PRML Project
Glass vs No-Glass classification

Vikram (B19CSE098), Vamsi Krishna Kodavali(B19CSE095), and Nikhil Raj(B19CSE059)

Abstract

In this documentation, we presented all the required information for deciding the best machine learning model to determine if a person is wearing glasses or not. Apart from this we have done all the necessary visualization using different plots using various inbuilt functions from the sklearn library.
The dataset is from a Kaggle project from the course T81-855: Applications of Deep Learning at Washington University in St. Louis.

Index Terms

I. Introduction

The Glass vs No-Glass project comes under supervised classification problem. In this problem statement we were expected to classify if the person in the given image is wearing glasses or not. However, glasses can be of any colour, any size and any type. Perhaps, considering all the factors, high accuracy and precision was expected from us.

For this problem statement we have been asked to use three different classifiers as taught in the PRML course, but as an extra effort we have used four different classifiers with best hyper parameters to improve the overall accuracy. The four models that were used are Decision Tree Classifier, Random Forest Classifier, Support Vector Machine, Multi Layer Perceptron whose explanation will be formally provided in the MODELs USED section in the document.

II. Data Visualisation

In this section all the plots that, that we found relevant subjected to the dataset are plotted.

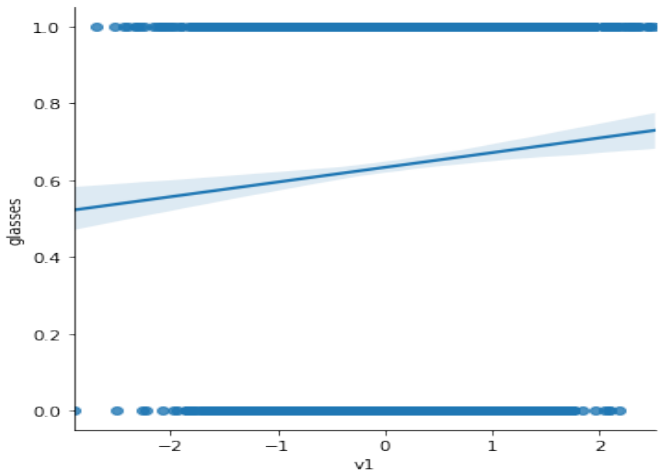


Fig. 1. Plot for visualising trend between vector 1 and Glasses (as class 0 and 1). As we can see, when we are traversing over the pixels in vector 1, we can see the probability of getting class 1 is increasing following a linear-like trend.

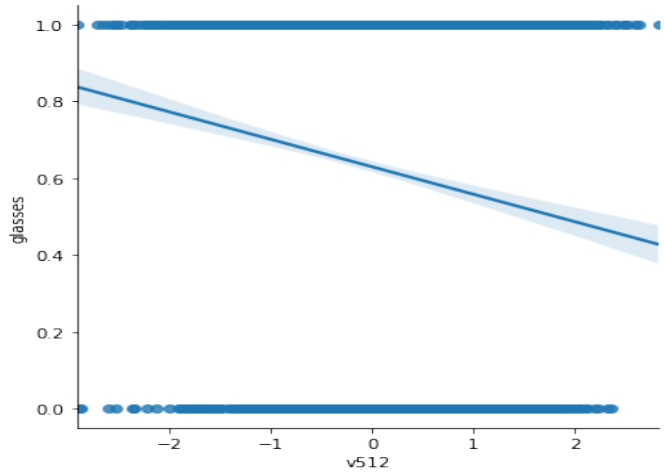


Fig. 2. Plot for visualising trend between vector 512 and Glasses (as class 0 and 1). As we can see, when we are traversing over the pixels in vector 512, we can see the probability of getting class 1 is decreasing following a linear-like trend.

