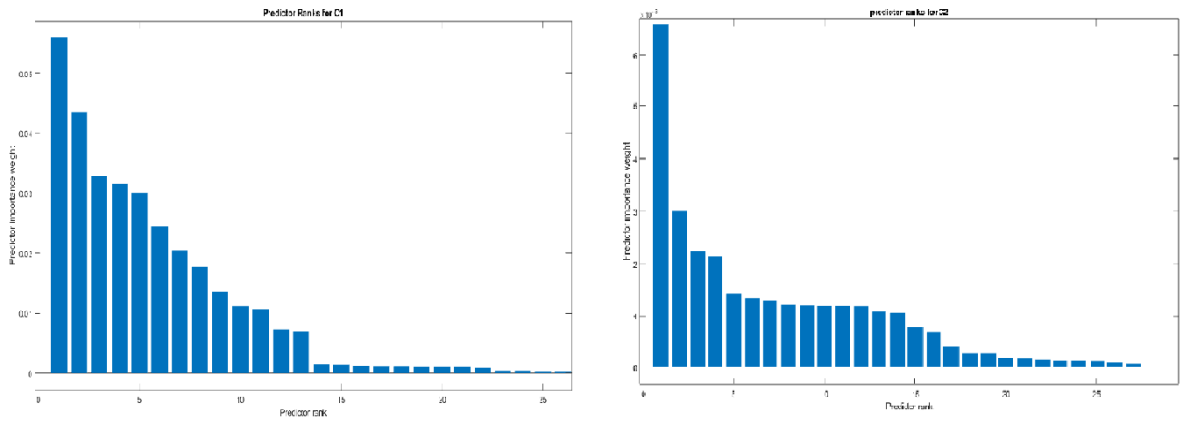

Figure 16: Predictor Ranks for C1 and C2

Variables - ranksdday

ranksdday

15x2 double

	1	2
1	169	169
2	177	116
3	203	55
4	170	109
5	185	126
6	55	127
7	116	65
8	48	6
9	107	47
10	168	67
11	109	4
12	190	48
13	47	66
14	126	128
15	127	5

Figure 17: Top 15 predictors shown of C1 and C2

9. Classification Strategy

9.1. Supervised Learning

Supervised Learning is the process of defining output variables for each input variable before training the dataset using a machine learning algorithm. This methodology does a one to one mapping of the inputs and outputs and lets the machine decide on the relation based on ranges of input and output relations and the distance between them whenever a new data is given. Supervised learning can further be divided into regression and classification where the output variables are continuous real value and discreet respectively.

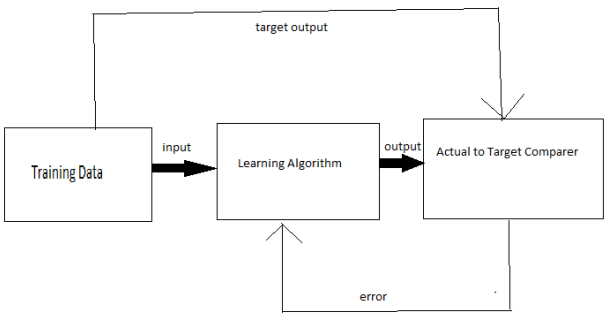

Figure 18: Supervised Learning Technique

9.2. Classification

The Classification of data is the methodology of determining a class of new data which is the input based on the training from the training database which consists of independent feature variables and corresponding labels which are discreet in nature. It predicts the class of the new data based on these labels. The unsupervised learning equivalent of classification is called clustering. Some popular classification algorithms include Support Vector Machines, Trees, Random Forests, Logistic Regression. The main distinguishing feature of classification is that the data points used for training must have discreet labels

9.2.1. Support Vector Machines

The n dimensional data points used for classification may be nonlinear in nature and may require a distinguishing method to separate them in order to classify.

