SMAI Mini Project -2

Analytical Report

Arhant Jain

20161237

Abstract :

In this project I have performed classification task on CIFAR-10 dataset and found out accuracy & f1-score for various combination of classifiers and dimensionality reduction techniques.

Classifiers used are **Decision Tree, Kernel SVM with RBF Kernel, Logistic Regression, MLP** and Dimensionality reduction techniques used are **PCA, LDA** and also performed for **Raw Data**.

I have taken train and test data set in 5:1 ratio (as done in CIFAR-10 documentation).

I have also tried to solve overfitting problem and its rectification.

Interesting Observations :

1. While changing **hyperparameter**, I noticed that on changing the value of **C and gamma** in kernel SVM , decision boundary changes and changes in the following manner.

a. A low c makes decision surface smooth.
b. A large C makes aims at classifying all training examples correctly.

2. Trends seen in changing the values of **gamma.**

a. If gamma has low value then far points decide the decision boundary.
b. If gamma has high value then close points will decide the decision boundary.

Results & Observations:

Table showing Accuracy and F1-scores for various classifier and features combination:

Classifier	Features	Accuracy	F1-Score
Decision Tree	**Raw Data**	40.25%	40.25%
Decision Tree	**PCA**	33.43%	33.43%
Decision Tree	**LDA**	37.26%	37.26%
Kernel SVM with RBF Kernel	**Raw Data**	43.34%	43.34%
Kernel SVM with RBF Kernel	**PCA**	45.67%	45.67%
Kernel SVM with RBF Kernel	**LDA**	35.22%	35.22%
Logistic Regression	**Raw Data**	31.87%	31.87%
Logistic Regression	**PCA**	40.12%	40.12%
Logistic Regression	**LDA**	33.55%	33.55%
MLP	**Raw Data**	49.26%	49.26%
MLP	**PCA**	26.70%	26.70%
MLP	**LDA**	31.28%	31.28%

Observations:

1. High Accuracy is obtained by using **Raw data** (without any dimensionality reduction techniques) but it takes highest time. Also in Raw data highest accuracy and f1 score is achieved by using MLP classifier.

2. Least time for processing is taken by **MLP** and highest accuracy is also obtained by using MLP as compared to other classifiers.

3. Accuracy and f1 score can be decreased in **MLP** classifier by increasing learning rate.

4. For all 4 classifiers(**Decision Tree, Kernel SVM with RBF Kernel, Logistic Regression, MLP**) computation time can be reduced by reducing tolerance but at the expense of accuracy and f1 score.

5. Computation time can be reduced in **PCA** and **LDA** by reducing number of components but it will also decrease accuracy and f1 score.

6. Same accuracy is obtained by using tanh and relu as **activation functions**.

7. In **MLP** highest accuracy can be achieved by using Solver lbfgs.

8. For **Kernel SVM with RBF kernel** using shrinking heuristic increases accuracy and f1 score.

Overfitting Problem:

Showing Cross validation score for determining overfitting:

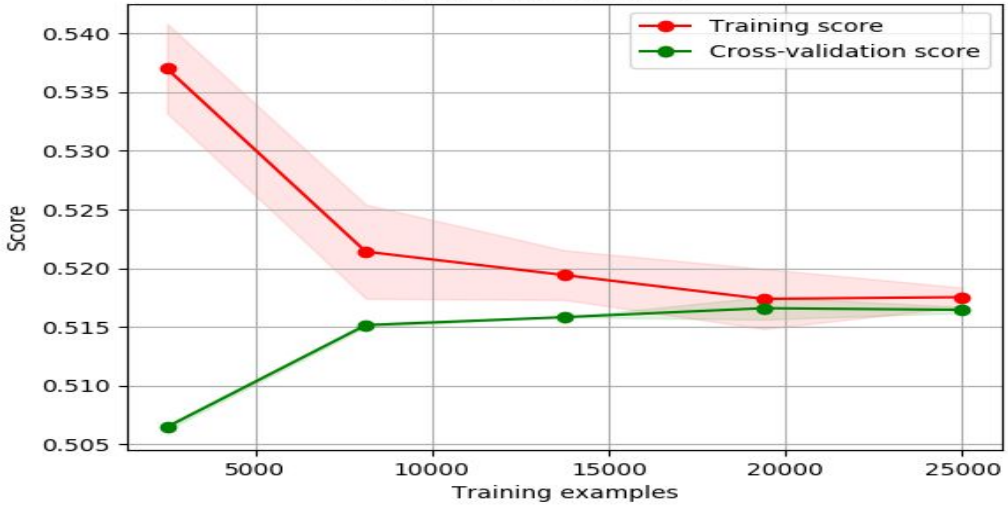
With the help of cross validation I can detect overfitting and can rectify it.

