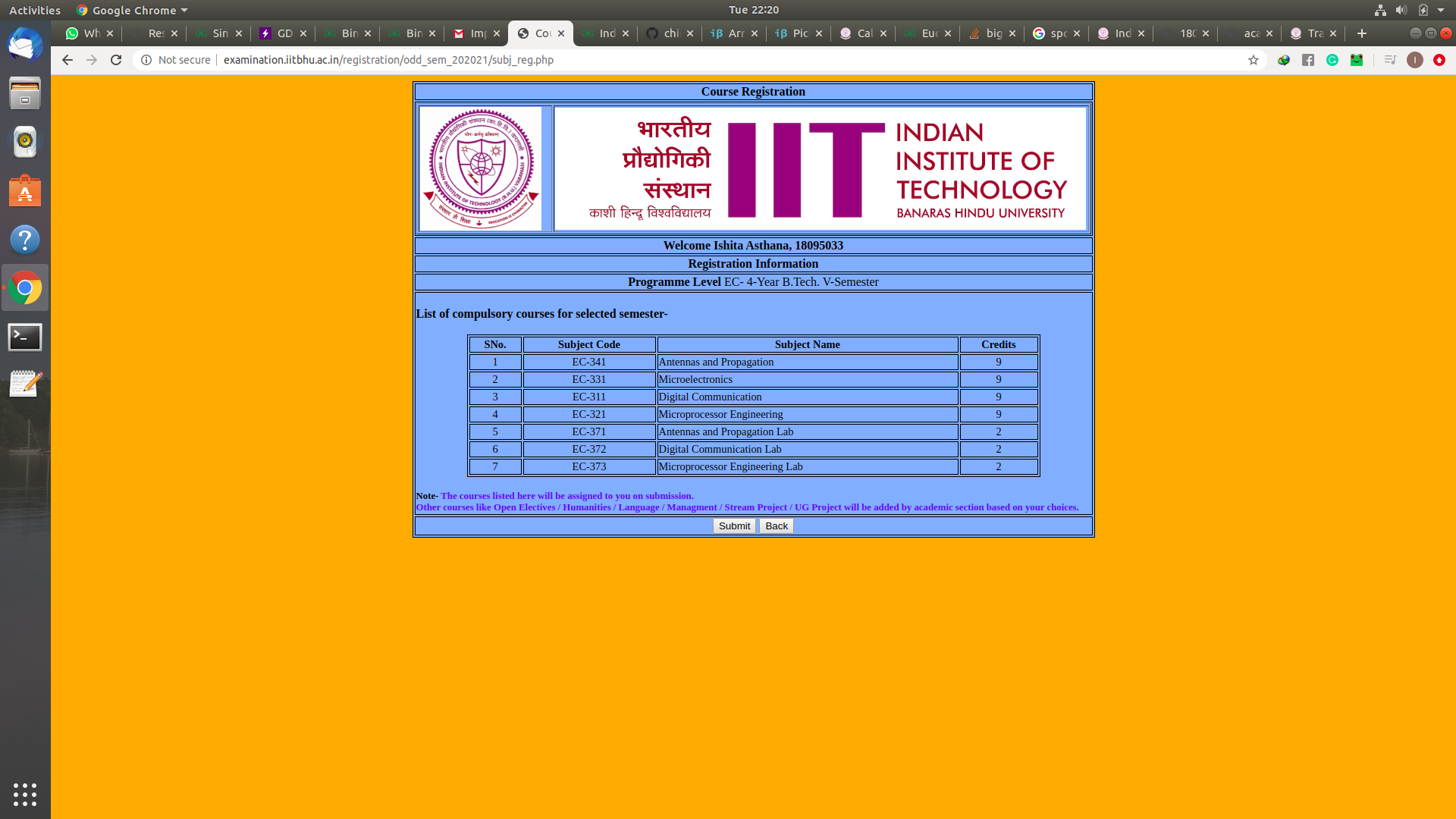
**Smartphone-Based Recognition of Human Activities and Postural Transitions using Federated Learning**



*Under the guidance of*

Dr. Hari Prabhat Gupta

Assistant professor

Computer science Department

IIT BHU

**Submitted by :**

Vaishnavi Tiwari 18045108

Pratiksha Pandey 18045067

Ishita Asthana 18095033

Vipin Sharma 18085080

**Acknowledgments**

We would like to express our special thanks of gratitude to Mr. Hari Prabhat Gupta, Assistant Professor, Department of Computer Science who gave us the opportunity to work on the project Smartphone-Based Recognition of Human Activities and Postural Transitions using Federated Learning, which also helped us to gain more knowledge on federated learning techniques. We are extremely thankful to him for providing us all the support and guidance required for the successful completion of the project.