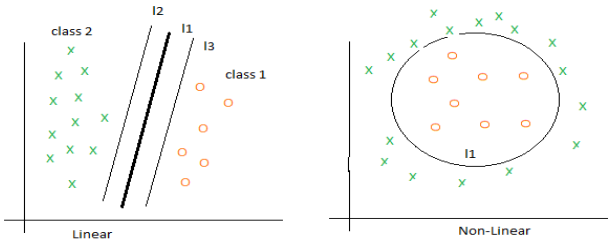

Figure 19 : Linear and Non Linear SVM

However sometimes it may not be possible in that dimension. This is where Support Vector Machines (SVM) converts this n dimensional data into an n+x dimensional data where x is the number required to convert the nonlinear data to linear data.

Essentially this is called Kernel trick where the data from a lower dimension and non-separable is transformed to a higher dimension in which it is separable. This data in the higher dimension is then separated by drawing of a line/curve which separates the data in two classes (Binary Classification). This line is called the Hyper-plane. It must not be close to either of the classes as shown in Fig 16. By lines l2 and l3. The best hyperplane is generally equidistant from the closest different class points.

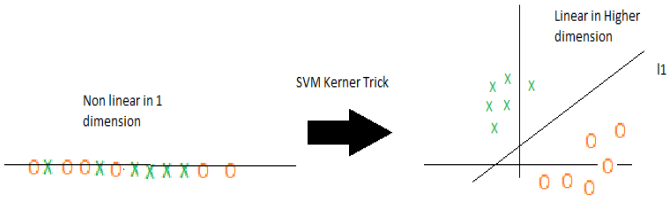

Figure 20 : Kernel Trick (Transforming to higher dimensions)

SVM's advantages are realized in high dimensional datasets where features have a clear separation. Also, it uses less memory. However, it doesn't perform well on noisy data and takes a long time for large datasets which do not have a high variance.

9.2.2. Trees

The reasons Trees are popular for classification is because they represent information clearly and are easy to use. It is a methodology which comprises of a hierarchical structure and takes one of two decisions at every step eventually proceeding from the root node on the top to the last leaf node which is one of the outputs. At each iteration the best feature is selected and perform a split based on thresholds derived from the specific splitting criterion. Then a cost function is calculated based on this threshold and this is evaluated for all features. The lowest cost function iteration is chosen as the final trained tree model. The measure of randomness of data is called purity of the tree. It is not feasible to predict class in an impure tree.

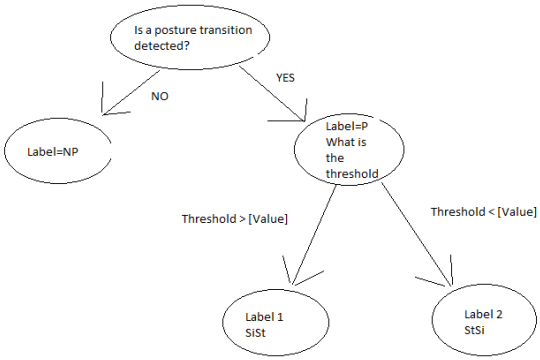

Figure 21: Trees for Postural Transitions

The threshold mentioned above for the tree to split accurately can be determined using two functions namely CART and ID3 Algorithm. The cart algorithm and ID3 algorithm derive the cost function based on two variables called Gini Index(GI) and Information Gain(IG) respectively.

The GI is defined as the proportion of all the variables with a specific class which is subtracted from 1. Lower the GI value, better the tree splits the dataset.

GI for a dataset of n classes = $1 - \sum_0^n P_x$ where P_x is the proportion of variables of class x

Information Gain is the difference of entropy of dataset before and after it is split based on feature F. It is a measure of the reduction in uncertainty of the dataset and higher the value it is the better the tree splits.

$$IG = H(S) - \sum_{x \epsilon T} p(x)H(x)$$

where IG= Information Gain

$H(S)$= Entropy of the split dataset S

$p(x)$= proportion of number of elements in x to the number of elements is S

$H(x)$=Entropy of Subset x

9.3. Splitting of Data into Test and Train

Once the optimum k-value is chosen and the features are ranked. The top 10 features are chosen and used to train the dataset on a classifier. However, it is beneficial to test the data on unknown data which is not used for training. Thus before the training stage the database is randomly split into training data and test data in a ratio of 90% to 10%. This can let us know the performance of the classifier on unseen data and can result in better quality classifiers.

