International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 12 Issue 9, September-2023, Impact Factor: 7.957

Human Activity Recognition Using Smartphone

Amrita Kumari[1], Prof. Adrija Bhattacharya[2]

[1]Master of Technology in Computer Science and Engineering, Heritage Institute of Technology, Kolkata
[2]Asst. Prof. Computer Science and Engineering, Heritage Institute of Technology, Choubaga road, Anandpur, Kolkata

INTRODUCTION

Attention deficit and hyperactivity disorder (ADHD) is a neurodevelopmental condition that affects, among other things, the movement patterns of children suffering from hyperactivity and impulsive behaviours, major symptoms characterizing ADHD, result not only in differences in the activity levels but also in the activity patterns themselves. Attention deficit and hyperactivity disorder (ADHD) is a neurodevelopmental disorder affecting between 3 and 5% of children of school age. ADHD symptoms such as inattention, impulsivity and hyperactivity profoundly affect the intellectual, emotional and social behaviour of children suffering it and have a relevant impact in their activity and movement pattern.

This movement pattern is tracked using a **tri-axial accelerometer** and a **gyroscope** is attached with the sensors. These sensors are available in two form wearable and non-wearable form. All the children are given these sensors to wear 24 hours. These sensors track each and every movement (Sitting, Walking, Moving downstairs, moving upstairs etc.) of the children.

Human activity recognition
HAR for short is a broad field of study concerned with identifying the specific movement or action of a person based on sensor data. Movements are often typical activities performed indoors, such as walking, talking, standing, and sitting. They may also be more focused activities such as those types of activities performed in a kitchen or on a factory floor.

The sensor data may be remotely recorded, such as video, radar, or other wireless methods. Alternately, data may be recorded directly on the subject such as by carrying custom hardware or smart phones that have accelerometers and gyroscopes.

Sensor-based activity recognition seeks the profound high-level knowledge about human activities from multitudes of low-level sensor readings

Historically, sensor data for activity recognition was challenging and expensive to collect, requiring custom hardware. Now smart phones and other personal tracking devices used for fitness and health monitoring are cheap and ubiquitous. As such, sensor data from these devices is cheaper to collect, more common, and therefore is a more commonly studied version of the general activity recognition problem.

The problem is to predict the activity given a snapshot of sensor data, typically data from one or a small number of sensor types.

It is a challenging problem as there are no obvious or direct ways to relate the recorded sensor data to specific human activities and each subject may perform an activity with significant variation, resulting in variations in the recorded sensor data.

The intent is to record sensor data and corresponding activities for specific subjects, fit a model from this data, and generalize the model to classify the activity of new unseen subjects from their sensor data.

Identifying ADHD children
Attention deficit and hyperactivity disorder (ADHD) is a neurodevelopmental condition that affects, among other things, the movement patterns of children suffering from hyperactivity. In ADHD children several disorders are noticed which is not observed in the person who is not suffering from ADHD. Few disorders are as below:

Co-Existing Disorder	Children with ADHD	Children without ADHD
Learning Disability	45%	5%
Conduct Disorder	27%	2%

Anxiety	18%	2%
Depression	15%	1.1%
Speech Problem	12%	3%

Classifying the ADHD category:
Age Category:

Age (in Yrs.)	Chances of ADHD
4-10	68%
11-14	11.40%
15-17	10.20%

Level of ADHD:

Average Age (in Yrs.)	level of ADHD
7	Mild
6.1	Moderate
4.4	Severe

Region wise:

Region	level of ADHD
White	8.7%
Black(African)	9.8%
Hispanic	5%

RELATED WORK

Providing accurate and opportune information on people's activities and behaviours is one of the most important tasks in pervasive computing. Innumerable applications can be visualized, for instance, in medical, security, entertainment, and actual scenarios. Despite human activity recognition (HAR)being an active field for more than a decade, there are still key aspects that, if addressed, would constitute a significant turn in the way people interact with mobile devices. This paper surveys the state of the art in HAR based on wearable sensors. We also propose a two-level taxonomy in accordance to the learning approach (either Supervised or semi-supervised) and the response time (either offline or online). Twenty eight systems are qualitatively evaluated in terms of recognition performance, energy consumption, obtrusiveness, and flexibility, among others.

Human Activity Recognition is an ability to interpret human body gesture or motions via sensors and determine the human activity. HAR can be supervised or unsupervised. HAR is also considered an important component in various scientific research contexts i.e. surveillance, health care and human computer interaction.

Surveillance System
In surveillance context, HAR was adopted in surveillance systems, installed at public places like banks, airports etc. It is also used for preventing crimes and dangerous activity by recognising human activity.

Healthcare
As above we have seen HAR has proved helpful in surveillance activity. Similarly, it has proved beneficial in the Healthcare system. It has been installed in residential environment, hospitals and rehabilitation centre. HAR is used at rehabilitation centre to identify the activity of the elderly people their chronic related disease and any other disease. HAR is also used to encourage the physical activities in the rehabilitation centre for the disable d children like motor disabilities, stroke patients, patients with dysfunction etc. It is also used to monitor the behaviours of the stereotypical motion conditions in the children, abnormal cardiac patients. Etc. . . . Thus, HAR has proved beneficial in monitoring the health of the people.

