03_Merge_Notebook

April 15, 2022

0.1 Predicting Airline Delays

Notebook: File Merge

Team: Jimmy Nguyen, Maha Jayapal, Roberto Cancel

[1]:

```
!pip install --upgrade numpy #ensure numpy and pandas are upgraded to same␣
 ↪versions for easier exploration (avoiding errors)
!pip install --upgrade pandas #ensure numpy and pandas are upgraded to same␣
 ↪versions for easier exploration (avoiding errors)

# IMPORT LIBRARIES REQUIRED THROUGHOUT THE NOTEBOOK
import boto3 # AWS SDK for Python
import pandas as pd # for importing and manipulating data
import numpy as np
import io # for encoding issues with raw data sets
from io import StringIO # converting dataframe to csv and uploading to s3␣
 ↪bucket /tranformed folder
```

```
/opt/conda/lib/python3.7/site-packages/secretstorage/dhcrypto.py:16:
CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_bytes
instead
  from cryptography.utils import int_from_bytes
/opt/conda/lib/python3.7/site-packages/secretstorage/util.py:25:
CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_bytes
instead
  from cryptography.utils import int_from_bytes
Requirement already satisfied: numpy in /opt/conda/lib/python3.7/site-packages
(1.21.5)
WARNING: Running pip as the 'root' user can result in broken permissions
and conflicting behaviour with the system package manager. It is recommended to
use a virtual environment instead: https://pip.pypa.io/warnings/venv
WARNING: You are using pip version 21.3.1; however, version 22.0.4 is
available.
You should consider upgrading via the '/opt/conda/bin/python -m pip install
--upgrade pip' command.
```

```
/opt/conda/lib/python3.7/site-packages/secretstorage/dhcrypto.py:16:
CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_bytes
instead
  from cryptography.utils import int_from_bytes
/opt/conda/lib/python3.7/site-packages/secretstorage/util.py:25:
CryptographyDeprecationWarning: int_from_bytes is deprecated, use int.from_bytes
instead
  from cryptography.utils import int_from_bytes
Requirement already satisfied: pandas in /opt/conda/lib/python3.7/site-packages
(1.3.5)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/lib/python3.7/site-
packages (from pandas) (2019.3)
Requirement already satisfied: numpy>=1.17.3 in /opt/conda/lib/python3.7/site-
packages (from pandas) (1.21.5)
Requirement already satisfied: python-dateutil>=2.7.3 in
/opt/conda/lib/python3.7/site-packages (from pandas) (2.8.1)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.7/site-
packages (from python-dateutil>=2.7.3->pandas) (1.14.0)
WARNING: Running pip as the 'root' user can result in broken permissions
and conflicting behaviour with the system package manager. It is recommended to
use a virtual environment instead: https://pip.pypa.io/warnings/venv
WARNING: You are using pip version 21.3.1; however, version 22.0.4 is
available.
You should consider upgrading via the '/opt/conda/bin/python -m pip install
--upgrade pip' command.
```

[2]:

```python
# IDENTIFY FILES IN S3 BUCKET
session = boto3.Session()

#Then use the session to get the resource
s3 = session.resource('s3')

my_bucket = s3.Bucket('ads-508-airline')

for my_bucket_object in my_bucket.objects.all():
    print(my_bucket_object.key)
```

```
merged/
merged/Dec_EDA.csv
merged/Dec_merged.csv
merged/Dec_modeling.csv
raw/
raw/B43_AIRCRAFT_INVENTORY.csv
raw/CARRIER_DECODE.csv
raw/ONTIME_REPORTING_12.csv
```

```
raw/P10_EMPLOYEES.csv
raw/airport_weather_dec_2019.csv
raw/airports_list.csv
transformed/
transformed/B43_AIRCRAFT_INVENTORY.csv
transformed/CARRIER_DECODE.csv
transformed/ON_TIME_REPORTING_12.csv
transformed/P10_EMPLOYEES.csv
transformed/airport_weather_dec_2019.csv
transformed/airports_list.csv
```

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx INGEST TRANSFORMED DATA SETS xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

[3]:
```
# INGEST FLIGHT DATA

s3_client = boto3.client("s3")

BUCKET='ads-508-airline'
KEY='transformed/ON_TIME_REPORTING_12.csv'

response = s3_client.get_object(Bucket=BUCKET, Key=KEY)
dec_flight = pd.read_csv(response.get("Body"))
dec_flight.head()
```

