Introduction to Big Data

Predicting Airline Delays

HU Haoran && CUI Yuxuan

M1 BI-2

1. Problem Definition

Today, the problem of airline delays is becoming more and more common. Due to certain problems, most airlines suffer from currency losses and loss of customer value. If we can predict the delay of the airline before departure, then we can save a lot of money and guarantee the value of the customer. But due to the presence of natural factors, these delays still occur. Therefore, we can use machine learning models to predict all these delays, such as security delays, aircraft delays, and so on. The data set of our project can be downloaded from Bureau of Transportation Statistics where it is described in detail. We hope to reduce the number of air delays through our efforts.

2. Tools

In this project, the tool we use is Python and scikit-learn.

3. Methodology

(1) In this project, we use many Machine learning package, for example, pandas (to process CSV files), StandardScaler, MinMaxScaler (our two methods) etc.

(2) We store the data in a csv file and extract it as an array.

(3) In many data, we delete unwanted data columns, after that, we got a complete data.

(4) We divide many air delays into six : ArrDelay, CarrierDelay, WeatherDelay, NasDelay, SecurityDelay, LateAircraftDelay and AllDelay.

- ArrDelay : This represents the difference between the scheduled arrival time and the actual arrival time. Advance arrival is represented by a negative number, and delayed arrival is represented by a positive number.
- CarrierDelay : This represents the minute difference between the planned and actual carrier time.
- WeatherDelay : This represents the minute difference between the scheduled and actual arrival time due to the weather.
- NasDelay : This represents National Air System Delay (NAS-Delay in minute).
- SecurityDelay : This represents the difference between the planned time and the actual time in the security department.
- LateAircraftDelay : This represents the minute difference between the plan and the actual time (to delay the aircraft).

- AllDelay : This represents the total of the previous six delays.

(5) About the Scaling and Normalizing Data, we divide two scaling, Standard Scaling and Min-Max Scaling.

- Standard Scaling : About this scaling, we subtract the average of each feature and divide it by the standard deviation of each feature to obtain data with an average of zero in this normalization technique.
- Min-Max Scaling : About this scaling, we tilt the values of each feature so that their minimum becomes zero and the maximum becomes ten. Basically, we tilt each feature from zero to one in this normalization technique.

(6) About the regression models, we divide two method: Linear regression and Decision tree.

- Linear regression: This regression is about a linear approach to modelling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables).
- Decision tree: This regression is about a decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility.

(7) Because we have seven delay, two scaling and two regression models, we have 28 different situations. In our code section, you can see our 28 different codes.

4. The results and Discussion

- **Standard Scaling**

Delay Type	Machine Learning Model	Error on testing
ArrDelay	Linear regression	61.10788591984514
ArrDelay	Decision tree	32.22683165248075
CarrierDelay	Linear regression	300.04350478245016
CarrierDelay	Decision tree	439.57761724244807
WeatherDelay	Linear regression	100.11801229935249
WeatherDelay	Decision tree	172.89916314998928
NasDelay	Linear regression	201.53902585086502
NasDelay	Decision tree	275.886746298548

SecurityDelay	Linear regression	0.7832722691925871
SecurityDelay	Decision tree	1.5934863982070906
LateAircraftDelay	Linear regression	293.0151557792206
LateAircraftDelay	Decision tree	412.91423121855854
AllDelay	Linear regression	219.10028139449045
AllDelay	Decision tree	121.43500941754286

- Min-Max Scaling

Delay Type	Machine Learning Model	Error on testing
ArrDelay	Linear regression	61.107885919845934
ArrDelay	Decision tree	50.52059700069142
CarrierDelay	Linear regression	300.0435047824517
CarrierDelay	Decision tree	434.2389767064826
WeatherDelay	Linear regression	100.11801229935196
WeatherDelay	Decision tree	167.76673580812056
NasDelay	Linear regression	201.53902585086576
NasDelay	Decision tree	276.3502610685931
SecurityDelay	Linear regression	0.7832722691925871
SecurityDelay	Decision tree	1.7943971580478268
LateAircraftDelay	Linear regression	293.01515577921987
LateAircraftDelay	Decision tree	406.47202632143626
AllDelay	Linear regression	219.10028139450088
AllDelay	Decision tree	122.98194215959755

We use a lot of data to conduct test analysis to draw conclusions. We use machine learning models and test them in two ways (Min Max Scaling and Standard Scaling). Through the test results, we found that the distribution and trend are almost the same, so the data is not skewed, and each feature is different in its distribution from its value. Therefore, in machine learning, we can extend these features according to their distribution and judge whether these features perform best.

