A7: Prediction

Harshali Singh, Vishal Mehta

- Strategy
- Time Duration
- Conclusion

Strategy

1. Modeling: We have used weka API to build our predictive model. In that we first express the problem with features, train a classifier and then use that classifer to set the dataset and produce the output.

    i. Expressing problem with features(Attributes): We have 8 numeric features and 1 nominal feature (ARR_DELAY) in our program.

```
    attributes.add(new Attribute("dayOfMonth"));
    attributes.add(new Attribute("dayOfWeek"));
    attributes.add(new Attribute("carrier"));
```

    ii. Train a classifier: Training requires 1) having a training set of instances and 2) choosing a classifier(Naive Bayes here).

```
iFlight.setValue((Attribute)fvWekaAttributes.get(0), flight.getDayOfMon
th());
iFlight.setValue((Attribute)fvWekaAttributes.get(1), flight.getDayOfWee
k());
iFlight.setValue((Attribute)fvWekaAttributes.get(2), flight.getCarrier
().hashCode());
```

    iii. Use the classifier.

```
            iFlight.setDataset(isTestingSet);
            double[] predictionDistribution = classifier.distributionFo
rInstance(iFlight);
            String prediction = null;
            if(predictionDistribution[0] > predictionDistribution[1]){

                prediction = "TRUE";

            } else prediction = "FALSE";
```

2. Prediction

We have used Naive Bayes theorem to predict the delays. Naive Bayes is a simple probabilistic classifier based on applying Bayes' theorem (or Bayes's rule) with strong independence (naive) assumptions.Bayes's rule: P(H | E) = P(E | H) x P(H) / P(E) The basic idea of Bayes's rule is that the outcome of a hypothesis or an event (H) can be predicted basedon some evidences (E) that can be observed.

In our program, we have taken DayOfMonth, DayOfWeek, Carriercode, Origin , Destination, Sch dep Time, Days Till Nearest Federal Holiday, Arrival delay as numeric features while training the classifier.

Also, the ARR_DELAY is used as class attribute and for predicting if the flight is delayed or not. If the probability prediction distribution comes out to be TRUE then we predict that the flight is delayed otherwise not.

3. FLOW OF THE PROGRAM:

The prediction Mapper gives the output having month as a key and all the other attributes as value to the reducer. The Prediction Reducer train the models(based on month) and use the classifier to emit the data based on monthly models using Naive Bayes classifier of Weka.

After that, the program reads the test file, uses the monthly models to predict if the flight is delayed or not using probability prediction distribution as described above.

The final output from TestReducer would be flight number, flight date and Sch Departure time as the key AND a boolean value (True or False) describing if the flight is predicted delayed or not.

4. CONFUSION MATRIX

A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known.

Terminologies:

```
a. true positives (TP): These are cases in which we predicted yes (flight is delay
ed), and the flight actually got delayed.
b. true negatives (TN): We predicted no, and the flight is not delayed.
c. false positives (FP): We predicted yes, but the flight is not delayed.
d. false negatives (FN): We predicted no, but the flight is actually delayed.
```

We have written a Spark program to generate the confusion matrix which takes the output of the Prediction program and a validate file to determine the accuracy of the prediction. The result of confusion matrix is:

```
    TP : 1266738
    TN : 1533718
    FP : 918197
    FN : 1344705

    Accuracy : 55.308275654% approx. (TP+TN/TP+TN+FP+FN)
```

Time Duration

We have used python script to ping the emr cluster to know whether it has terminated or not. It also calculates the running time of the program.

Prediction on pseudo: 8 minutes approx Prediction on EMR: 5 minutes approx

Conclusion

As the most likely reason for the flight to get delayed is bad weather which is highly unpredictable, and the accuracy which we got from this confusion matrix comes out to be between 50-60% which is pretty good for this kind of flight data. If the accuracy would have been more than 60% then the result would have been erroneous. We found Naive Bayes algorithm better than Random Forest algorithm for prediction because the flights are independent of each other which is a good requirement for Naive Bayes classification.