PROJECT 3-CLASSIFICATION

1. Objective:

This project is to implement machine learning methods for the task of classification. We implement an ensemble of four classifiers for a given task. Then the results of the individual classifiers are combined to make a final decision.

The classification task will be that of recognizing a 28_ 28 grayscale handwritten digit image and identify it as a digit among 0, 1, 2, ... , 9. You are required to train the following four classifiers using MNIST digit images.

The following models are to be implemented:

1. Multilayer perceptron neural network
2. Logistical Regression
3. Support Vector Machine
4. Random Forest

1.MULTILAYER PERCEPTRON NETWORK:

A two layer neural network has been implemented for both MNIST and USPS dataset. Layer 1 uses the "sigmoid" activation function and layer 2 uses sgd as the optimizer "adam".

Hyperparameter tuning:

1. Sigmoid activation function produced the best accuracy compared to the other activation functions.
2. Tanh was observed to produce the lowest accuracy among all activation functions.
3. Sgd optimizer produced an accuracy of 0.842 for the MNIST dataset.
4. Adam produced a relatively higher accuracy compared to the sgd optimizer .
5. For the MNIST dataset it produced the accuracy of 0.9455.
6. USPS dataset produced a poor accuracy of 0.31271563577582834.

7. Even though hyperparameters were tuned with different options for both optimizers , activation functions and increasing the number of layers in the neural network. It did not produce an accuracy as good as the MNIST dataset.

Accuracy:
MNIST Dataset :0.9477
USPS Dataset : 0.3181659082879638

2. LOGISTICAL REGRESSION:

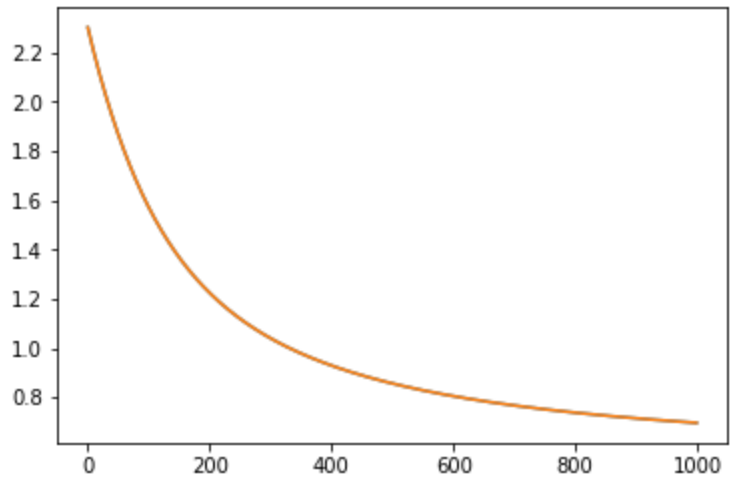
- We use Softmax regression , a generalization of logistical regression to handle our multiclass classification problem.
- It allows us to handle mutiple number of classes. In our case we have nine classes [0-9] making softmax regression a good choice to solve the problem.

Accuracy:

1. MNIST dataset:
   Training Accuracy: 0.8575
   Test Accuracy: 0.8712

2. USPS dataset:
   Test Accuracy: 0.3317165858292915

Loss graph:



Observation:
The softmax regression model was able to produce a better accuracy fior MNIST daatset when compared to the USPS dataset.

3. SUPPORT VECTOR MACHINE:

- We use the support vector machine tool in sklearn.svm.SVM to perform classification on the MNIST and USPS dataset.
- SVM uses linear seperating hyperplanes to to divide the features space into two regions.
- It sees the features of the new unseen objects and puts them into one of the two regions.

Parameters:

1. C : Refers to the penalty parameter of the error term. By default this value is set as 1.
2. Kernel: Specifies the typoe of the kernel to be used. We set the kernel value as 'linear'. If type of kernel is not specified 'rbf' is used as default.
3. Gamma: Refers to kernel coeffiecient. We set gamma to 1 and test the accuracy and repeat the same process with gamma set to default which is 'auto'.

