

Foundations of Machine Learning End Term

Data Set: - Letter Challenge Unlabelled

Letter Challenge Dataset consists of many features like x-box, y-box, height, width, and many other which predicts the letter (Target Variable) either +/- . So, based on pixels, edges and some of the other features it will predict the outcome of letter. There are 20,000 Instances in the Data and 16 Features and 1 Target Variable. Except Letter all the other features are integers but there are some unnecessary '?' in the Data Set. So, around 10,000 ? are present in the data. - includes 7240 and + includes 2760.

Cleaning Data: -

So, '?' are replaced with Blanks and then replaced with NaN's. Then after, for the analysis we replace blanks with - . So, Now - values are 17240 and + values are 2760.

Normalization: -

Rescaling the Numerical attributes into the range of 0 and 1. So, we have only Numerical Columns in the Data Set. By adding all these Numerical Columns into a Data Frame 'Data' we can Normalize these columns which results in between 0 and 1.

For our Analysis purpose we change the +/- with 0 and 1. Then, by converting 'letter' from String/Object to Integer.

Logistic Regression: -

For Logistic Regression, we use Normalized Data and storing these 'x'. To Predict we use our Target Variable. Next, we split the Data using train_test_split. Test Size I have chosen is 0.01. Then by using sklearn.preprocessing we import standardscaler. StandardScaler transforms data in such a way that the distribution will have 0 mean and standard deviation of 1.

Importing Logistic Regression from Scikit Learn. Accuracy obtained in the Logistic Regression is 88.5%.

```
cm
```

```
array([[ 1, 20],  
       [ 3, 176]], dtype=int64)
```

Confusion Matrix of Logistic Regression

From the above matrix, we can say that 23 observations were predicted wrong and 177 observations were predicted right.

Neural Network: -

So, using the same train_test_split and test size of 0.01 we are performing Neural Network. Again, From Scikit Learn we import Multi-Layer Perceptron Classifier with Hidden Layers 100

and 40. So, the accuracy obtained in Neural Network is 90%. So, There's a increase in Accuracy from Logistic to Neural Network because of Hidden Layers.

Confusion Matrix: -

From *SkLearn.metrics* we import Confusion Matrix to check True Positives, False Positives, True Negatives, True Positives. Below, is the Confusion Matrix of Neural Network.

```
nn_cm      #Out of 200 obs 180 were
array([[ 12,   9],
       [ 11, 168]], dtype=int64)
```

Confusion Matrix

From Confusion Matrix, we got about 200 Observations in which 180 are predicted true and 20 were falsely predicted.

Support Vector Machine: -

Scikit Learn provides many Libraries for Supervised Learning. So, Support Vector Machine is an algorithm which transforms data and finds the optimal boundary between possible outputs.

Accuracy of SVM is 91.5% which has increased from Neural Network.

Random Forest: -

“Random Forrest is an Ensemble of Decision Trees. Ensemble is an Aggregation function.”

We import RandomForestClassifier from sklearn.ensemble to perform Random Forest.

Accuracy of Random Forest is 99.5% which is of better accuracy than the other algorithms.

Accuracy Scores of Different Algorithms

Algorithms	Accuracy Score
Logistic Regression	88.5%
Neural Network	90%
Support Vector Machine	91.5%
Random Forest	99.5%

So, from the above algorithms we can say that Random Forest is the best model to predict the letters which has the best accuracy of all the other algorithms.