[3]:
```
   DAY_OF_MONTH  DAY_OF_WEEK OP_UNIQUE_CARRIER TAIL_NUM  ORIGIN_AIRPORT_ID  \
0             8            7                WN   N8651A              15016
1             8            7                WN   N939WN              15016
2             8            7                WN   N7741C              15016
3             8            7                WN   N550WN              15016
4             8            7                WN   N8319F              15016

  ORIGIN DEST  DEP_DEL15 DEP_TIME_BLK ARR_TIME_BLK  CANCELLED  \
0    STL  SAN        0.0    1100-1159    1300-1359        0.0
1    STL  SAT        0.0    1200-1259    1400-1459        0.0
2    STL  SAT        0.0    2100-2159    0001-0559        0.0
3    STL  SEA        0.0    0900-0959    1200-1259        0.0
4    STL  SFO        1.0    1800-1859    2000-2059        0.0

   CRS_ELAPSED_TIME  DISTANCE  DISTANCE_GROUP  CARRIER_DELAY  WEATHER_DELAY  \
0             245.0    1557.0               7            0.0            0.0
1             145.0     786.0               4            NaN            NaN
2             140.0     786.0               4            NaN            NaN
3             275.0    1709.0               7            NaN            NaN
4             270.0    1735.0               7            NaN            NaN

   NAS_DELAY  SECURITY_DELAY  LATE_AIRCRAFT_DELAY
0       18.0             0.0                  0.0
```

```
1        NaN             NaN                    NaN
2        NaN             NaN                    NaN
3        NaN             NaN                    NaN
4        NaN             NaN                    NaN
```

[4]:

```
# INGEST AIRCRAFT DATA - raw data that requires encoding='latin1'

KEY='transformed/B43_AIRCRAFT_INVENTORY.csv'

response = s3_client.get_object(Bucket=BUCKET, Key=KEY)
s3_data = io.BytesIO(response.get('Body').read())
aircraft = pd.read_csv(s3_data, encoding='latin1')
aircraft.head()
```

[4]:

```
   MANUFACTURE_YEAR TAIL_NUM  NUMBER_OF_SEATS
0              1944   N54514              0.0
1              1945   N1651M              0.0
2              1953   N100CE              0.0
3              1953   N141FL              0.0
4              1953   N151FL              0.0
```

[5]:

```
# INGEST CARRIER NAMES DICTIONARY

KEY='transformed/CARRIER_DECODE.csv'

response = s3_client.get_object(Bucket=BUCKET, Key=KEY)
names = pd.read_csv(response.get("Body"))
names.head()
```

[5]:

```
   AIRLINE_ID OP_UNIQUE_CARRIER  CARRIER_NAME
0       21754               2PQ    21 Air LLC
1       20342                Q5   40-Mile Air
2       20342               WRB   40-Mile Air
3       19627               CIQ    A/S Conair
4       19072               AAE  AAA Airlines
```

[6]:

```
# INGEST CARRIER EMPLOYEE / STAFFING DATA

KEY='transformed/P10_EMPLOYEES.csv'

response = s3_client.get_object(Bucket=BUCKET, Key=KEY)
employees = pd.read_csv(response.get("Body"))
employees.head()
```

[6]:

```
  OP_UNIQUE_CARRIER  PILOTS_COPILOTS  PASSENGER_HANDLING  PASS_GEN_SVC_ADMIN  \
0               0WQ               53                   0                  19
1               2HQ               32                   0                  24
```

```
2               3EQ               29                   0                  32
3                5V               54                   0                   0
4                5X             1882                   0                   0

   MAINTENANCE
0           29
1           32
2            2
3          104
4         1383
```

[7]:
```
# INGEST DECEMBER 2019 DAILY WEATHER OBSERVATIONS

KEY='transformed/airport_weather_dec_2019.csv'

response = s3_client.get_object(Bucket=BUCKET, Key=KEY)
weather_report = pd.read_csv(response.get("Body"))
weather_report.head()
```

[7]:
```
        DATE                                              NAME  PRCP  SNOW  \
0  12/1/2019  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  0.04   0.0
1  12/2/2019  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  0.00   0.0
2  12/3/2019  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  0.00   0.0
3  12/4/2019  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  0.00   0.0
4  12/5/2019  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  0.00   0.0

   SNWD  TMAX   AWND
0   0.0  67.0  16.11
1   0.0  48.0  16.78
2   0.0  49.0  11.18
3   0.0  60.0  11.18
4   0.0  65.0   5.82
```

[8]:
```
# INGEST CITY AND AIRPORT NAME DICTIONARY

KEY='transformed/airports_list.csv'

response = s3_client.get_object(Bucket=BUCKET, Key=KEY)
cities = pd.read_csv(response.get("Body"))
cities.head()
```