Human Computer Interaction
HAR is used quite commonly in gaming and full body motion based games for older adults and adults with neurological injury. Through HAR, human body gestures are used to instruct the gaming technology i.e. simple gestures are used to interact with the games Pattern recognition approaches to accelerometer data processing have emerged as viable alternatives to cut-point methods. However, few studies have explored the validity of pattern recognition approaches in pre-schoolers; and none have compared supervised learning algorithms trained on hip

and wrist data. To develop, test, and compare activity class recognition algorithms trained on hip, wrist, and combined hip and wrist accelerometer data in pre-schoolers. We test these features on ADHD children.

Attention deficit and hyperactivity disorder (ADHD) is a neurodevelopmental condition that affects, among other things, the movement patterns of children suffering it. Inattention, hyperactivity and impulsive behaviours, major symptoms characterizing ADHD, result not only in differences in the activity levels but also in the activity patterns themselves. This paper proposes and trains a Recurrent Neural Network (RNN) to characterize the moment patterns for normally developing children and uses the trained RNN in order to assess differences in the movement patterns from children with ADHD. Each child is monitored for 24 consecutive hours, in a normal school day, wearing 4 tri-axial accelerometers (one at each wrist and ankle). The results for both medicated and non-medicated children with ADHD and for different activity levels are presented. While the movement patterns for non-medicated ADHD diagnosed participants showed higher differences as compared to those of normally developing participants, those differences were only statistically significant for medium intensity movements. On the other hand, the medicated ADHD participants showed statistically different behaviour for low intensity movements.

Some of the activity recognition work also focuses on the use of multiple accelerometer and possibly other sensors developed an automatic physical activities recognition system in a controlled environment using accelerometer.

Sensing technology

HAR plays an important role in recognizing the human activity. The below figure depicts how the sensor works in recognizing the human activity by recognition gesture. Different technology are used to recognize the human activity like RGB camera based, depth -sensor based and wearable -based sensor.

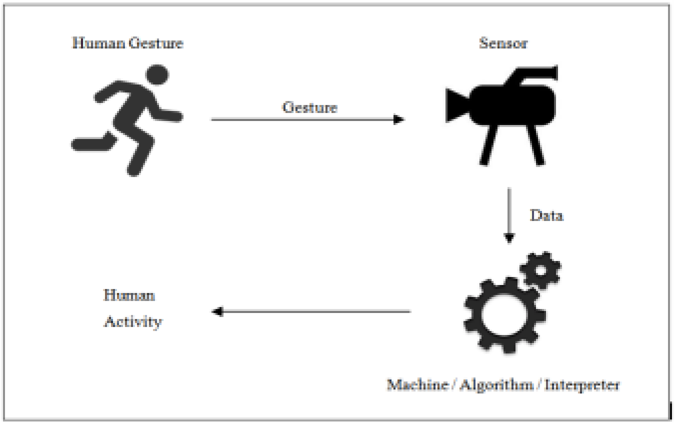


Fig. 1. General structure of HAR system

Researches depict that for the recognizing the activity mostly used sensor is depth -sensor based and wearable -based sensor whereas RGB camera based technique has obtained less emphasis. Thus, we have used wearable -based sensor as our human activity recognizing. The data obtained is analysed via different algorithm like Support Vector Machine.

HAR using wearable-based requires single or multiple sensors to be attached to the human body. Most commonly used sensor includes 3D-axial accelerometer, magnetometer, and gyroscope. With the latest smart phone technologies with inbuilt accelerometer and gyroscope is being used as the recognizing the human activity. A physical human activity can be identified easily through analysing the data generated from various wearable sensing after being process and determine by classification algorithm.

Wearable device is also be used to monitor the Human activities. Thus, we are using wearable sensors devices for collecting the acceleration data for human activity recognition for obtaining 94% accuracy. Users are tri axial accelerometer on different parts of their body i.e. in right Hip , Wrist, Upper Arm, Ankle and Thigh so that their activities can be tracked on mainly the six different activities like Sitting, Standing, Walking, Walking Upstairs , Walking Downstairs and Lying. The result obtained from the sensor of the thigh was identified to be most powerful.

Three different settings were used for identifying the dataset:

1. Data collected for a single subject over different days.
2. Data collected for a multiple subject over different days.
3. Data collected for a single subject and multiple subjects for the training and testing data.

Data Collected from the of 30 different users each having android phones in their pockets as they performed any of the six activities i.e. Sitting, Standing, Walking, Walking Upstairs , Walking Downstairs and Lying were recorded. Different learning algorithms were used to study the data like Support vector Machine, Logistic Regression, Import Vector Machine etc. Accordingly at last we got the overall accuracy above 90% .

DATA COLLECTION

Data is collected from the sensors wear/carried (in case of non –wearable sensors) by the particular person. Nowadays, data collection has become easy using Smartphone. Presently, all the Smartphone are having inbuilt accelerometer and gyroscope which is used to track each and every movement of the person and records the data in its memory. Thus, we may transfer these data and use it to analyse the human activity. As such, sensor data from these devices is cheaper to collect, more common, and therefore is a more commonly studied version of the general activity recognition problem.

Accelerometer is used to record gravitational acceleration and vibration in the device. The data is recorded with the proper timestamp which can be used to track details if required in future.

Gyroscope monitors and control device Positions, orientation, direction, angular motion and rotation of the device. It is also used for gesture recognition.

Further, Support Vector Machine is used to analyse the data collected from above sensors.

