Case Study-3

1. Upload, explore, clean, and preprocess data.

a. Why a logistic regression model may be used in this case? Why may you not apply a multiple linear regression model in this case? Provide brief answers to both questions.

Explanation: -

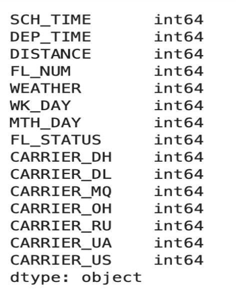
- Logistic regression is suitable in this scenario because the response variable (flight delay status) is categorical and binary, reflecting whether a flight is delayed ("Yes" or 1) or not delayed ("No" or 0). Logistic regression effectively estimates probabilities for binary outcomes.
- Multiple linear regression is not appropriate here because it requires the response variable to be continuous and normally distributed. Using linear regression with a categorical, binary response ("Yes"/"No" or 1/0) would produce meaningless predictions outside the binary range.

b. Create a flight_df data frame by uploading the original data set into Python. Remove 'DEST' and 'ORIGIN' variables from the flight_df data frame. Convert 'CARRIER' and 'FL_STATUS' into binary variables. This question 1b of part 1 will not be graded.

Explanation: -

The dataset has been successfully processed:

- Variables DEST and ORIGIN have been removed.
- Variables CARRIER and FL_STATUS have been converted into binary (dummy) variables.



```
SCH_TIME      int64
DEP_TIME      int64
DISTANCE      int64
FL_NUM        int64
WEATHER       int64
WK_DAY        int64
MTH_DAY       int64
FL_STATUS     int64
CARRIER_DH    int64
CARRIER_DL    int64
CARRIER_MQ    int64
CARRIER_OH    int64
CARRIER_RU    int64
CARRIER_UA    int64
CARRIER_US    int64
dtype: object
```

c. Why does the output variable 'FL_STATUS' need to be converted into binary variables for logistic regression? Briefly explain

Explanation: -

The output variable FL_STATUS must be converted into a binary numeric format for logistic regression because logistic regression models the probability of a binary event (e.g., delayed or not delayed) numerically. This numeric encoding (typically 0 and 1) allows the algorithm to mathematically estimate the probability of each outcome clearly and efficiently.

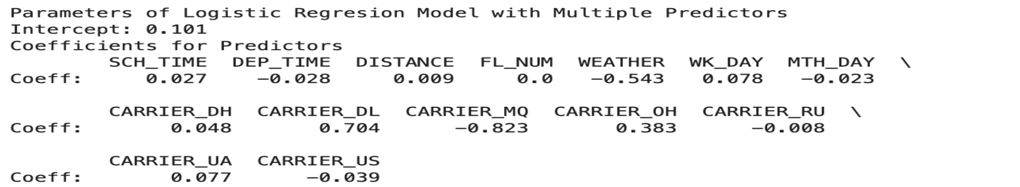
2. Develop a logistic regression model for the Flight Delays case.

a. Develop in Python the predictor variables (14 variables) and outcome variable ('FL_STATUS') and partition the data set (80% for training and 20% for validation partitions, random_state=1). Train a logistic regression model using LogisticRegression()3. with the training data set and display in Python the model's parameters (intercept and regression coefficients). Provide these parameters in your report and also present the mathematical equation of the trained logistic regression model.

Explanation:

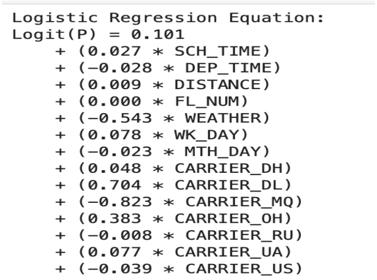
- A logistic regression model was trained to predict flight status based on multiple predictors.
- The model uses the liblinear solver and C=1e42.
- The intercept and coefficients for each predictor variable were extracted and displayed, showing their effects on the flight status prediction.

