**CS697A – Topic in Computer Science – Machine Learning**

**Summer 2021**

**Assignment 3 (Solutions)**

**Group number : 11**

**Group members:**

**Abinaya Swaminathan - yf7584**

**Dhruvil Shah - pw9773**

**Megha Dusane - eb3597**

**Q1 [1pts]** Create the following **training datasets** from the optdigits.tra set:

X25: Randomly chosen N=25 instances from each class.

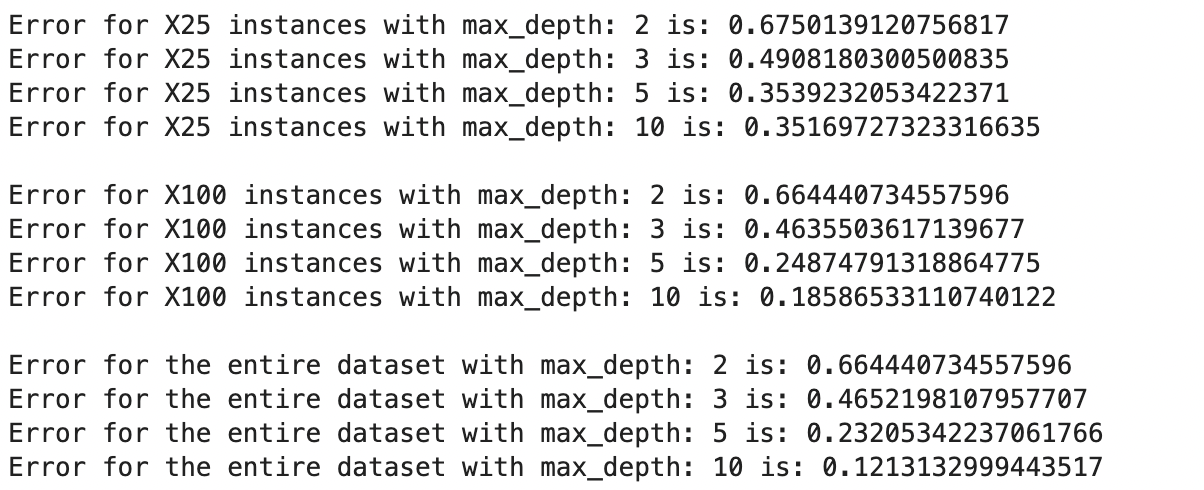
Contains (250,65) rows and columns of data

X100: Randomly chosen N=100 instances from each class.

Contains (1000,65) rows and columns of data

**Q2 [2pts]:** Decision Trees, classification:

For the DecisionTreeClassifier determine the value of the **tree depth** parameter (experiment with depth=2, 3, 5, 10) that results in the best test error. Report the training and test errors for each depth value and the training set. How does the best depth value change as the number of instances change?

****

The best depth value with the least error rate is X100 with max\_depth=10. As the number of instances change from 25 to 100, the error rate gets minimal and accuracy improves.

**Q2 [2pts]:** Repeat Q1 using XGBoost and LightGBM algorithms. Experiment with at least 5 different parameter settings to see their effect on training and test errors. How do best parameters change as the #training instances change?

