**Dharanidharan Ramasamy Karuppanasamy**

**SUID : 320028178**

**Problem**: To modify the back-propagation algorithm to obtain the best results to reduce the number of false positive rates in a predicted output.

**Dataset**:

The Adult dataset in UCI dataset database has a class value that tells if the individual earns more than “50K” or not. Based upon the attributes we have we need to predict the same.

Steps:

**Data pre-processing:**

1. The dataset has the class values (i.e., the output values) in string. Since it is a binary classification (>=50K or <50K) the values in the output class has been changed to +1 and 0 respectively. +1 represents the class with values >=50K and 0 represents the class value <50K.
2. The dataset has few missing values and so the instances of the dataset that has missing values in any parameter is removed.

**traindata=traindata.dropna()**

1. The dataset is split into test data and train data using train test split from sklearn. To make the training dataset, the dataset is split as (x, y) format where x is the input and y being the output.

**from sklearn.model\_selection import train\_test\_split**

**train,test= train\_test\_split(traindata,test\_size=0.3)**

1. The dataset has many categorical values like work class, sex, race, country, education, etc. If we randomly assign values to each category there is a chance that the neural network start forming correlation between those values. To remove this constraint, we are applying one hot encoding on this data so that the neural network doesn’t form any correlation between the values.

**train\_nn\_ip = pd.get\_dummies(train\_ip)**

**test\_nn\_ip = pd.get\_dummies(test\_ip)**

1. To make all the values under a standard scalar, standard scalar library function is used. Once the pre-processing is done this data is fed to the neural network.

**sc = StandardScaler()**

**X\_train = sc.fit\_transform(train\_nn\_ip)**

**X\_test = sc.transform(test\_nn\_ip)**

**Neural Network:**

The number of input layers (87) , the number of hidden layers (174) and the number of output layer is defined as per the following code

**model = keras.Sequential([keras.layers.Dense(174,input\_dim=87,activation=tf.nn.sigmoid),keras.layers.Dense(1),])**

**model.compile(optimizer='SGD',loss='mse',metrics=['accuracy'])**

**model.fit(X\_train,train\_op,validation\_data=(X\_test,test\_op),epochs=50)**

The output of the corresponding model and the confusion matrix can be formed by the following code snippet

**from sklearn.metrics import confusion\_matrix,classification\_report,accuracy\_score**

**pred = model.predict(X\_test, verbose=1)**

**predictions = (pred > 0.5)**

**labels = (test\_op == 1)**

**print(confusion\_matrix(predictions,labels))**

**Sample output :**

|  |  |  |
| --- | --- | --- |
| n = 9049 | Predicted = Yes | Predicted = No |
| Actual = Yes | 6199 | 732 |
| Actual = No | 604 | 1514 |

Total no of true positives = 6199

Total no of false positive = 604

Total no of true negative = 1514

Total no of false negative = 732

**Ways to improve the back propagation algorithm to reduce the number of false positive:**

1. One of the cause that the false positives might happen is due to high learning rate and because of high learning rate , the neurons get saturated to classify almost everything towards the positive class.
2. One way to reduce the number of false positive counts is by creating a custom loss function that penalizes more when the neural network incorrectly classifies the given input.
3. Another way to reduce the number of false positives is by decreasing the learning rate of the algorithm time to time.

The neural network model is being run with various values of learning rate and the graph is plotted between learning rate and number of false positives.

|  |  |
| --- | --- |
| learning rate | False positive rate |
| 1 | 175 |
| 0.5 | 173 |
| 0.3 | 112 |

In this table when the learning rate is 0.3 the false positive rate is nearly equal to one percent of the entire dataset.

Hence by either by changing the weights of the nodes or by decreasing the learning rate, the false positives count can be reduced further more.