Kaggle
Multi Classification Model
Method

We were given the data in three files -train data, test data and classification labels.
First I uploaded them on the drive and read them.
When the data read the feature vector(list) was being read as a string so those were converted to list.
Data was clean.
First I tried many classification algorithms such as Knn, Decision Tree and logistic regression but the score on these datasets was not good and the model was overfitting.
Then I tried to reduce the features using PCA and i reduced them to 50
For doing this I combined the train and test data and then performed PCA,then I disjointed the training and test data in ‘train’ and ‘test’(i was getting wrong predictions when i used PCA on test and train data separately)

Result

The best score that I got on Kaggle was .84375 and I got this score when I used 50 features and entire training data to train my model.
(I tried on different n values and different training and test splits and this was the best I got)

Conclusions

In making this model I came to a conclusion that when the number of features is more and data is less SVM classifier works best.(from those which we have studied in class)
If features are reduced using techniques like PCA we can get rid of a large number of features which may help us in improving the accuracy of our model.
I also observed that train_test_split produces different models with the same test_train ratio as the data that it takes for training and testing differs each time we run the code.Using cross validation we can find the range of scores that a particular split can produce.
Model score does not mean that the accuracy that we will get on some other data will be the same.But it provides us a good idea of how our model may perform before actually deploying the model to real world data.The models where I was

getting good model_score were also the ones that performed well but model_accuracy and test_accuracy was not same.Sometimes the model performed better than the score while at some it performed less than the score.