9.4. Training the data

The data is trained on 2 classifiers which are named as C1 and C2. C1 detects whether a postural transition has occurred or not and C2 detects if a postural transition which has occurred is a SiSt/StSi transition. The data is trained using MATLAB's machine learning toolbox with a holdout. The data holdout splits the training data in train and holdout data on which the trained model is tested. This process is repeated multiple times for different holdout data which is done automatically by the Classification Learner toolbox in MATLAB. The accuracy of the trained model is the mean of all the holdout test accuracies. The holdout is chosen as 10%. Thus every iteration 10% of the data from the training sample is kept aside for inside the classifier testing.

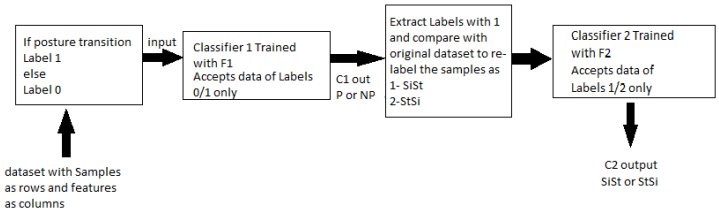

Figure 22: Relationship of C1 and C2

Iteration Number	Training Sets	Testing Sets
1	1,2,3	4
2	1,2,4	3
3	1,3,4	2
4	2,3,4	1

Table 2 : An example of 25% Holdout with 4 samples.

9.5. Test Results

9.5.1. For Classifier 1

Classifier 1 predicts the labels P and NP where P (1) signifies that a Postural Transition has occurred and NP (0) signifies that a Postural Transition has not occurred. The training for C1 has been performed on many classifying algorithms of which the top 2 have been compared below.