Cross-validation allows us to tune hyperparameters with only our original training set. With this our test dataset will be completely separate and unseen for selecting final model.

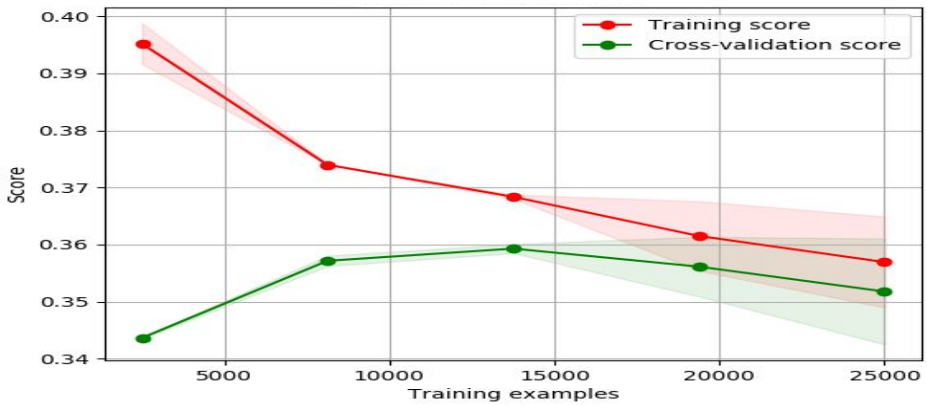
In standard k-fold cross-validation, I have partitioned the data into k subsets, called folds. Then, I iteratively train the algorithm on k-1 folds while using the remaining fold as the test set.

Below are training and cross-validation scores for various combinations of classifiers and dimensionality reduction techniques:

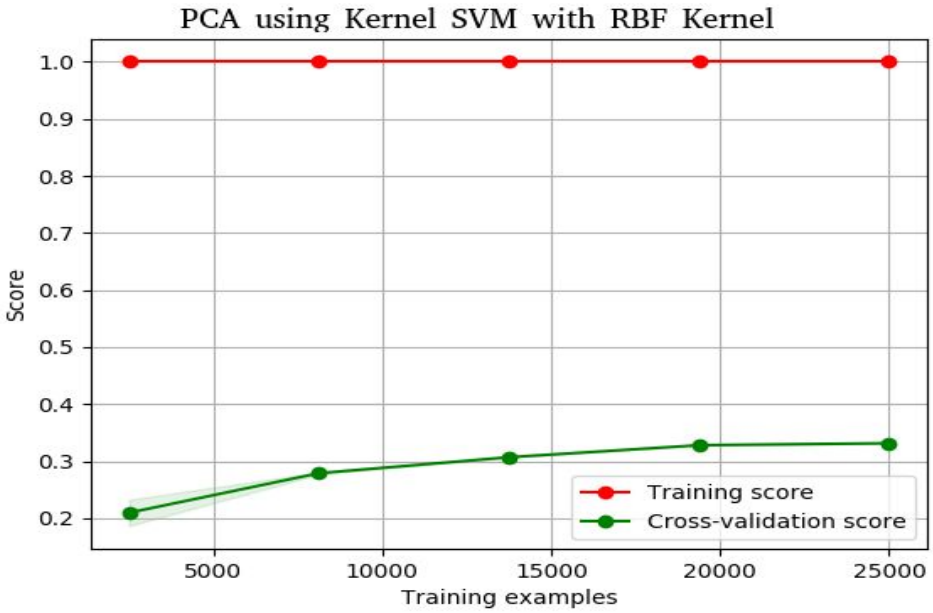
1. When using **Decision Tree** with LDA as Dimensionality reduction technique:



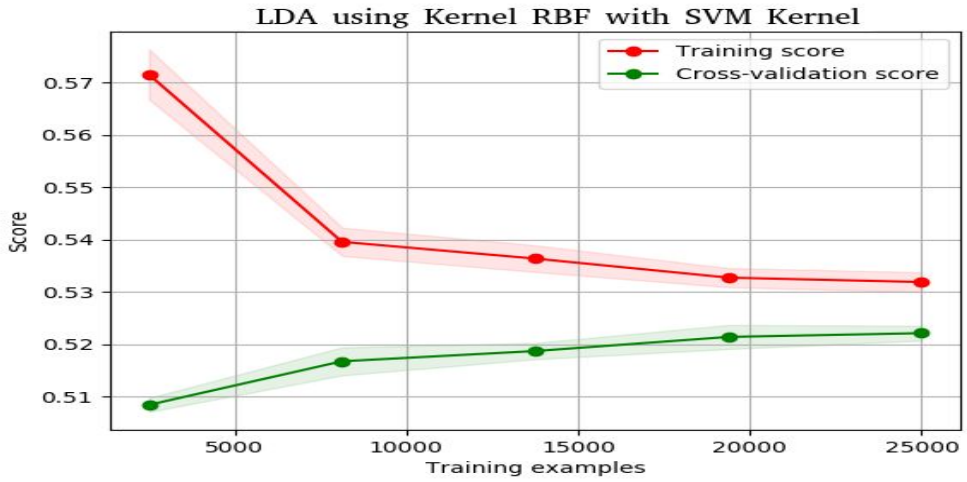
2. When using **Decision Tree** with PCA as Dimensionality reduction technique:



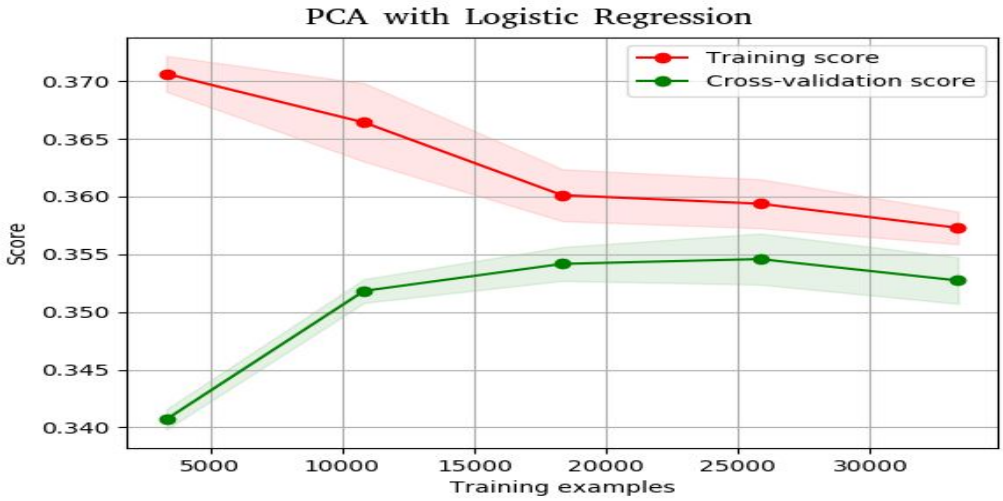
3. When using **Kernel SVM with RBF Kernel** with PCA as Dimensionality reduction technique:



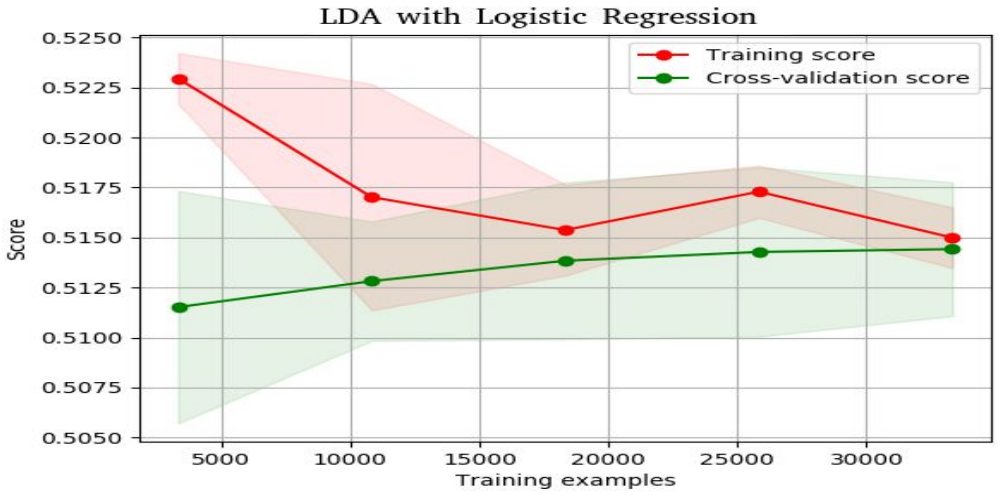
4. When using **Kernel SVM with RBF Kernel** with LDA as Dimensionality reduction technique:



