**CASE STUDY 2**

1. a. The flight\_df Data frame dimensions:

Number of rows in the data frame: 2201

Number of columns in the data frame: 11

Shape of the data frame (rows, columns): (2201, 11)

b. After removing the ‘DEST’ and ‘ORIGIN’ columns from the flight\_df data frame, the data types of the remaining columns in flight\_df data frame are shown below.

SCH\_TIME int64

CARRIER object

DEP\_TIME int64

DISTANCE int64

FL\_NUM int64

WEATHER int64

WK\_DAY int64

MTH\_DAY int64

FL\_STATUS object

The data types of the columns are integer data type for all the columns except for ‘CARRIER’ and ‘FL\_STATUS’, which have object data types.

c. Leaving the data type of the outcome variable ‘FL\_STATUS ’ unchanged in the data frame, the data type of the predictor variable ‘CARRIER’ is changed to category data type and introduced dummy variables.

Modified list of 15 column variables (including dummy variables):

Index(['SCH\_TIME', 'DEP\_TIME', 'DISTANCE', 'FL\_NUM', 'WEATHER', 'WK\_DAY', 'MTH\_DAY', 'CARRIER\_DH', 'CARRIER\_DL', 'CARRIER\_MQ', 'CARRIER\_OH', 'CARRIER\_RU', 'CARRIER\_UA', 'CARRIER\_US', 'FL\_STATUS'], dtype='object')

Modified data types of columns in data frame:

SCH\_TIME int64

DEP\_TIME int64

DISTANCE int64

FL\_NUM int64

WEATHER int64

WK\_DAY int64

MTH\_DAY int64

CARRIER\_DH uint8

CARRIER\_DL uint8

CARRIER\_MQ uint8

CARRIER\_OH uint8

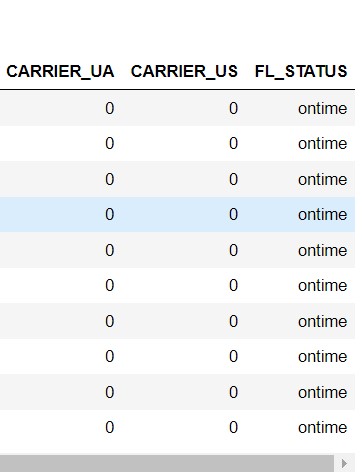
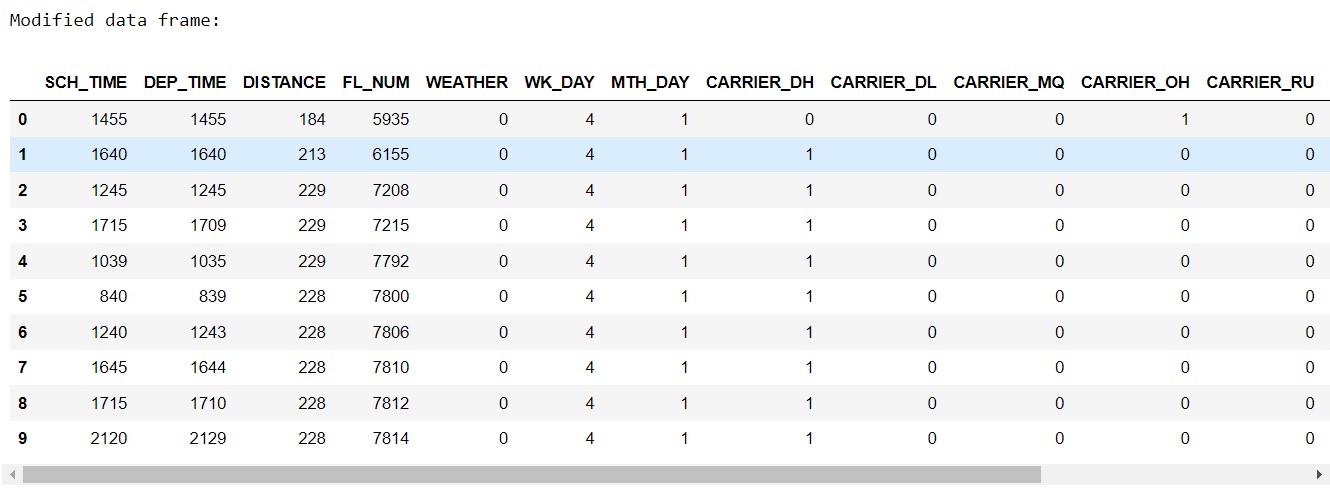
CARRIER\_RU uint8