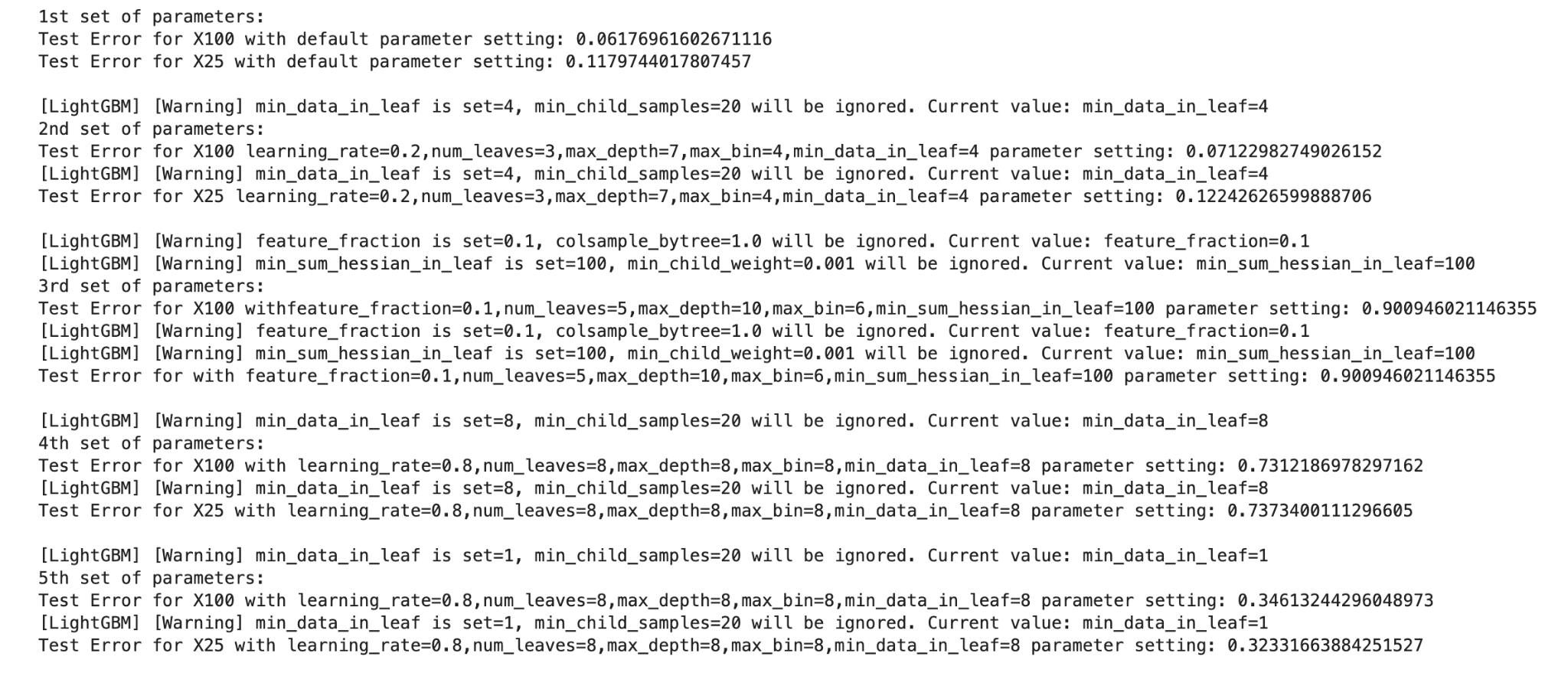
XGBoost:



As the number of training instances change and with varied best parameters, the error rate for these parameters reduces for increased instances.

The best parameter setting from my observation with minimal error rate is for the default parameter setting

LightGBM:

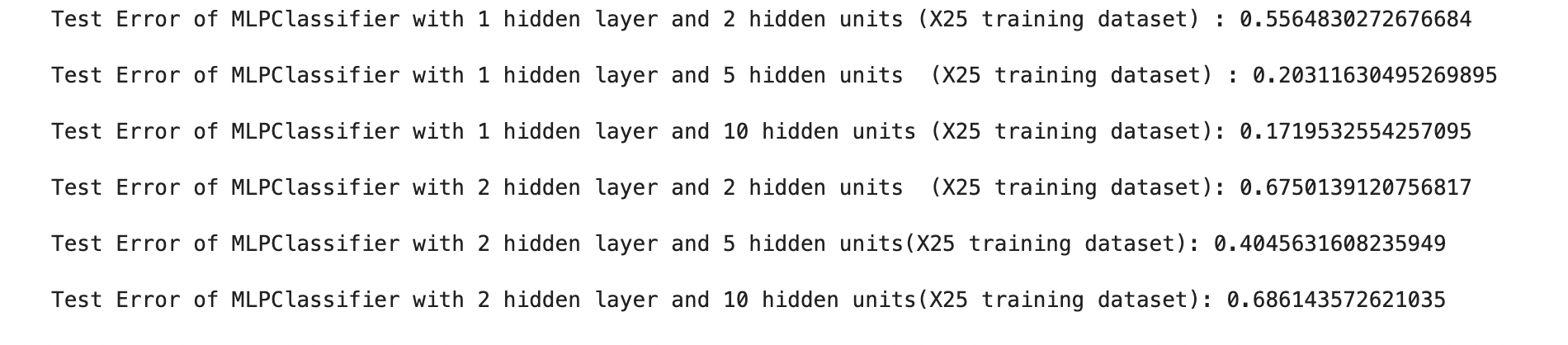


As the number of training instances change and with varied best parameters, the error rate for these parameters reduces for increased instances.