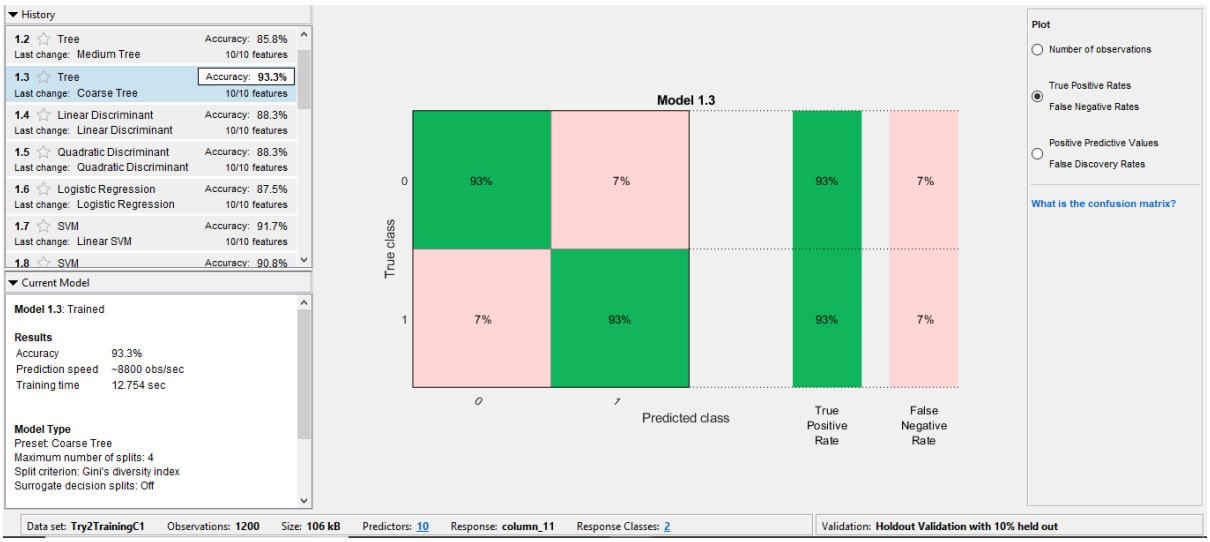

Figure 23 : Coarse Tree Confusion Matrix just after Tests on Holdout

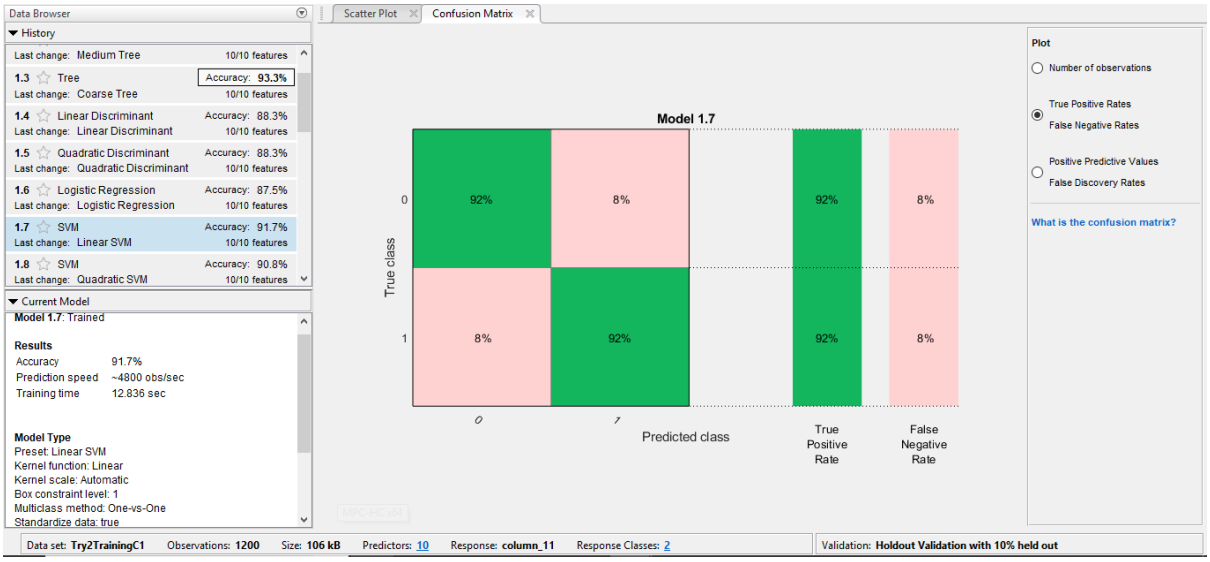

Figure 24: Linear SVM Confusion Matrix just after Tests on Holdout

After running the 'test data' through both the classifiers in C1 we derive their confusion matrices and see the results.

It is observed over here that :

For Coarse Tree

TP = 19

TN = 14

FP = 0

FN = 7

Specificity = $\frac{\text{TN}}{\text{TN+FP}} = \frac{14}{14}$ which corresponds to 100%

Sensitivity = $\frac{\text{TP}}{\text{TP+FN}} = \frac{19}{26}$ which corresponds to 57%

Accuracy = $\frac{\text{TP+TN}}{\text{TP+TN+FN+FP}} = \frac{33}{11}$ which corresponds to 80.48%

	Predicted 0	Predicted 1
Actual 0	14	0
Actual 1	7	19

Table 3 : Coarse Tree Confusion Matrix

For Linear SVM

TP = 25

TN = 13

FP = 1

FN = 1

Specificity = $\frac{\text{TN}}{\text{TN+FP}} = \frac{13}{14}$ which corresponds to 92%

Sensitivity = $\frac{\text{TP}}{\text{TP+FN}} = \frac{25}{26}$ which corresponds to 96%

Accuracy = $\frac{\text{TP+TN}}{\text{TP+TN+FN+FP}} = \frac{38}{40}$ which corresponds to 95%

	Predicted 0	Predicted 1
Actual 0	13	1
Actual 1	1	25

Table 4: Linear SVM Confusion Matrix

9.5.2. For Classifier 2

Classifier 2 is connected to Classifier 1 in such a way that the output of classifier one is used again for classifier 2 but without the 'No Transition' data and the 'Transition' data is labelled as 1 and 2 which means 'SiSt' and 'StSi' respectively. Various algorithms were tried for classifier 2 of which the ones with the top 2 accuracies after training are shown below.