CARRIER\_UA uint8

CARRIER\_US uint8

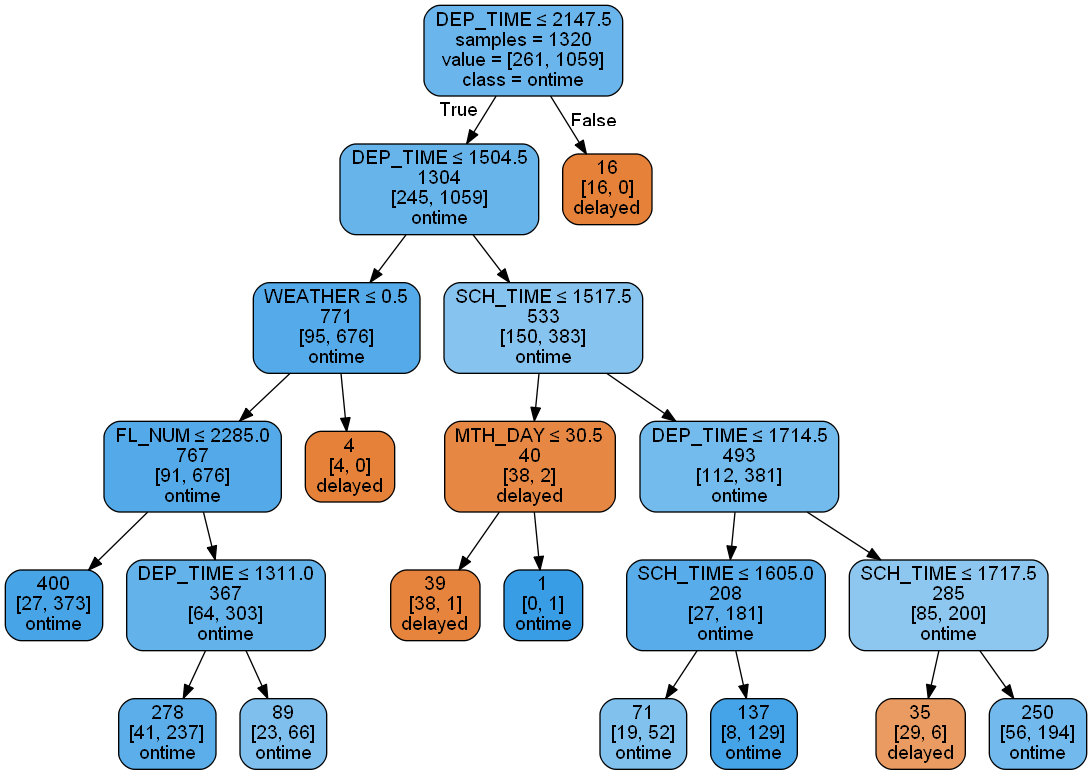
FL\_STATUS object

**d.** The first 10 records of the modified data frame are displayed below.

****

The outcome variable that we need to predict is flight arriving status i.e., “FL\_STATUS” using the 14 predictor variables such as 'SCH\_TIME', 'DEP\_TIME', 'DISTANCE', 'FL\_NUM', 'WEATHER', 'WK\_DAY', 'MTH\_DAY', 'CARRIER\_DH', 'CARRIER\_DL', 'CARRIER\_MQ', 'CARRIER\_OH', 'CARRIER\_RU', 'CARRIER\_UA', 'CARRIER\_US'.

1. a. The following figure displays a classification tree model using DecisionTreeClassifier() with the training data set and the following tree control parameters: (a) maximum depth (number of splits) equals 5; (b) minimum impurity decrease per split of 0.001; and (c) minimum number of node records (samples) to split equals to 10.



