CSE 572: DATA MINING

Assignment 3
Spring 2018

GROUP 2

Dwarsala Venkata Manoj Reddy	1213185010
Dipika Ramaswamy	1213237517
Kiran Teja Settipalli	1213019858
Rohit Polisetti	1212933824
Srinivas Puranam	1211168267

Table of Contents

1. Data Preparation:

For this assignment, we worked on data collected by 10 users. The new feature matrix obtained in assignment 2 is used for training and testing the three machines. The data has been split in 60:40 ratio for training and testing the machines. Since we have 10 different gestures, we created 10 training data matrices and 10 test data matrices.

Data for Training:
For each gesture, 60% of the rows from new feature matrix have been taken and given them a class label '1'. Then, we picked data points from the other 9 gestures and assigned them a class label '0'. Since the number of no-class data points will be very high compared to yes-class, we picked only few data points from the remaining 9 gestures to avoid **Class Imbalance problem**. Now, this entire data will be used to train the machines for that particular gesture. The similar steps are repeated for all the other gestures.

Data for Testing:
For each gesture, we have taken the remaining 40% of the rows from new feature matrix and given them a class label '1'. Then, we picked data points from the other 9 gestures which are different from the ones picked for training and assigned them a class label '0'. Again, we picked only few data points from the remaining 9 gestures. Now, this entire data will be used to test the machines for that particular gesture. The similar steps are repeated for all the other gestures.

Since data from all the users is combined together for creating the training and test data, we reported the overall accuracy metrics instead of user-wise metrics. The three accuracy metrics-Precision, Recall and F1-Score have been computed for each of the 10 Decision Trees, 10 Support Vector Machines and 10 Neural Networks. The values are reported in the corresponding sections below.

2. Decision Tree:

Decision Tree is a supervised learning model primarily used for classification and regression analysis. This model forms a tree structure based on the training data supplied. While the tree is incrementally being developed, the data is broken down into smaller and smaller subsets. Given a set of training examples where each data-point is assigned to one of the two classes, decision tree builds a model such that it can classify the data points. When the test data is passed to this decision tree model, it tries to categorize the data into one of the two classes.

The following steps have been performed in this phase to classify data using Decision Tree.

- In the first step of this phase, the data collected from each user has been divided into training and test data.
- All the data from each user has been accumulated into a single file for each category or class.

- Then, in the matlab code, Decision Tree model is built based on the training data and its class labels using the inbuilt matlab function fitctree().
- Once the decision tree model has been built, we use the matlab inbuilt function predict() which takes the testing data and the built model as input.
- The predict() function then predicts and outputs the labels for the testdata.
- These labels are then used to determine 3 types of accuracy metrics, Precision, Recall and F1 Score by constructing the confusion matrix.

The accuracy measures are calculated using the following formulae,

Precision:
This is used to measure the fraction of relevant data from a group of newly predicted labels. The values of precision for each class are calculated and displayed in the command window of matlab.

Precision = (True-positive) / (True-positive + False-positive)

Recall:
This is used to measure the fraction of relevant data that have been correctly classified by the Decision Tree. The Recall values are displayed in the command window of matlab.

Recall = (True-positive) / (True-positive + False-negative)

F1 Score:
In binary classification, this is used to test the accuracy of Decision Tree using the calculated values of Recall and Precision. The F1-score values for each class have been calculated and displayed in the command window. The F1-score is calculated using the following formula:

F1-Score = Harmonic Mean(Precision and Recall) = 2*Precision*Recall/(Precision+Recall)

Accuracy Metrics for each class using Decision Tree:

Class	Precision	Recall	F1-Score
About	0.8241	0.8333	0.8287
And	0.9425	0.9111	0.9265
Can	0.8181	0.9000	0.8571
Cop	0.6666	0.8222	0.7363
Deaf	0.8681	0.8777	0.8729
Decide	0.8350	0.9000	0.8663
Father	0.8735	0.8444	0.8587
Find	0.8200	0.9111	0.8631
Go Out	0.7934	0.8111	0.8021
Hearing	0.8085	0.8444	0.8260

3. Support Vector Machine:

SVM is a supervised learning models mainly used for classification and regression analysis. Given a set of training examples where each data-point is assigned to one of the two classes, SVM builds a model based on the training data. Then, test data is passed to the SVM model which tries to categorize the data into one of the two classes as accurately as possible.