**Contents**

1. Introduction
2. Problem Definition and Algorithm
3. Experimental Evaluation

3.1 Methodology

3.2 Result

1. Related Work/Future Work
2. Conclusion

**Introduction**

We have used the HAPT dataset from UCI Machine Learning Repository. This data set includes 3 static postures-standing, sitting, lying and 3 dynamic movements-walking, walking upstairs, walking downstairs. This also comprises 6 postural transitions which are stand-to-lie, lie-to-stand, lie-to-sit, sit-to-lie, stand-to-sit and sit-to-stand. The data is randomly divided into 2 sets - 70% training data and 30% test data. Here a total of 561 features are there.

Model used here is Support Vector Machine using linear Kernel

**Problem Definition and Algorithm**

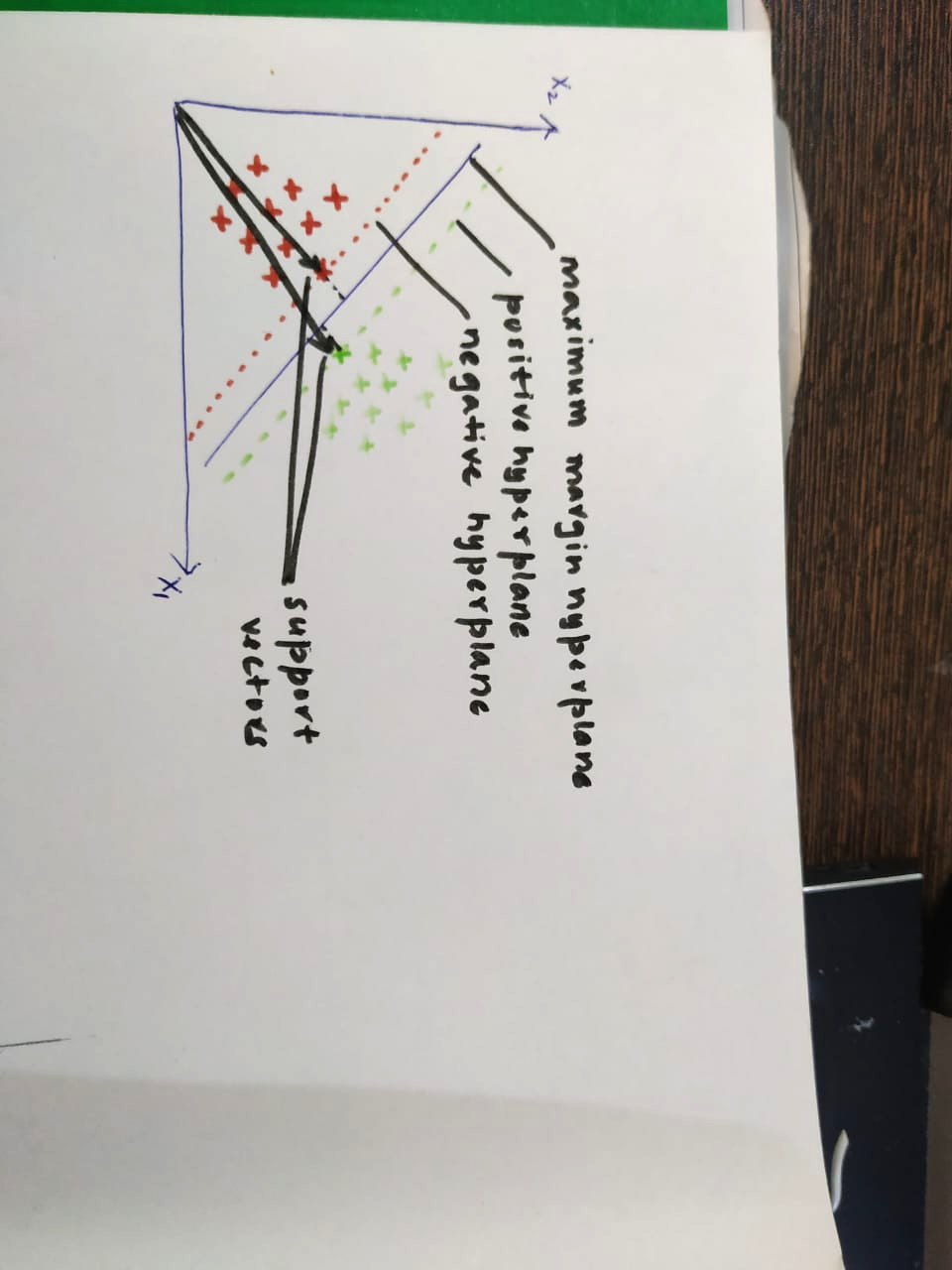
Human activity recognition systems capture the heterogeneous sensor signals of the accelerometer and gyroscope in a continuous fashion to comment on the state of the user and its environment.The most common method of gathering these data points is via the smartphone sensors which are held in pockets/hands and therefore give the readings in accordance to user’s body motion. This is of utmost importance in healthcare applications, for automatic and intelligent daily activity monitoring. The fitness apps leverage this technology to track the physical activities and fitness status of the user, sometimes also providing health/monetary benefits in exchange. This process demands accurate measurements. To build a robust and highly accurate model, federated learning techniques are used that combine the public dataset along with the encoded private datasets to access large amounts of data while preserving the privacy.