[8]:
```
   ORIGIN_AIRPORT_ID               DISPLAY_AIRPORT_NAME ORIGIN_CITY_NAME  \
0              12992                        Adams Field  Little Rock, AR
1              10257                Albany International       Albany, NY
2              10140  Albuquerque International Sunport  Albuquerque, NM
3              10299             Anchorage International    Anchorage, AK
4              10397                  Atlanta Municipal      Atlanta, GA
```

```
                                                NAME
0                   NORTH LITTLE ROCK AIRPORT, AR US
1                ALBANY INTERNATIONAL AIRPORT, NY US
2           ALBUQUERQUE INTERNATIONAL AIRPORT, NM US
3  ANCHORAGE TED STEVENS INTERNATIONAL AIRPORT, A…
4  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…
```

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx CLEAN AND PREPARE SUPPLEMENTARY DATA SETS FOR MERGING xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

[9]:
```python
# Drop Duplicates to retain a dictionary
aircraft.drop_duplicates(subset='TAIL_NUM', inplace=True)
aircraft.head()
```

[9]:
```
   MANUFACTURE_YEAR TAIL_NUM  NUMBER_OF_SEATS
0              1944   N54514              0.0
1              1945   N1651M              0.0
2              1953   N100CE              0.0
3              1953    N141FL             0.0
4              1953    N151FL             0.0
```

[10]:
```python
# Drop Duplicates to retain a dictionary
names.drop_duplicates(subset='OP_UNIQUE_CARRIER', inplace=True)
names = names.reset_index(drop=True)
names
```

[10]:
```
      AIRLINE_ID OP_UNIQUE_CARRIER           CARRIER_NAME
0          21754               2PQ             21 Air LLC
1          20342                Q5            40-Mile Air
2          20342               WRB            40-Mile Air
3          19627               CIQ             A/S Conair
4          19072               AAE           AAA Airlines
…          …               …                   …
1739       20379               ZKQ   Zantop International
1740       19771               ZAQ  Zas Airline Of Egypt
1741       21118                37               Zeal 320
1742       22069                ZG      ZIPAIR Tokyo Inc.
1743       19894               ZUQ     Zuliana De Aviacion

[1744 rows x 3 columns]
```

[11]:
```python
# Weather clean up missing values
weather_report.isna().sum()
```

[11]:
```
DATE       0
NAME       0
PRCP       0
SNOW    1004
SNWD    1055
TMAX       2
AWND      31
dtype: int64
```

[12]:
```
# Impute missing snow values with 0 for days with 0 preciptation
weather_report.SNOW.where((~weather_report.SNOW.isna()) & (weather_report.PRCP !
 ↪= '0.00'),
 '0.00', inplace=True)

# Impute missing snow depth (SNWD) values with 0 for days with no snow
weather_report.SNWD.where((~weather_report.SNWD.isna()) & (weather_report.SNOW !
 ↪= '0.00'),
 '0.00', inplace=True)

# Re-evaluate missingness of weather data
weather_report.isna().sum()
```

[12]:
```
DATE     0
NAME     0
PRCP     0
SNOW     0
SNWD     0
TMAX     2
AWND    31
dtype: int64
```

DISCUSSION POINTS: Need to address how we will handle missing TMAX, and AWND values One Option is to merge as is and then drop observations with missing values for weather data

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx MERGE CLEANED AND TRANSFORMED DATA SETS xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

[13]:
```
# Merge weather report and cities for future merging
weather = pd.merge(cities, weather_report, how='left', on='NAME')
weather.head()
```

[13]:
```
   ORIGIN_AIRPORT_ID DISPLAY_AIRPORT_NAME ORIGIN_CITY_NAME  \
0              12992          Adams Field  Little Rock, AR
1              12992          Adams Field  Little Rock, AR
2              12992          Adams Field  Little Rock, AR
3              12992          Adams Field  Little Rock, AR
4              12992          Adams Field  Little Rock, AR
```

```
                             NAME       DATE  PRCP  SNOW  SNWD  TMAX   AWND
0  NORTH LITTLE ROCK AIRPORT, AR US  12/1/2019  0.00   0.0   0.0  53.0  10.07
1  NORTH LITTLE ROCK AIRPORT, AR US  12/2/2019  0.00   0.0   0.0  47.0   4.03
2  NORTH LITTLE ROCK AIRPORT, AR US  12/3/2019  0.00   0.0   0.0  56.0   4.03
3  NORTH LITTLE ROCK AIRPORT, AR US  12/4/2019  0.00   0.0   0.0  62.0   2.91
4  NORTH LITTLE ROCK AIRPORT, AR US  12/5/2019  0.01   0.0   0.0  65.0   4.92
```