The following steps have been performed in this phase to classify data using SVM.

- In the first step of this phase, the data collected from each user has been divided into training and test data.
- All the data from each user has been accumulated into a single file for each category or class.
- In the matlab code, SVM model is built based on the training data and its labels using the matlab inbuilt function fitcsvm().
- Once, the SVM model has been built, we use the matlab inbuilt function predict() which takes the testing data and the built model as input.
- The predict() then predicts and outputs the labels for the testdata.
- These labels are then used to determine 3 types of accuracy metrics, Precision, Recall and F1 Score, by calculating true-positive, false-positive and false-negative.

Similar to decision trees, the accuracy measures are calculated using the following formulae,

Precision:
This is used to measure the fraction of relevant data from a group of newly predicted labels. The values of precision for each class are stored in the PrecisionVals[] matrix of matlab.

Precision = (True-positive) / (True-positive + False-positive)

Recall:
This is used to measure the fraction of relevant data that have been correctly classified by the SVM. The Recall values are stored in the RecallVals[] matrix of matlab.

Recall = (True-positive) / (True-positive + False-negative)

F1 Score:
In binary classification, this is used to test the accuracy of SVM using the calculated values of Recall and Precision. The F1-score values for each class have been calculated and stored in F1scores[] matrix of matlab. The F1-score is calculated using the following formula.

F1-Score = Harmonic Mean(Precision and Recall) = 2*Recall*Precision/(Recall+Precision)

Accuracy Metrics for each class using SVM:

Class	Precision	Recall	F1-Score
About	0.6463	0.6625	0.6543
And	0.8734	0.8625	0.8679
Can	0.6310	0.6625	0.6463
Cop	0.3929	0.4125	0.4024
Deaf	0.6020	0.7375	0.6629
Decide	0.7765	0.8250	0.8000
Father	0.6395	0.6875	0.6627
Find	0.7179	0.7000	0.7089
Go Out	0.6220	0.6375	0.6296
Hearing	0.4857	0.4250	0.4533

4. Neural Network:

Neural networks classifiers are modeled similar to the network of neurons in the human brain. They are made up of an input layer, an output layer and one or more hidden layers consisting of one or more hidden nodes. They learn from the dataset by predicting class labels for records, which are then used to calculate the error between the actual class and the predicted class values. This error is propagated backward through all the layers to adjust the weights of the edges to nodes in the hidden layer.

There are many types of neural networks such as feedforward, convolutional neural networks and recurrent neural networks. Here, we have used a feedforward neural network for classification with the following characteristics:

- Number of hidden layers - 10
- Activation function - Tangent sigmoid function for hidden layers and Linear function for output layer.
- Number of nodes in each hidden layer - 10
- Performance function - Mean Square Error

The steps involved in classification are:

- The training and test datasets are prepared in 60:40 ratio.
- The neural network is configured with the newff() function from the Neural Network Toolkit in Matlab with the given number of layers and nodes.
- The training data along with class labels is used to train the neural network model using the train() command.
- The test data is fed into the network to get test classes predicted by the model.
- These predicted classes are now compared with the known test data classes to calculate different performance metrics like precision, recall and F1 score by constructing a confusion matrix.

Accuracy Metrics for each class using Neural Network:

Class	Precision	Recall	F1-Score
About	0.7875	0.8873	0.8344
And	0.9625	0.8953	0.9277
Can	0.9125	0.8111	0.8588
Cop	0.5625	0.8205	0.8101
Deaf	0.8000	0.8205	0.8101
Decide	0.8500	0.8947	0.8717
Father	0.8375	0.7976	0.8170
Find	0.8375	0.8170	0.8271
Go Out	0.8000	0.7441	0.7710
Hearing	0.9125	0.7087	0.7978

5. Conclusion:

The training data has been used to train the different types of machines and based on test data, the accuracy metrics - Precision, Recall and F1-Score have been computed. From our observation, Decision Tree and Neural Network gave better accuracy for most of the gestures when compared to Support Vector Machine.