So the problem statement boils down to leveraging Federated Learning Framework for the HAPT dataset. This is a classification problem wherein we have to classify the body movements or activities into categories(3 static postures-standing, sitting, lying ; 3 dynamic movements - walking, walking upstairs, walking downstairs; 6 postural transitions- stand-to-lie, lie-to-stand, lie-to-sit, sit-to-lie, stand-to-sit and sit-to-stand).

The Algorithms proposed for training are Support Vector Machine(SVM) and Principal Component Analysis(PCA).

Support Vector Machines(SVM):

This is a machine learning algorithm for performing multi-class classification with less memory requirements and low processing time. It is also known as the ‘out-of-the-box’ algorithm due to its good performance on unseen or new data points which are on the boundary during classification(support vectors) due to their property of sharing characteristics with multiple classes.

****

In case of perfectly separable data:

1. SVM creates a hyperplane having the maximum value of margin or minimum value of perpendicular distance of closest observation from hyperplane.
2. The observations that fall on the margin are known as support vectors.
3. This can’t be used for classes that aren’t separable by hyperplane and experience a drastic shift in hyperplane in case of addition of more observations, making it highly prone to overfitting.

In case of non perfect separable data:

1. We allow some observations to be incorrectly classified or be on the wrong side of the hyperplane.
2. This is done in accordance to a misclassification budget(B) which is used to limit the sum of perpendicular distances of misclassified observations from hyperplane to be less than B.

X1+x2+x3+...xn < B

1. We aim to maximise the margin while being within the limits of the budget.
2. In practice, C(cost multiplier of error term) is used which is inversely proportional to B.

Impact of C:

1. When C is small, margins will be wide and there will be many support vectors and many misclassified observations.
2. When C is large, margins will be narrow and there will be fewer support vectors and fewer misclassified values, which might lead to overfitting in many cases.
3. Hence, low cost value(C) prevents overfitting and may give better test set performance.
4. We iteratively find the optimal value of C at which we obtain the best test performance.

For non linearly separable data:

We use kernels to create non linear boundaries - polynomial or radial depending on the dataset.

Principal component analysis(PCA):

It is an unsupervised learning algorithm that uses dimensity reduction features. It finds its use in noise filtering, visualization and pattern/trend/signal analysis. It has two main goals - to identify patterns in data and detect correlation between variables.