Dataset and Inputs:
In order to achieve high performance in predictions with this model, a data collection was done in collaboration with 30 volunteers were asked to perform a protocol of activities. All participants were wearing a Smartphone on the waist during the experiments' execution. During each experiment **3-axial acceleration signals and 3-axial angular velocities signals** were captured using Smartphone sensors. These raw inertial signals were stored in **Raw Data** (folder).

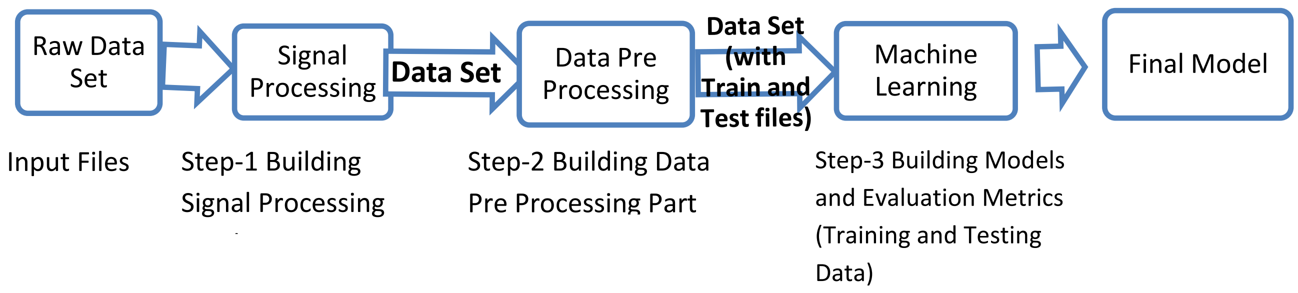
This folder includes 61 experiments each experiment has two log files:

1. **'Raw Data/acc_expXX_userYY.txt':** The raw tri-axial acceleration signal for the experiment number XX and associated to the user number YY. Every row is one acceleration sample (three axes) captured at a frequency of 50Hz. Units used for acceleration are 'gs.
2. **'Raw Data/gyro_expXX_userYY.txt'**: The raw tri-axial angular speed signal for the experiment number XX and associated to the user number YY. Every row is one angular velocity sample (three axes) captured at a frequency of 50Hz. Units used for angular velocities are radian per second (rad/s).

This folder includes also 1 additional file: labels file:

1. **'Raw Data/labels.txt':** includes all activity labels available for the dataset (1 activity label per row). Each row contains the start-end row numbers related a group of log samples located in acc and gyro files. These files can be identified using the **exp Id** and the **user Id** existing in that row.

Problem identification
Raw **Inertial Signals** mentioned above cannot be fed directly to machine learning models. A signal processing pipeline should be built to filter noise, extract useful signals. Split them into windows. From each window a vector of features will be generated to obtain classical datasets.



After exploring raw inertial signals, 3rd order median filter is used to reduce noise. Then a Fast Fourier Transform is applied to extract useful frequency components: **body components and gravity components.** AnInverse Fast Fourier Transform has also been applied to useful components to obtain them in time domain. All Body components will be derived in time domain to obtain jerk signals. The magnitude of each triaxialsignalhas been generated using the Euclidian Norm.

All useful have been sampled in **fixed-width sliding windows** of 2.56 sec (128 readings/window) using two methods. The first windowing method concerns only signal captures related to basic activities (activity id from 1 to 6). This method assures that all rows in a window will have the same activity id which could be considered as the window activity id (windows type I). The second one which is more realistic it concerns all captures in Raw Data (activity ids from 1 to 6). Actually, this windowing method doesn't care if all rows in a window have even an activity id or the same activity id as a result, a voting function should be created to define the activity id of each window (Windows type II).

A **Fast Fourier Transform** has been applied to some columns of each window, to obtain frequency windows. From each tuple of time and frequency windows, a vector of features will be generated by calculating variables from the time and frequency domain signals. Each features vector is a row in the final dataset.

Signal Processing

Signal processing is a technique that focuses on analysing, modifying, and synthesizing signals such as sound, images, and scientific measurements. These techniques can be also used to improve transmission, storage efficiency and subjective quality and to emphasize or detect components of interest in a measured signal. Now we will refine the Signals (raw data) obtained from the sensors (Accelerometer and Gyroscope). Thus, Refined Signal (processed data) will be used further for Machine Processing through which the accuracy of the data will be predicted.

Steps Involved in Signal Processing:

1. Importing Raw Data.
    a) Importing Accelerometer and Gyroscope Data
    b) Importing and Storing Data frames in Dictionary
    c) Define Labels functions and store it in the data frame.
    d) Define Activity involved and store it in activity data frame.
2. Analysing the data
    a) Counting the number of Rows available for each user.
    b) Separating the Useful number of rows per user.
    c) Counting the number of rows per activity.
    d) Calculate mean time taken per activity.
    e) Created a function to visualize the tri axial signal along with the accelerometer and gyroscope signals generated for the users.
    f) Created a lookup function for the label data frame created above.
    g) Created a function to visualize the tri axial signal of accelerometer and gyroscope signals generated for each activity.
3. Generation of the Time-Domain signals.
    a) Created Median Filter function and applied it in the signal and visualise it.
    b) Defined gravity, jerking and magnitude function.
    c) Generated the time domain signal using Fast Fourier Transform technique.
4. Creation for windows for analysing the signal.
5. Created a Fast Fourier Transform function and applied in the sample.

Importing Raw Data.

