*REPORT:*

Human activity recognition *with smart phone:*

*Many of* today’s *mobile devices are designed to automatically detect when we are engaging in a specific activity, such as running or cycling. This is a machine learning at work. To practice with this type of project, nvoice machine learning engineers use a dataset that contains fitness activity records for a few people (the more , the better) that was collect through mobile device equipped with inertial sensors. Learner can learn build classification models that will accurately predict future activities. This can also help them understand how to solve multi-classification problems.*

* *The dataset containing the different activities with a smartphone to the waists of 30 persons. The data is recorded with the help of sensors (accelerometer and Gyroscope) in that smartphone.*

*Importing dataset using pandas library:*

*The dataset contain the columns 'tBodyAcc-mean()-X', 'tBodyAcc-mean()-Y', 'tBodyAcc-mean()-Z',*

*'tBodyAcc-std()-X', 'tBodyAcc-std()-Y', 'tBodyAcc-std()-Z',*

*'tBodyAcc-mad()-X', 'tBodyAcc-mad()-Y', 'tBodyAcc-mad()-Z',*

*'tBodyAcc-max()-X',*

*...*

*'fBodyBodyGyroJerkMag-kurtosis()', 'angle(tBodyAccMean,gravity)',*

*'angle(tBodyAccJerkMean),gravityMean)',*

*'angle(tBodyGyroMean,gravityMean)',*

*'angle(tBodyGyroJerkMean,gravityMean)', 'angle(X,gravityMean)',*

*'angle(Y,gravityMean)', 'angle(Z,gravityMean)', 'subject', 'Activity'],*

*dtype='object', length=563)*

* *The data is recorded By using the sensors(Gyroscope and accelerometer) in a smartphone, they have captured '3-axial linear acceleration'(*tAcc-XYZ*) from accelerometer and '3-axial angular velocity' (*tGyro-XYZ*) from Gyroscope with several variations.*

*Data preprocessing:*

* *perform the cleaning and processing on the dataset.*
* *Data preprocessing is performed on the dataset.*

*Splitting the dataset:*

* *The readings from 70% of the volunteers were taken as trianing data and remaining 30% subjects recordings were taken for test data*
* 6 Activities:

1. Walking
2. WalkingUpstairs
3. WalkingDownstairs
4. Standing
5. Sitting
6. Lying.

* we will remove the commas and brackets to out features so that we can apply directly.
* save our data to csv file for future prediction .

Analysis:

* For the prediction there are 2 types of activities:
* Static and dynamic ,
* Static activity: The Static activity does not depend on motion information . those are:standing, sit, liedown
* Dynamic activity: The dynamic activities depends on motion information, those are: Walking, WalkingUpstairs,WalkingDownstairs.
* Ploting dist plot graph for the stationary and moving activities.
* Plot boxplot for the activities and Acceleration magnitude mean. prefix t means time.
* If tAccMean is < -0.8 then the Activities are either Standing or Sitting or Laying.
* If tAccMean is > -0.6 then the Activities are either Walking or WalkingDownstairs or WalkingUpstairs.
* If tAccMean > 0.0 then the Activity is WalkingDownstairs.
* We can classify 75% the Acitivity labels with some errors.
* Plot boxplot for the angleXgravityMean and the activity.
* In this angleXgravityMean of the lying activity is > 0 and for the other activities angleXgravityMean<0.
* Prove that standard neural networks fail to get the best results while implementing on sequential classification data. With exploratory data analysis done, we moved on to implementing machine learning models on the dataset. As the dataset is a 3-D matrix, we converted it to a (7352 X 1152) dataframe to implement these models.
* #importing standard neural network libraries

from sklearn.tree import DecisionTreeClassifier

from sklearn.neighbors import KNeighborsClassifier

from sklearn.svm import SVC

from sklearn.naive\_bayes import GaussianNB

from sklearn.discriminant\_analysis import QuadraticDiscriminantAnalysis

from sklearn.metrics import accuracy\_score # for evaluation

from sklearn.metrics import confusion\_matrix

Required modules:

* import itertools
* import numpy as np
* import matplotlib.pyplot as plt
* from sklearn.metrics import confusion\_matrix
* %matplotlib inline
* from IPython.display import Markdown, display
* from datetime import datetime
* for Logistic Regression:(modules need to import):
* from sklearn import linear\_model
* from sklearn import metrics
* from sklearn.model\_selection import GridSearchCV

Train the dataset:

* fit the model with x\_train and y\_train.

Test the data set(predict human activity):

* And test with the x\_test.

Conclusion:

* The model predicting the Human activity recognition with smart phone.