**Experimental Evaluation**

Firstly the Training Data set is divided into 4 parts - considering 3 private and 1 public dataset.

Then For better feature extraction we used Principal Component Analysis. The projection is generated for 2 principal components and scatter plot is projected.

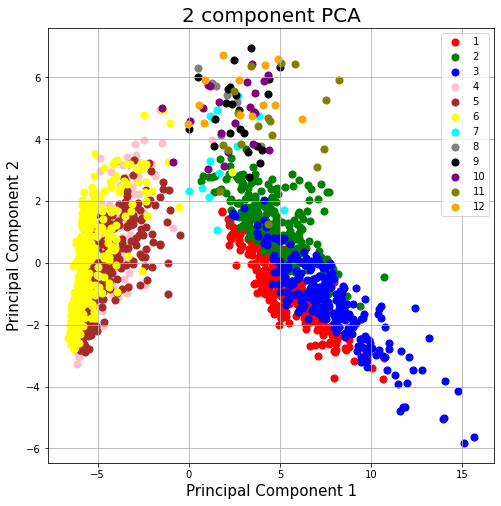
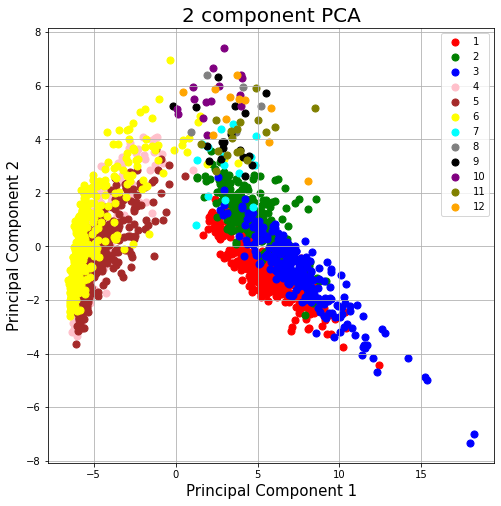
This is done for the 4 parts and projections are shown. PCA makes distinction between the various activities by clustering them.

**Methodology**

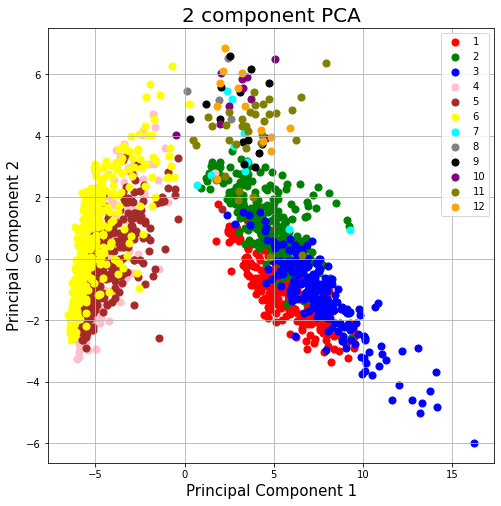
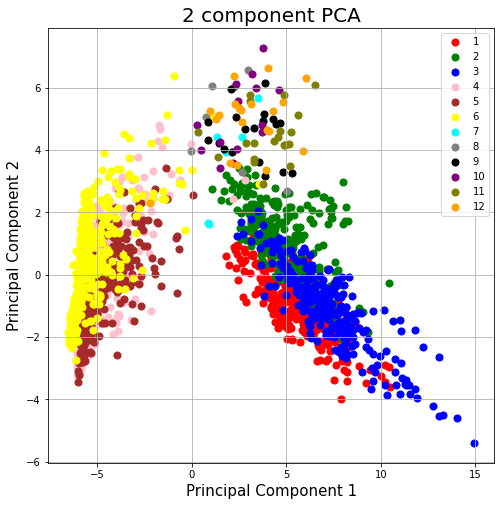
1. We imported the training data set into an numpy array
2. Separated the data into 4 equal arrays using the train\_test\_split() for randomly dividing the data
3. Run SVM on 3 datasets (considered private) for different values of C, in terms of 2^i. Obtained the optimum C value to avoid overfitting and underfitting.
4. For SVM we use 0.8 of training data to train the model and 0.2 of training data for validation.
5. Further we have tried to estimate precisely the most optimum C value by running a loop for smaller intervals ( as compared to 2^i ), by increasing C by 0.025 value per loop.
6. Since we have divided data equally, we take the mean C and train the 4th dataset on the optimum C value calculated.
7. On the trained model, the testing accuracy calculated on the test dataset is 92.69%