Importing Accelerometer and Gyroscope Data

The accelerometer and gyroscope data is obtained for 30 users which are bifurcated into 30 different text files so that all the users' data is analysed. These raw signals are imported and then converted into float format and a data frame is created for each user which contains both the accelerometer and gyroscope data.

Algorithm:

Step-1 Defined the path of the accelerometer and gyroscope data.
Step-2 Counted the no. of accelerometer and gyroscope file.
Step-3 Created a function for importing the raw data.
Step-4 Accelerometer and gyroscope data are appended in a single File.
Step-5 Converted the data in the files into the float type variable store it the data frame.
Step-6 Printed the data frame created above.

Labels functions and store it in the data frame.

Label file is created as an index for the accelerometer and gyroscope signal.

Algorithm:
Step-1 Created a function for importing the label containing the complete details of the data.
Step-2 appended all data in a single File.
Step-3 Converted the data in the files into the integer type variable store it the data frame.
Step-3 Named the Columns so that it can be identified distinctly.
Step-4 Printed the label data frame.

Activity label
Activity label is created defining the 6 activity observed by the user.

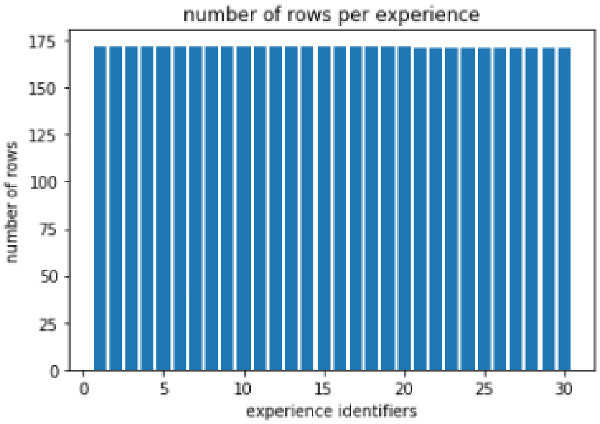
1. Walking.
2. Walking Upstairs.
3. Walking Downstairs.
4. Sitting.
5. Standing.
6. Lying.

Analysing raw data
Useful Rows
Rows having actually an activity ID some rows were captured while the user was not performing the activity protocol these rows considered as not useful in this part. Other rows are considered useful.
Data is analysed in two ways

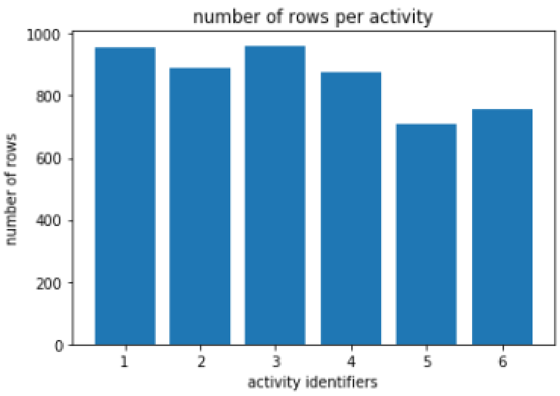
1. User Wise
2. Activity Wise

User Wise

The number of rows per experience and useful rows are plotted



Activity Wise
Number of Rows per activity



Visualization of Tri-axial Signal
Algorithm:
Step-1 Defined a function Visualize tri-axial signal
Step-2 Data stored in the dictionary is in the form of expXX_userYY. This data is stored is used as a key in the function.
Step-3 Experiment ID and User ID are stored in the respective variable.

Step-4 an if-else loop is used through which the data is stored in a new data frame according to the activity of the user. If all the activity considered then all the data is copied to new data frame else only data corresponding to the particular activity is copied to the data frame according to their starting and ending point.
Step-5 Then signal type is compared whether the data belong to accelerometer signal or gyroscope.
Step-6 Then the columns are created and graph is plotted.

Median Filter
The **Median Filter** is a non-linear digital filtering technique, often used to remove noise from an image or signal. Such noise reduction is a typical pre-processing step to improve the results of later processing (for example, edge detection on an image). Median filtering is very widely used in digital image processing because, under certain conditions, it preserves edges while removing noise.

Algorithm for Median Filter
Step-1 Defined a function Median Filter.
Step-2 Passed a signal in form of array in the function.
Step-3 Applied filter to the signal with the kernel size, by default is 3.
Step-4 Plotted the filtered signal.

Fast Fourier Transform:
A **fast Fourier transform** (**FFT**) is an algorithm that computes the discrete **Fourier transform** (DFT) of a sequence, or its inverse (IDFT). **Fourier** analysis converts a signal from its original domain (often time or space) to a representation in the frequency domain and vice versa.

Algorithm:
Step-1 Defined a function Components Selection one signal.
Step-2 Passed a time signal along with frequency in the function.
Step-3 Created parts of signal i.e. DC, noise and body component of signal is created according to their frequency. Frequency of signal ranging from -0.3 to 0.3 Hz are included in DC (Direct Current) component. For noise component from -25 to 20hZ and 20hZ to25hZ in which -25 and 25 Hz in included but -20 and 20hZ not included. For Body Component -20 to - 0.3 Hz and 0.3 Hz to 20 Hz in which -0.3 and 0.3 Hz not included but -20 and 20 Hz included.
Step-4 Total Component is generated after removing the noise component from the signal.
Step-5 Return the Total Component.

Data Pre-Processing
Handling Outlier:
Outliers are defined as samples that are significantly different from the remaining data. Those are points that lie outside the overall pattern of the distribution.