Number of nodes: 21

b. **Confusion matrices for smaller classification tree.**

**Training Partition for Smaller Tree**

Confusion Matrix (Accuracy 0.8629)

Prediction

Actual 0 1

0 87 174

1 7 1052

**Validation Partition for Smaller Tree**

Confusion Matrix (Accuracy 0.8558)

Prediction

Actual 0 1

0 54 113

1 4 700

**Training data set:**

Accuracy = 87+1052/1320 = 0.8629 = 86.29 %

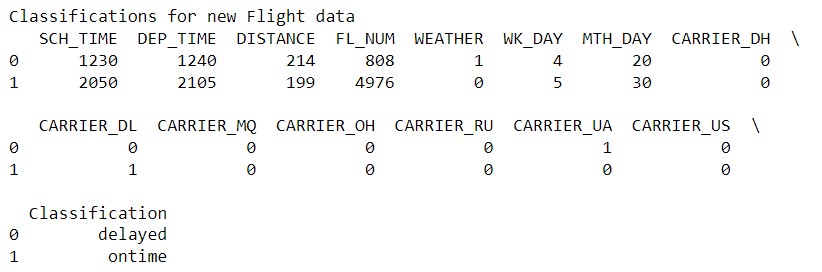
Misclassification = 1- 0.8629 = 0.1371 = 13.71 %

**Validation data set:**

Accuracy = 54+700/881 = 0.8558 = 85.58 %

Misclassification = 1- 0.8558 = 0.1442 = 14.42 %

There is not much difference in the accuracy scores of training and validation sets. We can say that there is no possibility of overfitting for this model. Hence this model can be used for making predictions.

c. s

For the first record the flight status (‘FL\_STATUS’) is classified as ‘delayed’ and for the second record, it is ‘on-time’.

**Classification of the First record:**

‘DEP\_TIME’ <= 2147.5 AND ‘DEP\_TIME’ <= 1504.5 AND ‘WEATHER’ >0.5 🡺 delayed

**Classification of the Second record:**

‘DEP\_TIME’ <= 2147.5 AND ‘DEP\_TIME’ > 1504.5 AND ‘SCH\_TIME’ > 1517.5 AND ‘DEP\_TIME’ > 1714.5 AND ‘SCH\_TIME’ > 1717.5 🡺 on-time

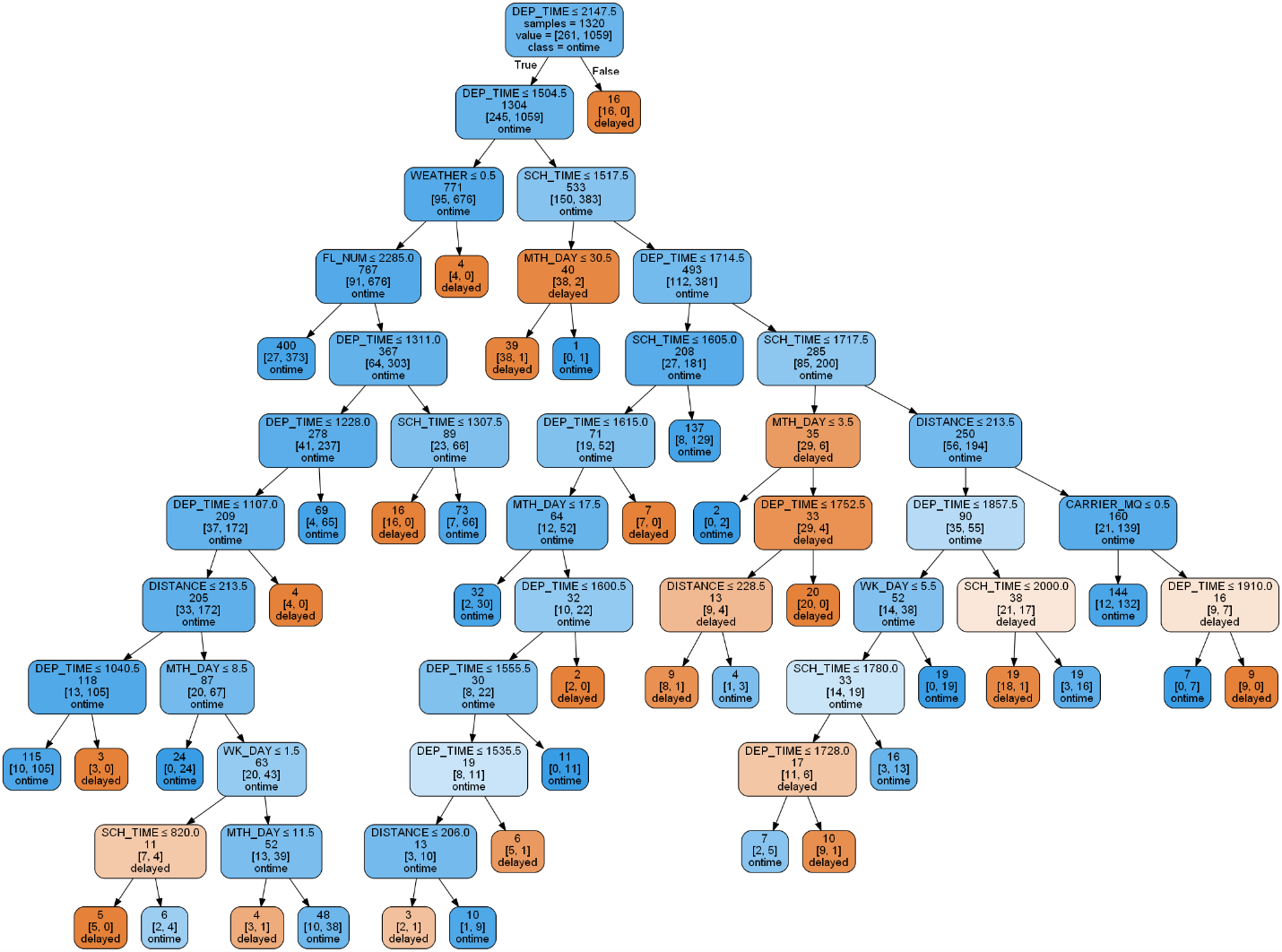
1. a. Using the GridSearchCV() algorithm in Python to improve (optimize) the classification tree and it is developed with control parameters as following: (a) maximum depth (number of splits) in the range from 2 to 30; (b) minimum impurity decrease per split of 0, 0.0005, and 0.001; and (c) minimum number of node records (samples) to split in the range from 5 to 30.