Figure 25 : Ensemble Bagged Tree Confusion Matrix after Tests on holdout

Figure 26 : Ensemble Bagged Tree Confusion Matrix after Tests on holdout

After running the 'test data' through both the classifiers in C2 we derive their confusion matrices and see the results.

It is observed over here that :

For Ensemble Bagged Tree

TP = 18

TN = 8

FP = 10

FN = 4

Specificity = $\frac{TN}{TN+FP} = \frac{8}{18}$ which corresponds to 44%

Sensitivity = $\frac{TP}{TP+FN} = \frac{18}{22}$ which corresponds to 81.8%

Accuracy = $\frac{TP+TN}{TP+TN+FN+FP} = \frac{26}{40}$ which corresponds to 65%

	Predicted 0	Predicted 1
Actual 0	8	10
Actual 1	4	18

Table 5: Ensemble Bagged Tree Confusion Matrix

For Medium Tree

TP = 18

TN = 14

FP = 4

FN = 4

Specificity = $\frac{TN}{TN+FP} = \frac{14}{18}$ which corresponds to 77.7%

Sensitivity = $\frac{TP}{TP+FN} = \frac{18}{22}$ which corresponds to 81.81%

Accuracy = $\frac{TP+TN}{TP+TN+FN+FP} = \frac{32}{40}$ which corresponds to 80%

	Predicted 0	Predicted 1
Actual 0	14	4
Actual 1	4	18

Table 6: Medium Tree Confusion Matrix

10. Conclusion

The results above can be tabulated as shown below:

C1 Algorithm Name	TP	TN	FP	FN	%Sensitivity	%Specificity	%Accuracy
Coarse Tree	19	14	0	7	57	100	80.48
Linear SVM	25	13	1	1	96	92	95

From this we can observe that for classifying whether a postural transition has occurred or not is done best in C1 by Linear SVM. This is partly because on observing the data the difference between the postural transition and non-postural transition is very clear and thus it is easy to draw a hyperplane. Also since postural transition and non-postural transition data occur at different frequencies, there was a clear differentiation in the harmonics in the frequency response features which were the best for classifying this dataset.

C2 Algorithm Name	TP	TN	FP	FN	%Sensitivity	%Specificity	%Accuracy
Ensemble Bagged Trees	18	8	10	4	81.81	44.4	65
Medium Trees	18	14	4	4	81.81	77.7	80

For the dataset which is used for classification in classifier 2, it is seen that the accuracy is slightly lesser. This can be because postural transitions like SiSt and StSi are similar natured activities which happen at similar frequencies. Also, in this case the features of the pressure sensor got a higher rank as it is the pressure at SIT level and STAND level which helps us clearly differentiate between these two transitions.

One of the approaches that could've increased the accuracy of the trained model was to record the volunteer data in laboratory conditions. Since the data was recorded in the non-laboratory conditions it is possible that some error might have got into the system. The volunteers were only 9 in number and it is always beneficial to have a large amount of data so that the algorithm can be trained for various types of people.

It was seen during feature ranking that while dealing with postural transitions it is the barometer features which are the most valuable and when used correctly against the accelerometer and gyroscope feature can provide an accurate postural transition detection.

Its is wholly possible that since it is a pressure sensor thre are certain changes like passing a door or air from a column or window which may affect the readings. These effects need to be looked at and minimized.

We used ReleifF for feature selection. However Wrapper happens to be a more accurate feature ranking algorithm but it takes up too much time and computation and thus it was not used.

Holdout validation with other percentages like 80% and 70% could be used to increase the predictive power of the model for any generalized dataset.

Feature extraction methods can be improved by accounting for jerks and body based non gravitational acceleration as well which can increase the sensitivity and specificity significantly.

The most notable impact of this project would be its future scope in creation of economically inexpensive and portable devices which can provide low cost monitoring for patients based on specific classes of movement disorders. The refining of the techniques used above for more sensitivity and specificity and the improvisation of the algorithms used above to clue all common postures and postural transitions can be used for reliable clinical trials.