The data is pre-processed by calculating the threshold value of the data. Outlier's row is calculated from the complete data frame. The rows whose features exceed the threshold value are considered as outlier row. A new data frame is created which have the data after removing the outlier data. Thus, this data frame is considered as clean data frame.

Feature Scaling:
In data pre-processing a data normalization step is performed known as Feature **scaling.** It is a method used to normalize the range of independent variables or features of data.
There are three different methods for feature scaling:

1. **Rescaling (min-max normalization)**
2. **Mean normalization**
3. **Standardization (Z-score Normalization)**

Rescaling (min-max normalization)
Also known as min-max scaling or min-max normalization, is the simplest method and consists in rescaling the range of features to scale the range in [0, 1] or [−1, 1]. Selecting the target range depends on the nature of the data. The general formula for a min-max of [0, 1] is given as:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Where x is an original value, x' is the normalized value.
To rescale a range between an arbitrary set of values [a, b], the formula becomes:

$$x' = a + \frac{(x - \min(x))(b-a)}{\max(x) - \min(x)}$$

Where a, b are the min-max values.

Window Creation:
Step-1 A window type function is created.
Step 2 Columns are taken from the time signal dictionary which is normalized.
Step-3 A cursor variable is taken which is used for the endpoint and also the start point is taken. A window is created with width of 32 so that the signals in the particular window are too observed.
Step-4 Return the data generated from the window.

Solution Proposal
Machine Learning:
Machine learning is a model which is used to process the data for more refinement purpose after the data is fed into the model. Different techniques are used in the machine learning model like Support Vector Machine, Logistic Regression etc.for determining the correct accuracy of the model.

Support Vector Machine
Support Vector Machine (SVM) model is used for identifying the children suffering from ADHD. SVM model is used to classify the different category of the ADHD level in the children. On survey it has been notices that chances of boy being prone to ADHD is more than twice of girls.

Support Vector Machine is used for classification of the data gathered from the sensors. It separates the data after classification i.e. keeps a single type of the data together. The complete data is spitted into **training** data and **testing** data using the function **train_test_split ().** Train_test_split() function has a parameter **train_size** whose default value is 0.75 i.e. it splits data in ration 75:25 (75% data in training and remaining 25% as testing data).

Two type of the classifier:
1. Linear Classifier
2. Non-Linear Classifier.

Linear Classifier
It generates the hyper plane in a manner which classifies the data completely. The two hyper planes are also drawn as below In Fig A in such a way that maximum margin is achieved for the data. The points on the margin points as in fig B is known as Support Vector

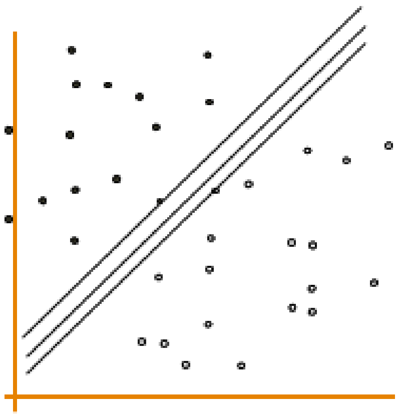


Fig a

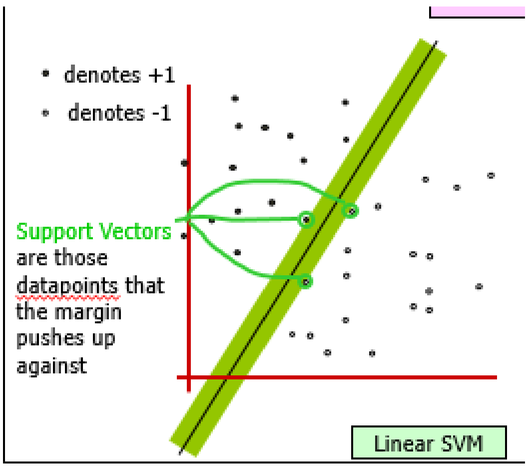


Fig B

Non-Linear Classifier
The data which cannot be classified linearly is classified Non-linearly using the **KERNEL** feature. Kernel is used to increase the dimension of the image file i.e. it converts the low dimensional data to high dimensional data. For example 2D data to 3D data as below thus now in 3D data has been classified easily.

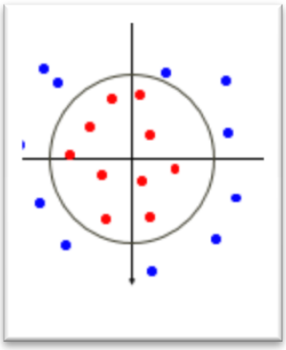


Fig a- 2D data

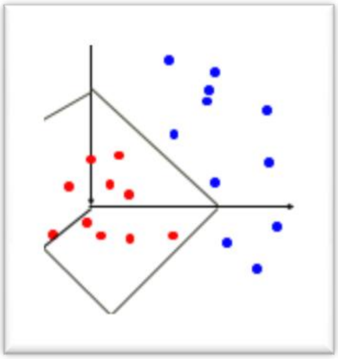


Fig B- 3D

Application of SVM

1. Text (and hypertext) categorization
2. Image classification
3. Bioinformatics (Protein classification,
4. Cancer classification)
5. Hand-written character recognition

Support Vector Controller

Support Vector Controller (SVC) under SVM is used to change the dimension of and coefficient of the data using the different parameter under SVC. The parameter is as below:

1. **C**-It is regularization parameter of error term.
2. **Kernel**-It specifies kernel type to be used by the algorithm. It can be “Linear”, “Poly”, “rbf”, “Sigmoid”. The default value of kernel is “rbf”.
3. **Degree**- It is degree of the polynomial kernel function (“poly”) and ignored for other kernel value. Default value is 3.
4. **Gamma**- It is the kernel coefficient for”rbf”, “poly” and “sigmoid”. If gamma is “auto” then 1/n features will be used.