Intercept and regression coefficients:



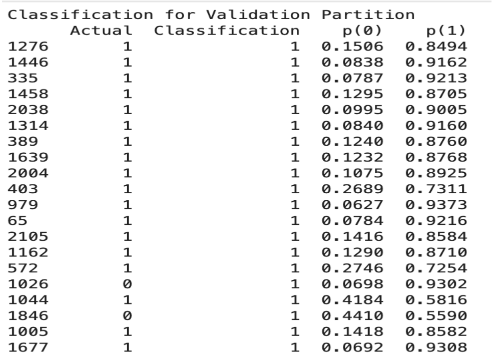
```
Parameters of Logistic Regresion Model with Multiple Predictors
Intercept: 0.101
Coefficients for Predictors
       SCH_TIME  DEP_TIME  DISTANCE  FL_NUM  WEATHER  WK_DAY  MTH_DAY  \
Coeff:    0.027    -0.028     0.009     0.0   -0.543   0.078   -0.023

       CARRIER_DH  CARRIER_DL  CARRIER_MQ  CARRIER_OH  CARRIER_RU  \
Coeff:      0.048       0.704      -0.823       0.383      -0.008

       CARRIER_UA  CARRIER_US
Coeff:      0.077      -0.039
```

Mathematical equation of the trained logistic regression model:



```
Logistic Regression Equation:
Logit(P) = 0.101
    + (0.027 * SCH_TIME)
    + (-0.028 * DEP_TIME)
    + (0.009 * DISTANCE)
    + (0.000 * FL_NUM)
    + (-0.543 * WEATHER)
    + (0.078 * WK_DAY)
    + (-0.023 * MTH_DAY)
    + (0.048 * CARRIER_DH)
    + (0.704 * CARRIER_DL)
    + (-0.823 * CARRIER_MQ)
    + (0.383 * CARRIER_OH)
    + (-0.008 * CARRIER_RU)
    + (0.077 * CARRIER_UA)
    + (-0.039 * CARRIER_US)
```

b. In Python, make predictions and identify probabilities p(0) and p(1) for the validation data set. For the first 20 records in the validation data set, display a table that contains the actual and predicted flight arrival status, and probabilities p(0) and p(1). Present this table in your report, and comment on the predicted vs. actual flight arrival status.

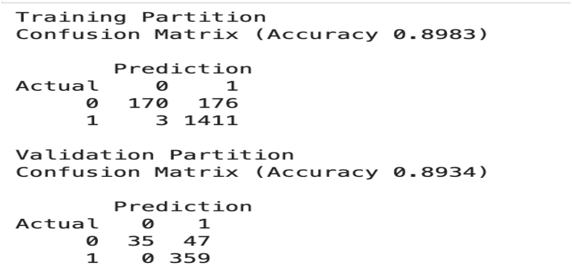


```
Classification for Validation Partition
      Actual  Classification    p(0)    p(1)
1276       1               1  0.1506  0.8494
1446       1               1  0.0838  0.9162
335        1               1  0.0787  0.9213
1458       1               1  0.1295  0.8705
2038       1               1  0.0995  0.9005
1314       1               1  0.0840  0.9160
389        1               1  0.1240  0.8760
1639       1               1  0.1232  0.8768
2004       1               1  0.1075  0.8925
403        1               1  0.2689  0.7311
979        1               1  0.0627  0.9373
65         1               1  0.0784  0.9216
2105       1               1  0.1416  0.8584
1162       1               1  0.1290  0.8710
572        1               1  0.2746  0.7254
1026       0               1  0.0698  0.9302
1044       1               1  0.4184  0.5816
1846       0               1  0.4410  0.5590
1005       1               1  0.1418  0.8582
1677       1               1  0.0692  0.9308
```

Explanation: -

- The logistic regression model predicted mostly correctly (with predicted and actual status matching) for these first 20 validation records.
- However, there are 2 misclassifications (indexes 1026 and 1846), where flights were actually delayed(Actual = 0), but the model predicted them as ontime (Predicted = 1).
- This suggests that the model might struggle to correctly classify delayed flights when the predicted probability of being delayed (p(0)) is lower than 0.5.

c. Identify and display in Python confusion matrices for the training and validation partitions. Present them in your report and comment on accuracy (misclassification) rate for both partitions and explain if there is a possibility of overfitting.