11. Acknowledgements

This project has been possible due to help from several people who have facilitated this evolution of the work and have extended their precious time in contributing to guidance and completion of this project. To some of them I make a mention of thanks.

To Dr.Andreu Catala Mallofre, Director of CETpD at UPC and this project mentor for the this project for setting the direction of this project and making me aware of the knowledge of various possibilities and requirements which I would experience as a part of working on this project.

To, Dr.Carlos Perez Lopez (CETpD-UPC) for guiding me on the difficult steps of this project and suggesting ideas and innovative solutions to solve roadblocks as explained below. And also to teach me whenever there was a conceptual error or vagueness in any step of this project.

To, UPC-ETSEIB and Amrita Vishwa Vidyapeetham for it was the bi-lateral agreement between the universities which enabled me to work as an exchange student abroad

To, Department of Electrical and Electronic Engineering Staff, HOD and to Dr. Anand R for being my guide and support in my home university.

To, all the volunteers for co-operating for recording data required for this project.

Finally, to my Parents and God for keeping me motivated and keeping me firm on my path.

Bibliography

[1] https://www.allianz.com/en/press/news/business/insurance/news-2012-09-10b.html

[2] M. E. Tinetti, M. Speechley, and S. F. Ginter, 1988, Risk factors for persons living in the community, N. Eng. J. Med., vol. 319, pp. 1701–1707.

[3] B. Najafi, K. Aminian, F. Loew, Y. Blanc, and P. A. Robert, 2002, Measurement of Stand–Sit and Sit– Stand Transitions Using a Miniature Gyroscope and Its Application in Fall Risk Evaluation in the Elderly, IEEE Trans. on Biomedical Engineering, Vol. 49, No. 8, pp. 843 – 851.

[4] M. E. Tinetti, T. F. Williams, and R. Mayewski, 1986, Fall risk index for elderly patients based on number of chronic disabilities," Amer. J. Med., vol. 80, pp. 429– 434.

[5] J. Parkka, M. Ermes, P. Korpipaa, J. Mantyjarvi, J. Peltola, and I. Korhonen, "Activity classification using realistic data from wearable sensors," IEEE Transactions on Information Technology in Biomedicine, vol. 10, no. 1, pp. 119–128, 2006.

[6] D. Rodrı'guez-Mart´ın, A. Sama, C. P ` erez-L ´ opez, and A. Catal ´ a,` "Identification of sit-to-stand and stand-to-sit transitions using a single inertial sensor." Studies in health technology and informatics, vol. 177, pp. 113–7, 2012

[7] B. Nafaji et al., Ambulatory System for Human Motion Analysis Using a Kinematic Sensor: Monitoring of Daily Physical Activity in the Elderly, IEEE Transactions on Biomedical Engineering 50 (2003), 711–723.

[8] R. T. Olszewski, C. Faloutsos, and D. B. Dot, Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data. 2001.

[9]https://towardsdatascience.com/a-step-by-step-explanation-of-principal-component-analysis-b836fb9c97e2

[10] Physical Human Activity Recognition Using Wearable Sensors Ferhat Attal 1 , Samer Mohammed 1,*, Mariam Dedabrishvili 1 , Faicel Chamroukhi 2 , Latifa Oukhellou 3 and Yacine Amirat 1 Received: 11 September 2015; Accepted: 8 December 2015; Published: 11 December 2015 Academic Editor: Vittorio M.N. Passaro

[11] Rodríguez-Martín, Daniel & Samà Monsonís, Albert & Pérez, Carlos & Català, Andreu & Cabestany, Joan. (2018). Posture transition analysis with barometers: contribution to accelerometer-based algorithms. Neural Computing and Applications. 10.1007/s00521-018-3759-8.

[12] Aminian K, Robert Ph, Buchser E, Rutschmann B, Hayoz D, Depairon M (1999) Physical activity monitoring based on accelerometry: validation and comparison with video observation. Med Biol Eng Comput 37:304–308

[13] Callaway, Andrew & Cobb, Jon. (2012). Linear Acceleration Measurement Utilizing Inter-Instrument Synchronization: A Comparison between Accelerometers and Motion-Based Tracking Approaches. Measurement in Physical Education and Exercise Science. 16. 151-163. 10.1080/1091367X.2012.669336.