Experimental Set Up

1. OS- 64 bit Windows 7
2. Memory- 4GB RAM , ROM-64GB
3. Database-SQL
4. IDE-**Pycharm**
5. Phytonversion-3.7.4

Coding

1. **Read_csv () -** It fetches the data from the CSV file and use it in the program.
2. **Train_test_split() -** It splits the data into **training** data and **testing** data. This function has a parameter **train_size** whose default value is 0.75 i.e. it splits data in ration 75:25 (75% data in training and remaining 25% as testing data).
3. **Preprocessing.LabelEncoder()-**it formats the data from one data type to other.
4. **Fit () -** It is used to properly accommodate all the data in the proper way to avoid over fitting.
5. **Accuracy score () -** It is used to calculate the accuracy of the data collected.

Splitting the train and test Data

For analysing the accuracy of the data we need to divide the complete data in two parts i.e. training and testing data. It can be done using the train test split function of the SVM. By default the training data is 75% and testing data is 25%. If we need to change splitting criteria then we need to mention it explicitly. Accuracy of the data varies according to their splitting criteria of the training and testing data. Through Logistic regression process the accuracy of the data is calculated and confusion matrix is generated.

Logistic Regression

Logistic regression (LR) is a well-known statistical modelfor binary classification. It is known as one of the generalizedlinear models.Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail which is represented by an indicator variable, where the two values are labelled "0" and "1". In the logistic model, the log-odds (the logarithm of the odds) for the value labelled "1" is a linear combination of one or more independent variables ("predictors"); the independent variables can each be a binary variable (two classes, coded by an indicator variable) or a continuous variable (any real value). The corresponding probability of the value labelled "1" can vary between 0 (certainly the value "0") and 1 (certainly the value "1"), hence the labelling; the function that converts log-odds to probability is the logistic function, hence the name.

The unit of measurement for the log-odds scale is called a **logit**, from logistic unit, hence the alternative names. Analogous models with a different **sigmoid function** instead of the logistic function can also be used, such as the probity model; the defining characteristic of the logistic model is that increasing one of the independent variables multiplicatively scales the odds of the given outcome at a constant rate, with each independent variable having its own parameter; for a binary dependent variable this generalizes the odds ratio.

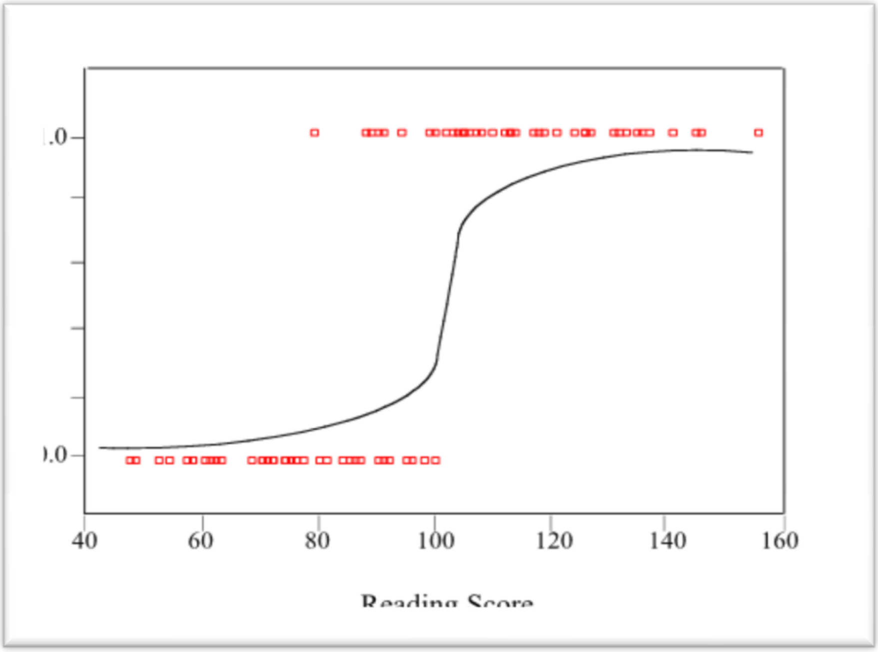
The simple logistic model has the form

Logit(Y) = natural log (odds) = ln (π/1-π) =α+βX
Taking the antilog of Equation 1 on both sides, one derives an equation to predict the probability of the occurrence of the outcome of interests follows:

π = Probability(Y = outcome of interest | X = as specific value of X) = e^ (α+βX)/ (1+ e^ (α+βX).