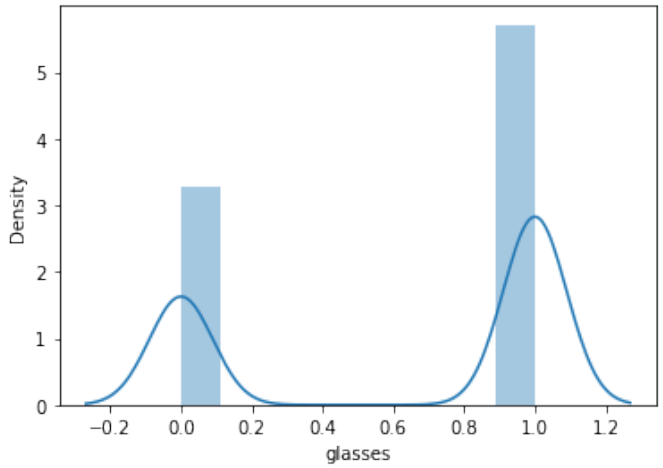
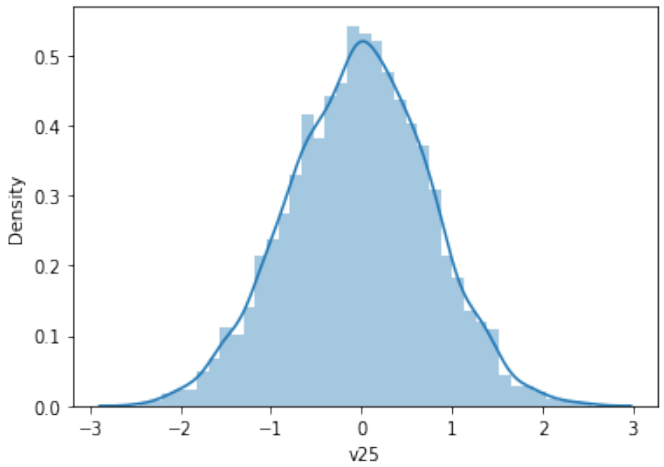


Fig. 3. Plot with a kernel density estimate with respect to different classes that are present in the dataset.



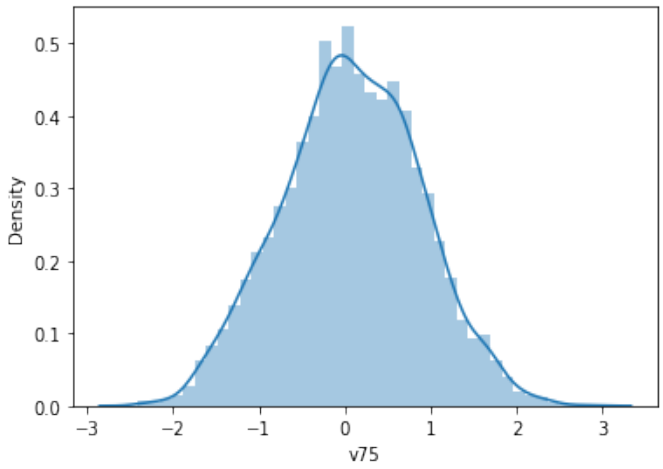


Fig. 4. Plot with the kernel density estimate with respect to random chosen vectors to visualise whether all the vectors are evenly distributed or some of the vectors have different behaviour. As we can see the for the vectors v75 and v25, it is exactly the same, hence the data is classifiable without much preprocessing required for outliers.

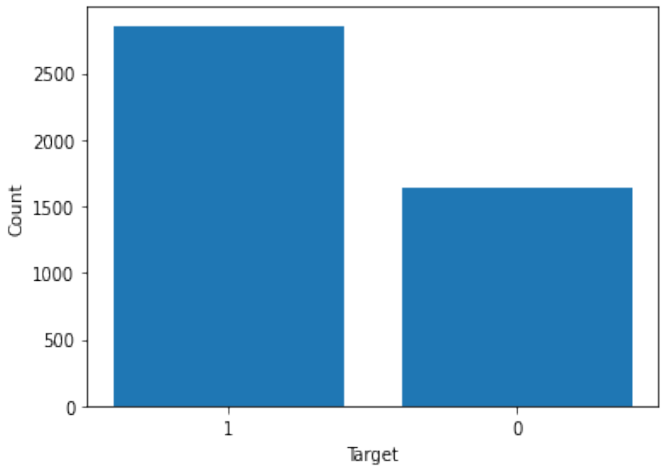


Fig. 5. Plot of count of classes where it can be seen that Class 1 has more frequency then class 0.