[14] Maguire D., Frisby R., "Comparison of feature classification algorithms for activity recognition based on accelerometer and heart rate data," in 9th IT and T Conference in Dublin Institute of Technology, 2009.

[15] https://medium.com/@srjoglekar246/overfitting-and-human-behavior-5186df1e7d19

[16] Kira, Kenji and Rendell, Larry (1992). The Feature Selection Problem: Traditional Methods and a New Algorithm. AAAI-92 Proceedings.

falls among elderly persons living in the community, N. Eng. J. Med., vol. 319, pp. 1701–1707.
[3] B. Najafi, K. Aminian, F. Loew, Y. Blanc, and P. A. Robert, 2002, Measurement of Stand–Sit and Sit– Stand Transitions Using a Miniature Gyroscope and Its Application in Fall Risk Evaluation in the Elderly, IEEE Trans. on Biomedical Engineering, Vol. 49, No. 8, pp. 843 – 851.
[4] M. E. Tinetti, T. F. Williams, and R. Mayewski, 1986, Fall risk index for elderly patients based on number of chronic disabilities," Amer. J. Med., vol. 80, pp. 429– 434.
[5] J. Parkka, M. Ermes, P. Korpipaa, J. Mantyjarvi, J. Peltola, and I. Korhonen, "Activity classification using realistic data from wearable sensors," IEEE Transactions on Information Technology in Biomedicine, vol. 10, no. 1, pp. 119–128, 2006.

[6] D. Rodrı'guez-Mart´ın, A. Sama, C. P ` erez-L ´ opez, and A. Catal ´ a,` "Identification of sit-to-stand and stand-to-sit transitions using a single inertial sensor." Studies in health technology and informatics, vol. 177, pp. 113–7, 2012

[7] B. Nafaji et al., Ambulatory System for Human Motion Analysis Using a Kinematic Sensor: Monitoring of Daily Physical Activity in the Elderly, IEEE Transactions on Biomedical Engineering 50 (2003), 711–723.
[8] R. T. Olszewski, C. Faloutsos, and D. B. Dot, Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data. 2001.
[9]https://towardsdatascience.com/a-step-by-step-explanation-of-principal-component-analysis-b836fb9c97e2

[10] Physical Human Activity Recognition Using Wearable Sensors Ferhat Attal 1 , Samer Mohammed 1,*, Mariam Dedabrishvili 1 , Faicel Chamroukhi 2 , Latifa Oukhellou 3 and Yacine Amirat 1 Received: 11 September 2015; Accepted: 8 December 2015; Published: 11

December 2015 Academic Editor: Vittorio M.N. Passaro

[11] Rodríguez-Martín, Daniel & Samà Monsonís, Albert & Pérez, Carlos & Català, Andreu & Cabestany, Joan. (2018). Posture transition analysis with barometers: contribution to accelerometer-based algorithms. Neural Computing and Applications. 10.1007/s00521-018-3759-8.

[12] Aminian K, Robert Ph, Buchser E, Rutschmann B, Hayoz D, Depairon M (1999) Physical activity monitoring based on accelerometry: validation and comparison with video observation. Med Biol Eng Comput 37:304–308

[13] Callaway, Andrew & Cobb, Jon. (2012). Linear Acceleration Measurement Utilizing Inter-Instrument Synchronization: A Comparison between Accelerometers and Motion-Based Tracking Approaches. Measurement in Physical Education and Exercise Science. 16. 151-163. 10.1080/1091367X.2012.669336.

[14] Maguire D., Frisby R., “Comparison of feature classification algorithms for activity recognition based on accelerometer and heart rate data,” in 9th IT and T Conference in Dublin Institute of Technology, 2009.

[15] https://medium.com/@srjoglekar246/overfitting-and-human-behavior-5186df1e7d19

[16] Kira, Kenji and Rendell, Larry (1992). The Feature Selection Problem: Traditional Methods and a New Algorithm. AAAI-92 Proceedings.