[14]:

```
# Drop redundant features used for merging
weather_no = ['DISPLAY_AIRPORT_NAME', 'ORIGIN_CITY_NAME', 'NAME']
weather.drop(weather_no, inplace=True, axis=1)
```

[15]:

```
# Change date format to match flight data (Month & Day_of_Month)
weather['DATE'] = pd.to_datetime(weather['DATE'])
weather['MONTH'] = pd.DatetimeIndex(weather['DATE']).month
weather['DAY_OF_MONTH'] = pd.DatetimeIndex(weather['DATE']).day
# Drop the redundant Date features
date_no = ['DATE', 'MONTH']
weather.drop(date_no, inplace=True, axis=1)
```

[16]:

```
# Merge aircraft information into flight data file
Dec_flight = pd.merge(dec_flight, aircraft, how="left", on='TAIL_NUM')
```

[17]:

```
# Merge names and flight data for consistent airline names
Dec_flight2 = pd.merge(Dec_flight, names, how='left', on=['OP_UNIQUE_CARRIER'])
```

[18]:

```
# Merge Employee
Dec_flight3 = pd.merge(Dec_flight2, employees, how='left',␣
 ↪on=['OP_UNIQUE_CARRIER'])
Dec_flight3.head()
```

[18]:

```
   DAY_OF_MONTH  DAY_OF_WEEK OP_UNIQUE_CARRIER TAIL_NUM  ORIGIN_AIRPORT_ID  \
0             8            7                WN   N8651A              15016
1             8            7                WN   N939WN              15016
2             8            7                WN   N7741C              15016
3             8            7                WN   N550WN              15016
4             8            7                WN   N8319F              15016

  ORIGIN DEST  DEP_DEL15 DEP_TIME_BLK ARR_TIME_BLK  …  SECURITY_DELAY  \
0    STL  SAN        0.0    1100-1159    1300-1359  …             0.0
1    STL  SAT        0.0    1200-1259    1400-1459  …             NaN
2    STL  SAT        0.0    2100-2159    0001-0559  …             NaN
3    STL  SEA        0.0    0900-0959    1200-1259  …             NaN
4    STL  SFO        1.0    1800-1859    2000-2059  …             NaN

   LATE_AIRCRAFT_DELAY  MANUFACTURE_YEAR  NUMBER_OF_SEATS  AIRLINE_ID  \
0                  0.0            2014.0            175.0       19393
1                  NaN            2009.0            143.0       19393
```

```
2                  NaN            2007.0            143.0       19393
3                  NaN            2001.0            143.0       19393
4                  NaN            2012.0            175.0       19393

             CARRIER_NAME  PILOTS_COPILOTS  PASSENGER_HANDLING  \
0  Southwest Airlines Co.             8989                9668
1  Southwest Airlines Co.             8989                9668
2  Southwest Airlines Co.             8989                9668
3  Southwest Airlines Co.             8989                9668
4  Southwest Airlines Co.             8989                9668

   PASS_GEN_SVC_ADMIN  MAINTENANCE
0               15475         2482
1               15475         2482
2               15475         2482
3               15475         2482
4               15475         2482

[5 rows x 27 columns]
```

[19]:

```
# Merge Weather
Dec_merged = pd.merge(Dec_flight3, weather, how='left',␣
 ↪on=['ORIGIN_AIRPORT_ID', 'DAY_OF_MONTH'])
```