**Results**

The Principal Component Analysis of the 4 different parts

****

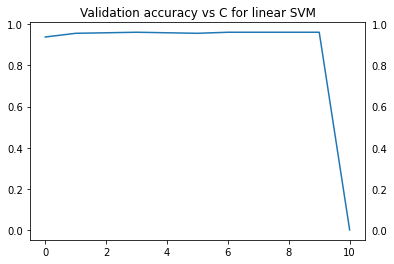
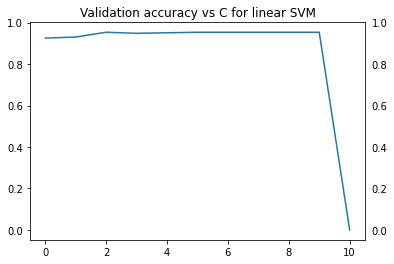
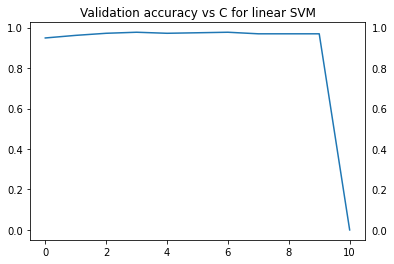
PCA of Part -1 PCA of Part-2



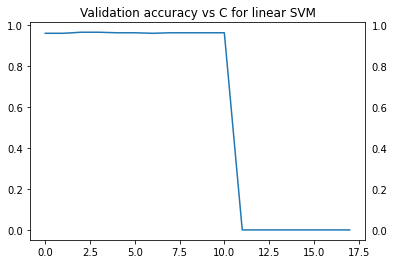
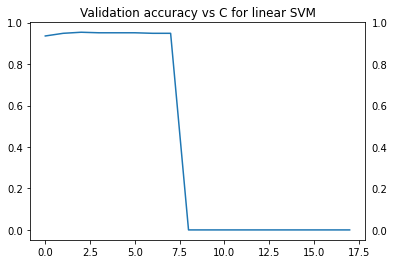
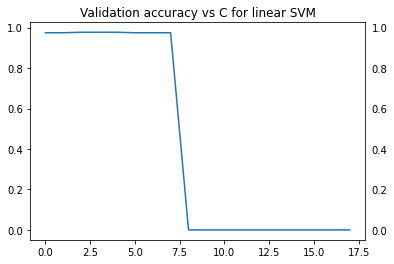
PCA of Part-3 PCA of Part-4

The above PCA projections display similar distribution in all four sub-datasets. Each of the 12 activities in the y\_train file has been represented using different color points in the scatter plot.

The Validation Accuracy for various values of C for the different parts are:-



During more precise estimation of the optimum C value, the graphs obtained are as follows:



We observe that the accuracy results before and after the precise estimation of C optimum remain the same. Both gave an accuracy of 92.69% on the given testing dataset.

**Future Scope**

Deep learning algorithms like Concurrent Neural Networks(CNN) and Long Short-Term Memory (LSTM) networks( a type of recurrent neural network(RNN)) can be used since they involve learning order dependence in sequence(signal) prediction problems.

Further research would also incorporate adapting the proposed data mining approach on embedded hardware using suitable implementation techniques so that it can be implemented on smartphone devices in a way that they use less processing time and space. Tensorflow Lite is one such technique that stores the weights of Deep Learning models and directly predicts the motion in a fraction of time. Apart from this, other powerful and novel unsupervised learning techniques need to be examined as model building in real-time on resource-constrained smartphones could be restrictive.

**Conclusion**

The conclusion is that we have classified the various activities and the accuracy is 92.69% on the testing set when we had used federated transfer learning . The federated learning technique is utilised and the linear kernel SVM has given a good performance.