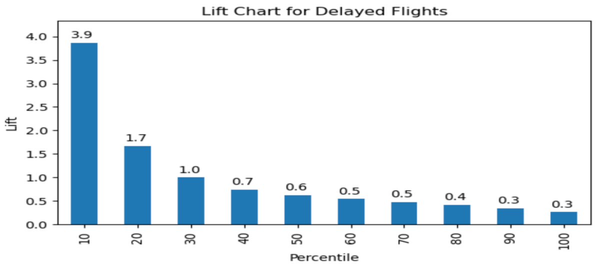
```
Training Partition
Confusion Matrix (Accuracy 0.8983)

       Prediction
Actual    0    1
     0  170  176
     1    3 1411

Validation Partition
Confusion Matrix (Accuracy 0.8934)

       Prediction
Actual   0   1
     0  35  47
     1   0 359
```

Explanation: -

Training Accuracy**:** 89.83%,
Misclassification for training data is : (1−0.8983)×100=10.17%
Validation Accuracy: 89.34%
Misclassification Rate = (1−0.8934)×100=10.66%

- Accuracy: The model performed well with high accuracy on both the training (89.83%) and validation (89.34%) sets.
- Misclassification Rates: The model misclassified about 10% of the cases in both partitions.
- Overfitting: The small gap between training and validation accuracy suggests good generalization and no overfitting.

d. Create and display in Python the Lift chart only for ‘delayed’ flight status. For that, use p(0) for .sort_values() and p(0) in liftChart(). Also use ncols=1 in plt.subplots() for a single plot and remove ax=axes[1] from liftChart(). Present this Lift chart in your report and briefly explain what the chart demonstrates and what conclusion(s) can be made.



Explanation: -

- The lift chart for delayed flights demonstrates that the logistic regression model effectively identifies delayed flights, especially within the top percentiles. In the top 10% of predictions, the model achieves a lift of 3.9, meaning it is 3.9 times more effective at detecting delayed flights compared to random guessing.
- The lift value remains consistently strong, with a lift of 1.7 in the top 20% of predictions, and 1.0 at the 30% percentile, indicating that the model is effective at identifying delayed flights in these ranges.
- After the 30th percentile, the lift gradually declines, and beyond the 70th percentile, the model's predictive power drops to random guessing (lift ≈ 1.0), indicating diminishing returns as you move to lower percentiles.

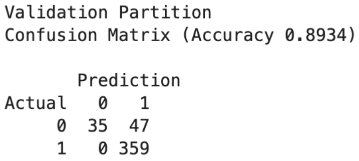
Conclusion:

This chart confirms that the model is most useful in identifying delayed flights in the top 10–30% of predictions. Airlines and operations managers can use this insight to focus on the highest-risk flights for better resource allocation and proactive management. The lift chart clearly shows that the model provides meaningful value in identifying delayed flights, particularly in the top-ranked predictions.

3. Compare results of logistic regression model vs. classification tree model for the same data set.

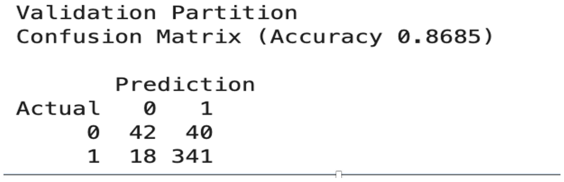
a. Present and compare in your report the validation confusion matrix for the logistic regression model in 2c of this case versus the validation confusion matrix using the GridSearchCV() algorithm for the classification tree in the previous case study #2. Using the accuracy value (misclassification rate), which model would you recommend applying for classification (prediction) of flight arrival status? Briefly explain your answer.

Logisticregression():



```
Validation Partition
Confusion Matrix (Accuracy 0.8934)

       Prediction
Actual   0   1
     0  35  47
     1   0 359
```

GridSearchCV():



```
Validation Partition
Confusion Matrix (Accuracy 0.8685)

       Prediction
Actual   0   1
     0  42  40
     1  18 341
```

Explanation: -

Logistic Regression Model:

- Higher accuracy (89.34%).
- Lower misclassification rate (10.66%).

Classification Tree (GridSearchCV):

- Lower accuracy (86.85%).
- Higher misclassification rate (13.15%).

Recommendation:

Based on the validation confusion matrices, the Logistic Regression model demonstrates better performance with a higher accuracy (89.34%) and a lower misclassification rate (10.66%) compared to the Classification Tree (GridSearchCV) model, which achieved an accuracy of 86.85% and misclassification 13.15%.

Therefore, the Logistic Regression model is recommended for the classification (prediction) of flight arrival status, as it shows a more reliable and balanced predictive performance.