Each time we run the GridSearchCV() and Classification tree, the results are varying. The most frequently displayed result is shown here.

Improved score:0.8659

Improved parameters: {'max\_depth': 11, 'min\_impurity\_decrease': 0.001, 'min\_samples\_split': 11}

**Best Classification Tree with Grid Search:**



Number of nodes: 73

**Confusion matrices for grid search classification tree:**

**Training Partition**

Confusion Matrix (Accuracy 0.9250)

Prediction

Actual 0 1

0 169 92

1 7 1052

**Validation Partition**

Confusion Matrix (Accuracy 0.8785)

Prediction

Actual 0 1

0 90 77

1 30 684

**Training data set:**

Accuracy = 169+1052/1320 = 0.9250 = 92.50 %

Misclassification = 1- 0.9250 = 0.075 = 7.5 %

**Validation data set:**

Accuracy = 90+684/881 = 0.8785 = 87.85 %

Misclassification = 1- 0.8785 = 0.1215 = 12.15 %

The difference in the accuracy scores of training and validation data sets is not significant enough to claim the possibility of overfitting. We can say that there is no possibility of overfitting for this model. Hence this model can be used for making predictions.

b) S**maller classification tree**

**Training data set:**

Accuracy = 87+1052/1320 = 0.8629 = 86.29 %

Misclassification = 1- 0.8629 = 0.1371 = 13.71 %

**Validation data set:**

Accuracy = 54+700/881 = 0.8558 = 85.58 %

Misclassification = 1- 0.8558 = 0.1442 = 14.42 %

**Grid search classification tree**

**Training data set:**

Accuracy = 169+1052/1320 = 0.9250 = 92.50 %

Misclassification = 1- 0.9250 = 0.075 = 7.5 %

**Validation data set:**

Accuracy = 90+684/881 = 0.8785 = 87.85 %

Misclassification = 1- 0.8785 = 0.1215 = 12.15 %

Although there is no possibility of overfitting in both the classification trees, the accuracy rates for the Grid search classification tree are more for both the training and the validation sets compared to the accuracy rates of the Smaller classification tree. The Grid Search Classification is recommended for making predictions in this case of flight arrival status.

## Classification for the same new data using Grid Search classification tree

Text

Description automatically generated with medium confidence

Both the classification trees gave same results for the two new Flight records.