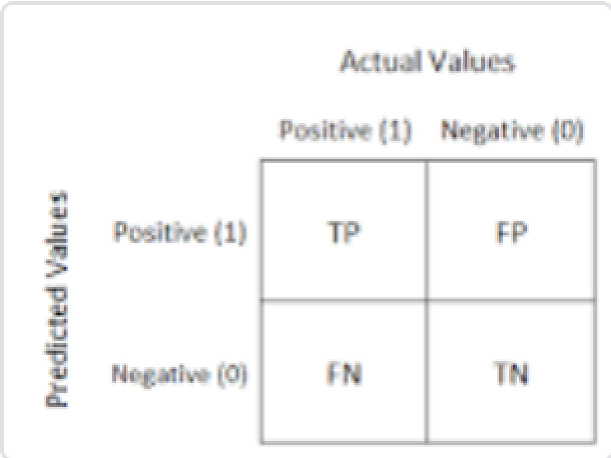
Where
π - Probability of the outcome of interest or event.
α - Y intercept
β- Regression coefficient
Extending the logic of the simple logistic regression to multiple predictors (say X1 = reading core and X2 = gender), one can construct a complex logistic regression for Y (recommendation for remedial reading programs) as follows:

Logit(Y) =ln (π/1-π) =α+β1X+β2X
Therefore,
π = Probability(Y = outcome of interest | X = x, a specific value of X) = e^ (α+β1X+ β2X)/ (1+ e^ (α+β1X+ β2X).



Confusion Matrix
A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known.

1. TP-True Positive-When it's actually yes, how often does it predict yes?
2. FP- False Positive-When it's actually no, how often does it predict yes?
3. FN- False Negative- When it's actually no, how often does it predict no?
4. TN-True Negative-When it's actually no, how often does it predict no?



RESULT

As mentioned above in splitting of training and testing data we have observed how the accuracy of varies with the size variance of the training and testing data. The detailed observation is shared in the below table. Also a confusion matrix is also generated along with the accuracy as we are observing for the six activities i.e. Sitting, Standing, Walking, Walking Upstairs, Walking Downstairs and Lying. Therefore, a 6*6 matrix is generated which is called confusion matrix.

S No	train size	test size	accuracy
1	70	30	89.379
2	76	24	89.189
3	66	34	89.165
4	80	20	90.697
5	86	14	90.547

Result for 80% training data and 20% testing data

________LogisticRegression results:__________

Accuracy and duration per training size

	10% of train	50% of train	100% of train
train_time	0.261597	2.020670	5.005649
pred_time	0.008277	0.007028	0.007917
acc_train	0.607667	0.890333	0.917000
acc_test	0.401993	0.860465	0.906977

Confusion Matrix Sensitivity and Recall when 100% of train is achieved

	ST	SI	LD	WK	WD	WU	data points number	precision %	sensitivity %	specificity %
ST	171	2	0	0	0	0	173	100	98.8439	100
SI	0	117	13	0	0	0	130	82.3944	90	96.7658
LD	0	19	132	4	0	0	155	80.4878	85.1613	95.7219
WK	0	3	12	146	8	0	169	93.5897	86.3905	98.6376
WD	0	1	7	6	107	6	127	90.678	84.252	98.5825
WU	0	0	0	0	3	146	149	96.0526	97.9866	99.2042
Total							data points number=903			accuracy= 90.697%

Thus, from our research we have obtained the above data also the data collected has been 90.697% accurate. The 6 different activities of the people and sample of 30 users have been taken as reference was used for the detailed study for tracking the activity of the human. The sample has proved beneficial for tracking the human activity.

CONCLUSION

Firstly, we built the signal processing pipeline from scratch to produce classical datasets where **LogisticRegression** was applied to verify if the signal processing steps whether it was implemented correctly or not. After that we found a way to delete outliers, and scaling features, to improve the model's performance which will be considered as benchmark results.

When we proposed this project, we had ideas of using tweaking models' parameters as an easy task compared to signal processing steps but after implementation it was not so. As we proceed through the machine learning part. Accuracy increased by 10-20% after using signal processing.

Now we have a better understanding of the supervised learning approach, signal processing and activity recognition, parameters tuning and definitely, we also look forward for doing more projects in Activity Recognition using other approaches.

Free Form Visualization:

	Activity 1	Activity 2	Activity 3	Activity 4	Activity 5	Activity 6
count	30.000000	30.000000	30.00000	30.000000	30.000000	30.000000
mean	25.866667	23.666667	25.50000	28.600000	21.800000	24.500000
std	4.384403	3.565479	2.19325	3.756374	3.818286	2.909556
min	19.000000	16.000000	22.00000	22.000000	13.000000	18.000000
25%	22.250000	22.000000	24.00000	25.250000	20.000000	22.250000
50%	25.000000	23.500000	25.50000	28.500000	22.500000	25.000000
75%	30.500000	26.000000	27.00000	31.750000	24.000000	26.000000
max	33.000000	31.000000	28.00000	35.000000	28.000000	31.000000

The table above represents Summary of all activity along with different functions on the dataset.

FUTURE SCOPE

The project presented above (signal processing pipeline+ final models) could be used as an online HAR tool. By modifying necessary parts in the signal processing pipeline related to targets. In real predictions inputs will be log files of acceleration and gyroscope without labels, the online signal processing part should be adapted to produce final features correctly and smoothly. To improve predictions 'accuracy, we propose to use the final model which can predict basic activities with a high accuracy.

Postural transitions types could be deducted automatically from the previous and the next predictions around it. Since in online mode all obtained rows are chronologically ordered. For example, if the previous state is sitting (row X) and the next state is standing (row Y) with an acceptable time difference between X and Y. All rows' features between them and predicted as postural transitions using final model and including postural transition type "From Sit to Stand".