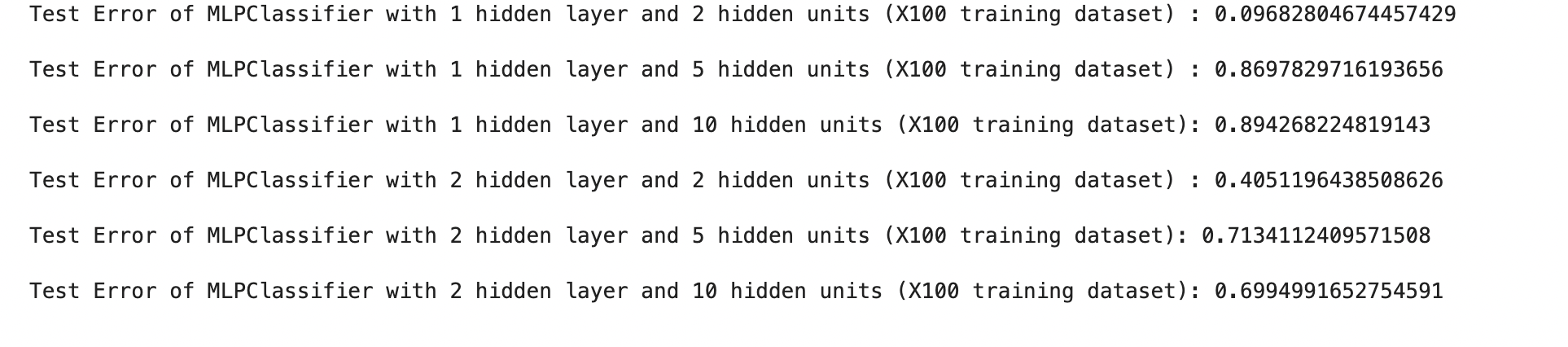
The best parameter setting from my observation with minimal error rate is for the default parameter setting

**Q3 [2pts]:**

For the neural network (MLP) determine the value of the best **hidden\_layer\_sizes** (experiment with 1 and 2 hidden layers and 2, 5, 10 hidden units in each layer) that results in the best test error for each of the training data sets you created.



The MLP Classifier gives the best test error for 1 hidden layer and 10 hidden units for the X25 instance training dataset and the error is approximately around 0.1720. From the above screenshot we can say that the error rate increases as the hidden layer increases and the error rate decreases as the hidden unit increases.



The MLP Classifier gives the best test error for 1 hidden layer and 2 hidden units for the X100 instance training dataset and the error is approximately around 0.097.

From the screenshot we can notice that the error rate increases for the increased values of hidden layers and hidden units.

**Q4 [3pts]:** **Regression for digit completion**: regression

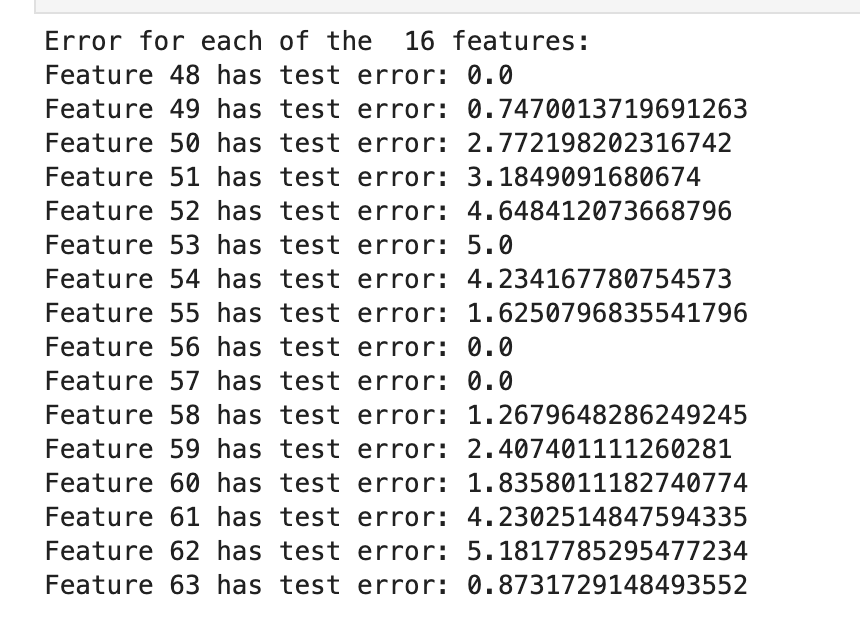
Use library sklearn.neural\_network.**MLPRegressor**

Using only the data for class 6 and class 9 in X100 for training, use the first 48 features as inputs and predict the next 16 features, i.e. create a neural network with 16 outputs.

Report the test error (use only the instances from classes 6 and 9).

Which pixels are easier to predict?

(Clarification, each of your models will have the same set of features, namely features 1…48.)



Pixels 48, 56 and 57 are easier to predict with 0 error rate and feature 63 and 49 xwith a minimal error rate