Accuracy:

Setting 1: C= 2, Gamma = 1, kernel ='linear'
MNIST Dataset : 0. 939
USPS Dataset : 0. 3012645632281614

Setting 2 : C=1, Gamma = 'auto', kernel='linear'
MNIST Dataset : 0. 939
USPS Dataset : 0.2912645632281614

Observation:
1. The accuracy seem to be almost same for both the settings.
2. Support Vector Machine model produced a better result for the USPS dataset compared to the neural network and softmax regression model.

4. RANDOM FOREST MODEL:

- We use the Random forest Tool tool in sklearn.ensemble to perform classification on the MNIST and USPS dataset.
- Here number of estimators refer to the number of trees.
- Hyperparameter tuning:
  We set the number of estimators as 10 and test the accuracy for different values for number of trees.

Accuracy:
Setting 1: Number of trees=10
MNIST Dataset : 0.9435
USPS Dataset : 0. 31166558327916394

Setting 2: Number of trees =5
MNIST Dataset : 0.9225
USPS Dataset : 0.26666333316665836

Observation: Setting 1 with 10 trees produces better accuracy when compared to setting 2 with 5 trees.

5.COMBINED MODEL USING MAJORITY VOTING:

- This model involves combining all four classifiers based on majority voting.
- The Ensemble Vote classifier implemets “hard” and “soft” voting.
  - Hard voting (or) majority Voting predicts the final class label as the class label that has been predicted most frequently by the classification models.

- Soft Voting predicts the class labels by averaging the class-probabilities.
- Here we implement Hard voting which is also called majority voting.

Hard Voting:

It is the simplest case of majority voting. We predict the class label using the majority voting of each classifier.

Accuracy: 0.9446

6.CONFUSION MATRIX :

- Confusion Matrix is a table used to describe the performance of a classification model or classifier on a test data for which the true values are known.
- It allows to easily identify the confusion between classes.
- It summarizes the prediction results on a classification problem.
- The number of correct and incorrect predictions are summarized with count values and broken down by class.

6.1 Neural Network:

MNIST Dataset:

```
[[ 969    0    0    1    1    3    4    1    1    0]
 [  0 1120    3    2    0    1    4    2    3    0]
 [  8    3  967    9   10    2   10   12    9    2]
 [  4    1   12  948    0   16    1   12   14    2]
 [  1    1    4    0  941    0    9    3    2   21]
 [  6    3    5   28    6  810   11    3   14    6]
 [ 12    3    4    1    8   13  912    0    5    0]
 [  1    9   16    7    4    1    0  973    1   16]
 [  6    4    7   15    7   17    7   15  890    6]
 [ 10    6    1   10   21    6    0    9    4  942]]
```

USPS:

```
[[ 363   0  318  182  124  272  128  357   54  202]
 [  74  261  468  189  277  229   33  318  125   26]
 [  94    4  1361  102   28  223  112   38   28    9]
 [  51    1   383   987   4  470   29   37   26   12]
 [  19    4   93  103  941  196   48  373   142   81]
 [  33    2  580  215  10  1005   62   62   24    7]
 [ 156    2  747   65   42  312  497  117   12   50]
 [  76   38  145  748   36  163   35  597  129   33]
 [ 120    4  162  648  107  443  106  172  221   17]
 [   7    6  119  523  127   60   19  776  199  164]]
```

6.2 Support Vector machine:

MNIST:

```
[[ 958    0    5    1    1    3    8    1    1    2]
 [   0 1117    4    4    0    1    2    1    6    0]
 [   6   11  960   13    3    1   12   10   14    2]
 [   4    2   19  944    3   13    1    7   14    3]
 [   2    1    9    0  944    0    5    1    2   18]
 [  15    7    4   39    5  787   11    1   19    4]
 [  10    3   11    1    5   13  912    1    2    0]
 [   0   10   20   10    5    2    0  960    4   17]
 [  11    6    7   24   10   22    8    9  869    8]
 [   7    7    2   13   33    3    0   22    9  913]]
```