III. Models Used

1) Decision Tree Classifier:
The images given is in the form of GAN vectors. When we apply Decision tree classifier, what it does is dividing the vectors so that the final obtained tree can converge to the classes 0 and 1. Since class 0 are the images where the person is not wearing glasses and class 1 are the images where the person is wearing glasses, so eventually we are getting a very good classification result. The applied criterion for the algorithm is Gini as using Gridsearch CV and on hyperparameter tuning, Gini Impurity gave the better result against Entropy. So Decision Tree Classifier actually facilates to make a good performing classifier on image classification without even converting it into matrix form and then producing an image.
2) Random Forest Classifier :
Random Forest can be said as a subset of Boosting algorithm. In the context of Glass vs No Glasss problem, Random Forest uses voting technique to determine the final class from the attribute given. So in the beginning with all the GAN vectors, we set the nestimators in Random forest to 100, so that it can take 100 random vectors and train the model, and then take other 100 in various iterations and records the class from the randomly chosen parameters. Once all the various outputs are recorded, by the use of voting, it givesthe class having majority vote. Since the GAN vectors contains different features of images, So Random Forest Algorithm choose the parameters which ensure the presence of Glass or No Glass, and classifies according to that.
3) Support Vector Machine:
Support vector machines are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. In our dataset, each marked for belonging to one of two categories 0 and 1,

the SVM builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. Our SVM model represents the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

4) Multi Layer Perceptron:
In MLP, what is happening is that, it takes all the GAN vectors, which consists of features of the image and fires neurons corresponding to every new hidden layers formed. Based on the initial weights it classifies the images giving labels 0 and 1 representing persons wearing no-glass and glass respectively and if the classification is wrong back propogation takes place where the weights are recomputed and the same process takes place.

IV. Conclusion

We have used principal component analysis(PCA) for dimentionlaity reduction and here are are performance comparisions before and after applying PCA.
Before applying PCA:

1) Decision Tree Classifier: 0.7844444444444445
2) Random Forest Classifier: 0.9788888888888889
3) Support Vector Machine: 1.0
4) Multi Layer Perceptron: 1.0

After applying PCA:

1) Decision Tree Classifier: 0.7977777777777778
2) Random Forest Classifier: 0.9677777777777777
3) Support Vector Machine: 1.0
4) Multi Layer Perceptron: 1.0

Coming to the test cores for five fold cross validation before and after applying PCA.
Before applying PCA:

1) Decision Tree Classifier: 0.78111111, 0.77888889, 0.77111111, 0.78 , 0.79
2) Random Forest Classifier:0.96, 0.96222222, 0.98222222, 0.97111111, 0.96666667
3) Support Vector Machine: 1., 1., 1., 1., 1.
4) Multi Layer Perceptron: 0.99444444, 0.99777778, 0.99222222, 0.99111111, 0.99

After applying PCA:

1) Decision Tree Classifier: 0.95111111, 0.96888889, 0.96111111, 0.96111111, 0.93888889
2) Random Forest Classifier: 1., 1., 1., 1., 1.
3) Support Vector Machine: 0.99777778, 0.99888889, 1., 1., 0.99888889
4) Multi Layer Perceptron: 1., 0.99888889, 0.99888889, 0.99555556, 0.99777778

On observing the accuracies and cross validation scores the best performing model for the given dataset is Support Vector Machine.

V. Contribution

All of us have equally contributed in completing the project.
1. Research:
Vikram:25 percent
Vamsi: 50 percent
Nikhil: 25percent
2. Code:
Vikram:50 percent
Vamsi: 25 percent
Nikhil: 25 percent
3. Report and Analysis:
Vikram:25 percent
Vamsi: 25 percent
Nikhil: 50 percent