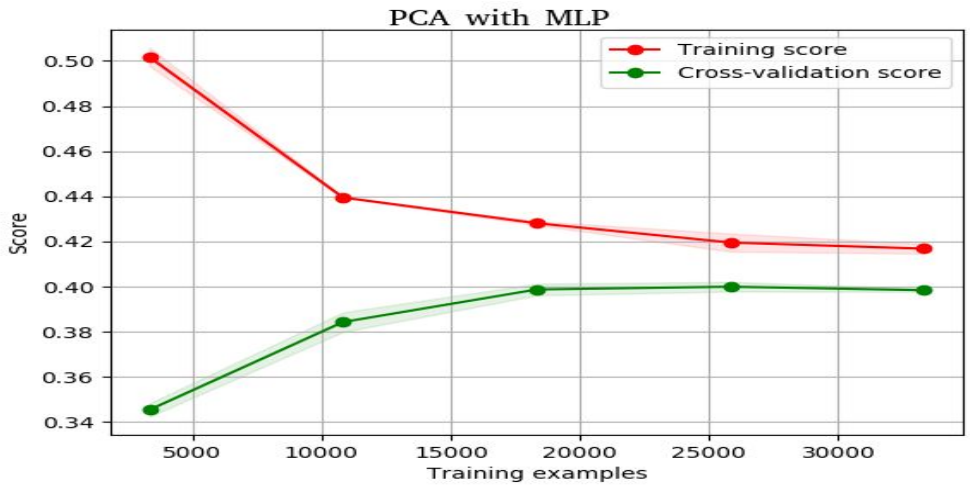
5. When using **Logistic Regression** with PCA as Dimensionality reduction technique:



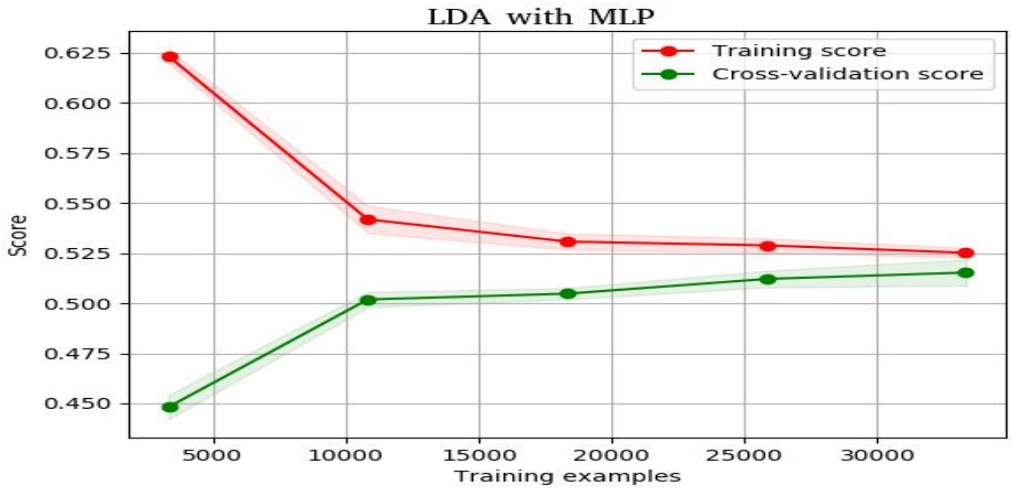
6. When using **Logistic Regression** with LDA as Dimensionality reduction technique:



7. When using **MLP** with PCA as Dimensionality reduction technique:



8. When using **MLP** with LDA as Dimensionality reduction technique:



Second approach to measure overfitting is as the distinction between the test accuracy and the train accuracy. This idea of overfitting already occurs on the existing dataset. Another idea of overfitting is the gap between the test accuracy and the accuracy on the underlying data distribution.

Rectifying overfitting:

To remove overfitting, I make new dataset called as CIFAR-B and measure the accuracy of CIFAR-10 classifiers by creating a new test set of unseen

images. The data collection for CIFAR-B will be designed to minimize the distribution shift relative to the original dataset.

Problems Faced:

1. First and major problem was computation time(really irritating) for various combinations of classifiers and dimensionality reduction techniques. And highest time was taken when computed for Raw data.
2. To get optimal solution (for each combination) I have to change hyperparameters for various classifiers and dimensionality reduction techniques. So, it was also taking time.

References:

- Scikit learn Website for understanding how to use various classifier functions and hyperparameters in python.
- For plotting curves for training and cross validation scores https://scikit-learn.org/stable/modules/learning_curve.html
- https://github.com/Hvass-Labs/TensorFlow-Tutorials for getting python codes for downloading and loading CIFAR-10 dataset.
- http://rodrigob.github.io/are_we_there_yet/build/classification_datasets_results.html#43494641522d3130