USPS:

```
[[ 358    1  493  172  239  316   69  166   11  175]
 [  59  282  572  265  240  162   15  339   44   22]
 [ 132   79 1256  131   35  224   61   48   21   12]
 [  65   52  364  884   14  501    8   43   50   19]
 [  28   27  214   90  820  213    8  456   80   64]
 [  46   26  682  249   45  824   37   38   36   17]
 [ 152   17  916   64   81  250  450   38    2   30]
 [  20   71  190  715   61  296   12  518   84   33]
 [ 121   17  278  488  123  648   83   68  154   20]
 [  13   35  200  579  166  105    8  587  146  161]]
```

6.3 Random Forest:

MNIST Dataset:

```
[[ 963    1    1    1    1    7    2    1    2    1]
 [   0 1117    4    6    0    2    3    0    3    0]
 [  13    1  984    8    3    2    3    8    9    1]
 [   1    2   14  956    0   12    2   12    9    2]
 [   3    2    3    3  930    1    8    1    7   24]
 [   6    4    2   33    7  820    6    3    9    2]
 [   9    2    5    1    5    9  923    1    3    0]
 [   1   10   22    5    4    2    1  963    3   17]
 [  14    1    8   24    9   13    6    4  883   12]
 [   8    7    6    9   24   10    1    5   10  929]]
```

USPS Dataset:

```
[[624  76 244  90 406 167 118  97  16 162]
 [ 53 611 143 172  70  71  40 798  30  12]
 [184 148 935 146  85 180  77 200  24  20]
 [ 95  98 202 904 115 362  22 154  26  22]
 [ 53 258  97  74 892 143  35 332  64  52]
 [250 135 170 249  77 888  56 116  27  32]
 [460 123 299  87 191 261 453  81  23  22]
 [119 536 337 181  79 174  23 504  34  13]
 [215 174 286 261 191 557  88  73 126  29]
 [ 69 356 298 339 294 113  29 333  75  94]]
```

6.4 Logistical Regression:

MNIST Dataset:

```
[[ 947    0    3    3    0    3   15    1    8    0]
 [   0 1094    5    3    1    2    4    0   26    0]
 [  16   19  849   26   19    0   27   22   47    7]
 [   5    3   22  883    1   28    8   19   27   14]
 [   3    8    5    0  858    1   17    2   11   77]
 [  26   15    5   83   23  644   28    9   42   17]
 [  20    5   13    2   13   19  880    0    6    0]
 [   4   39   26    1   13    0    4  887   10   44]
 [   9   14   14   39   12   22   18   14  812   20]
 [  14   13   11   12   53   10    1   26   11  858]]
```

USPS Dataset:

```
[[ 708    5  414   46  331   43   73   37   85  258]
 [ 294  290  162  254  281   34   42  303  326   14]
 [ 285   39 1120  116   79   51  104   99   89   17]
 [ 175    4  141 1116   47  205   47   72  127   66]
 [ 144   96   39   51 1078   91   30  127  240  104]
 [ 250   23  239  227   56  848  141   81   96   39]
 [ 524   15  387   92  121  120  626   21   66   28]
 [ 213  243  370  359   70   78   47  283  300   37]
 [ 277   40  191  204  181  409  137   36  442   83]
 [ 105  224  180  403  199   64   17  364  321  123]]
```

Questions:

1. **Do the results support the "No free lunch Theorem"**

   No free lunch Theorem:
   The theorem states that there is no model that works best for every problem. A great model for one problem may not work well for another problem. Therefore we have to try different models.

   -The MNIST dataset does does not support the "No free lunch Theorem" as all of the accuracies are above 50%.
   -The USPS dataset supports the "No free Lunch Theorem" as all of the accuracies are below 50%.

2. **Which classifier produces the best performance based on confusion matrix?**
   - Neural Network , Random Forest and Support Vector Machine produces the best result in comparison to Logistic Regression.

- The three classifiers have a negligible difference.
- So it seems to predict the the correct labels for each test instance.
- Logistical Regression on the other hand is not able to predict the correct labels for a lot of instances making it weaker compared to the other three models.

3. **On combining the models using majority voting classifier is the overall performance better than any of the individual classifier?**
    - The majority voting classifier produces a better result than logistical Regression.
    - In case of Neural Network, SVM and Random Forest it differs only by a negligible difference.