5. The guideline

(1) The package required to import the project.

```
import pandas as pd
import numpy
import os
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import MinMaxScaler
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
from sklearn.tree import DecisionTreeRegressor
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
```

(2) Import csv file.

```
os.chdir('C:\\Users\\dell\\Desktop')
data=pd.read_csv('2008.csv')
print (data.shape)
print (data['ArrDelay'].isnull().sum())
data['ArrDelay'].fillna(-1, inplace=True)
```

(3) Delete unwanted data columns

```
data=data.drop(['UniqueCarrier','Origin','TailNum','Dest','CancellationCode'],axis=1)
data.isnull().sum()
```

(4) Define the type of the delay.(Seven kinds)

```
data=data.fillna(0)
ArrDelay=data.pop('ArrDelay')
CarrierDelay=data.pop('CarrierDelay')
WeatherDelay=data.pop('WeatherDelay')
NasDelay=data.pop('NASDelay')
SecurityDelay=data.pop('SecurityDelay')
LateAircraftDelay=data.pop('LateAircraftDelay')
AllDelay=ArrDelay+CarrierDelay+WeatherDelay+NasDelay+SecurityDelay+LateAircraftDelay
```

(5) Define the Normalizing. (Standard Scaling and Min-Max Scaling)

```
# # Standard Scaling

scaler=StandardScaler().fit(data)
std_scaled = scaler.transform(data)

# # Min-Max Scaling

minmax_scaler=MinMaxScaler(feature_range=(0,1))
minmax_scaled = minmax_scaler.fit_transform(data)
```

(6) The linear regression (Standard Scaling)

```
# # Linear regression

print ("Using Linear Regression")
model = LinearRegression()
model.fit(x_train, y_train)
predictions = model.predict(x_test)
lin_mse=mean_squared_error(y_test, predictions)
plt.scatter(y_test,predictions)
plt.xlabel('True values')
plt.ylabel('Predictions')
plt.show()
print ("Error on Testing is : ", lin_mse)
lin_rmse = numpy.sqrt(lin_mse)
print(lin_rmse)
```

(7) The decision Tree (Standard Scaling)

```
# # Decision tree

print ("Using Decision Trees Regression")
tree_reg = DecisionTreeRegressor()
tree_reg.fit(x_train, y_train)
predictions = tree_reg.predict(x_test)
tree_mse = mean_squared_error(y_test,predictions)
plt.scatter(y_test,predictions)
plt.xlabel('True values')
plt.ylabel('Predictions')
plt.show()
print ("Error on Testing is : ", tree_mse)
tree_rmse = numpy.sqrt(tree_mse)
print(tree_rmse)
```

(8) The linear regression (Min-Max Scaling)

```
# # Linear regression

print ("Using Linear Regression")
model = LinearRegression()
model.fit(x_train, y_train)
predictions = model.predict(x_test)
lin_mse=mean_squared_error(y_test, predictions)
plt.scatter(y_test,predictions)
plt.xlabel('True values')
plt.ylabel('Predictions')
plt.show()
print ("Error on Testing is : ", lin_mse)
lin_rmse = numpy.sqrt(lin_mse)
print(lin_rmse)
```

(9) The decision Tree (Min-Max Scaling)

```
# # Decision tree

print ("Using Decision Trees Regression")
tree_reg = DecisionTreeRegressor()
tree_reg.fit(x_train, y_train)
predictions = tree_reg.predict(x_test)
tree_mse = mean_squared_error(y_test,predictions)
plt.scatter(y_test,predictions)
plt.xlabel('True values')
plt.ylabel('Predictions')
plt.show()
print ("Error on Testing is : ", tree_mse)
tree_rmse = numpy.sqrt(tree_mse)
print(tree_rmse)
```

We have an annex, There are many distribution trend figures in the annex.

6. The problem

When we do this project, we meet some problem. For example, when we use the dataset, Some data has shown N/A, we thought that these data is a bad data, we couldn't use these data to analyze, so we deleted these data and saved good data; and the other problem is when we use One-Hot encoding, due to insufficient memory in our computer, it showed "Memory Error", so we gave it up.

7. Conclusion

Through this project, we first have a better understanding of big data. Secondly, we also learned to use some software related to big data. In the following days, we continue to learn big data in depth.