REFERENCES

[1]. Jie Yin, Qiang Yang Senior Member, IEEE, and Jeffrey Junfeng Pan. Sensor-Based Abnormal Human-Activity Detection, IEEE Transactions on Data Knowledge and Data Engineering, Vol 20, No.8, August 2008.

[2]. A.P. Bradley, "The Use of the Area under the ROC Curve in the Evaluation of Machine Learning Algorithms," Pattern Recognition, vol. 30, pp. 1145-1159, 1997.

[3]. Oscar D. Lara and Miguel A. Labrador, A Survey on Human Activity Recognition using Wearable Sensors IEEE Communications Survey & Tutorials Vol. 15 No3, Third Quarter 2013.

[4]. J.Candamo, M. Shreve, D. Goldgof, D. Sapper, and R. Kasturi, "Understanding transit scenes: A survey on human behaviour-recognitionalgorithms," IEEE Trans. Intell. Transp. Syst.,vol.11, no.1, pp.206–224, 2010.

[5]. D. Choujaa and N. Dulay, "Tracme: Temporal activity recognition using mobile phone data," in IEEE/IFIP International Conference onEmbedded and Ubiquitous Computing, vol. 1, pp. 119–126, 2008.

[6]. J. Parkka, M. Ermes, P. Korpipaa, J. Mantyjarvi, J. Peltola, andI. Korhonen, "Activity classification using realistic data from wearablesensors," IEEE Trans. Inf. Technol. Biomed., vol. 10, no. 1, pp. 119–128, 2006.

[7]. T. Brezmes, J.-L. Gorricho, and J. Cotrina, "Activity recognition fromaccelerometer data on a mobile phone," in Distributed Computing,Artificial Intelligence, Bioinformatics, Soft Computing, and AmbientAssisted Living, vol. 5518, pp. 796–799, Springer Berlin / Heidelberg,2009.
[8]. Nello Cristianini and John Shawe-Taylor, "An Introduction to Support Vector Machinesand Other Kernel-based Learning Methods", Cambridge University Press, 2000.
[9]. C. Cortes and V. Vapnik. Support vector networks. Machine Learning, 20:273 – 297, 1995.
[10]. A Iosifidis, A. Tefas, and I. Pitas, "Multi-view action recognition based on actionvolumes, fuzzy distances and cluster discriminant analysis," Signal Processing, vol. 3, no. 6, pp. 1445–1457, Jun. 2013
[11]. Penang, Malaysia, 2014 Human Activity Recognition: A Review.IEEE International Conference on Control System, Computing and Engineering, 28 - 30 November 2014.
[12]. Burges C.,"A tutorial on support vector machines for pattern recognition", In "Data Mining andKnowledge Discovery". Kluwer Academic Publishers, Boston, 1998,(Volume2).
[13]. O. D. Lara and M. a. Labrador, "A Survey on Human Activity Recognition using Wearable Sensors,"IEEE Commun. Surv. Tutorials, vol. 15, no. 3, pp. 1192–1209, Jan 2013.
[14]. W. Ong, L. Palafox, and T. Koseki, "Investigation of Feature Extraction forUnsupervised Learning in Human Activity Detection," Bull. Networking, Comput. Syst. Softw., vol. 2, no. 1, pp. 30–35, 2013.
[15]. J. Yang, J. Lee, and J. Choi, "Activity Recognition Based on RFID Object Usage for Smart Mobile Devices," J. Comput. Sci. Technol., vol. 26, no. 2, pp. 239–246, Mar. 2011.
[16]. N. Alshurafa, W. Xu, J. Liu, M.-C. Huang, B. Mortazavi, C. Roberts, and M. Sarrafzadeh, "Designing a Robust Activity Recognition Framework for Health and Exergaming using Wearable Sensors.," IEEE J. Biomed. Heal. informatics, no. c, pp. 1–11, Oct. 2013.
[17]. A. M. Khan, "Human Activity Recognition Using a Single Tri-axial Accelerometer," Kyung Hee University, Seoul, Korea, 2011.
[18]. E. Kantoch and P. Augustyniak, "Human activity surveillance based on wearable body sensor network," in Computing in Cardiology (CinC), 2012, pp. 325 – 328.
[19]. D. T. G. Huynh, "Human Activity Recognition with Wearable Sensors," Technics Universität Darmstadt, 2008.
[20]. A. Reiss, G. Hendeby, and D. Stricker, "A Competitive Approach for Human Activity Recognition on Smartphones," in 21st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, 2013, no. Esann, pp. 455–460.
[21]. Wenjian Wang, Liang Ma, An Estimation of Optimal Gaussian Kernel Parameter for Support Vector Classification, Part-II pp. 627-635, September 2008.
[22]. Christanini and Shawe Taylor J. 2000. An Introduction to support Vector machine and other kernel based learning methods, March 2000.
[23]. Vladmir Cherkassy, Yunqian Ma, Partial Selection of SVM parameters and noise estimation for SVM regression in Neural Network 17(1):113-26, February 2004.
[24]. Rimah Amami, Dorra Ben Ayed and Noureddine Ellouze, Partial Selection of SVM supervised parameters with different feature representations for Vowel Recognition, July 2015.
[25]. Chao-Ying, Joanne Peng, Kuk Lida Leegary M. Ingersoll. An Introduction to Logistic Regression Analysis and Reporting in Journal Of Educational Research 96(1):3- 14, September 2002.