[20]:

```
Dec_merged
```

[20]:

```
        DAY_OF_MONTH  DAY_OF_WEEK OP_UNIQUE_CARRIER TAIL_NUM  \
0                  8            7                WN   N8651A
1                  8            7                WN   N939WN
2                  8            7                WN   N7741C
3                  8            7                WN   N550WN
4                  8            7                WN   N8319F
…              …          …               …      …
625758            31            2                B6   N193JB
625759            31            2                B6   N304JB
625760            31            2                B6   N193JB
625761            31            2                B6   N563JB
625762            31            2                B6   N907JB

        ORIGIN_AIRPORT_ID ORIGIN DEST  DEP_DEL15 DEP_TIME_BLK ARR_TIME_BLK  \
0                   15016    STL  SAN        0.0    1100-1159    1300-1359
1                   15016    STL  SAT        0.0    1200-1259    1400-1459
2                   15016    STL  SAT        0.0    2100-2159    0001-0559
3                   15016    STL  SEA        0.0    0900-0959    1200-1259
4                   15016    STL  SFO        1.0    1800-1859    2000-2059
…                   …    …  …        …          …          …
625758              13204    MCO  SWF        1.0    1300-1359    1600-1659
```

```
625759              11278    DCA  BOS        0.0    1400-1459    1500-1559
625760              14100    PHL  BOS        0.0    0700-0759    0800-0859
625761              10721    BOS  SJU        0.0    0800-0859    1300-1359
625762              14843    SJU  BOS        0.0    0001-0559    0700-0759

        …            CARRIER_NAME  PILOTS_COPILOTS  PASSENGER_HANDLING  \
0       …  Southwest Airlines Co.             8989                9668
1       …  Southwest Airlines Co.             8989                9668
2       …  Southwest Airlines Co.             8989                9668
3       …  Southwest Airlines Co.             8989                9668
4       …  Southwest Airlines Co.             8989                9668
…     …                     …              …                 …
625758  …         JetBlue Airways             2840                4905
625759  …         JetBlue Airways             2840                4905
625760  …         JetBlue Airways             2840                4905
625761  …         JetBlue Airways             2840                4905
625762  …         JetBlue Airways             2840                4905

        PASS_GEN_SVC_ADMIN  MAINTENANCE  PRCP  SNOW  SNWD  TMAX   AWND
0                    15475         2482  0.02   0.0   0.0  58.0   9.84
1                    15475         2482  0.02   0.0   0.0  58.0   9.84
2                    15475         2482  0.02   0.0   0.0  58.0   9.84
3                    15475         2482  0.02   0.0   0.0  58.0   9.84
4                    15475         2482  0.02   0.0   0.0  58.0   9.84
…                    …          …   …   …   …   …    …
625758                3888          726  0.00   0.0   0.0  71.0   8.28
625759                3888          726  0.00   0.0   0.0  53.0   7.61
625760                3888          726  0.00   0.0   0.0  47.0   9.17
625761                3888          726  0.27   0.0  0.00  44.0  13.42
625762                3888          726   NaN   NaN   NaN   NaN    NaN

[625763 rows x 32 columns]
```

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
MERGED FEATURES: NAME, DESCRIPTION, DROP RATIONALE (IF APPROPRIATE) xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

DAY_OF_MONTH - Day of the Month DAY_OF_WEEK - Day of the Week - 1= MON, 2=TUES, 3=WED, 4=THUR, 5=FRI, 6=SAT, 7=SUN, 9=UNKNOWN OP_UNIQUE_CARRIER - Unique Carrier Code (ID) - NEEDED TO MERGE WITH EMPLOYEES AND NAMES TAIL_NUM - Aircraft Tail # (ID) - NEEDED TO MERGE WITH AIRCRAFT ORIGIN_AIRPORT_ID - Departure Airport (NEEDED TO MERGE) ORIGIN - Departure Airport ID (LINKS TO WEATHER STATIONS) DEST - Arrival Airport ID (LINKS TO WEATHER STATIONS) DEP_DEL15 - Departure Delay Indicator, 15 minutes or more(1=YES) - **TARGET FEATURE** DEP_TIME_BLK - Scheduled Departure Time block, Hourly interval ARR_TIME_BLK - Scheduled Arrival Time Block, HOURLY INTERVALS CANCELLED - Cancelled Flight Indicator (1=YES) CRS_ELAPSED_TIME - Scheduled Elapsed Time DISTANCE - Distance Traveled in Miles DISTANCE_GROUP - Distance Traveled block in increments of 250 miles (1 < 250, 2 = 250-499, …, 11=2500+ miles CARRIER_DELAY - Carrier Delay in Minutes WEATHER_DELAY - Extreme Weather Delay in Minutes NAS_DELAY - National Air System Delay in Minutes SECURITY_DELAY - Security Delay in Minutes LATE_AIRCRAFT_DELAY - Late Arrival Delay in Minutes MANUFACTURE_YEAR - Year plane was manufactured NUMBER_OF_SEATS - Plane seat capacity AIRLINE_ID - Airline identifier needed for merging CARRIER_NAME - Carrier ID PILOTS_COPILOTS - Airline specific count of pilots/copilots PASSENGER_HANDLING - Airline Specific count of flight attendants PASS_GEN_SVC_ADMIN - Airline Specific count of ground personnel MAINTENANCE - Airline Specific count of maintenance personnel PRCP - Daily preciptation value SNOW - Daily falling snow value SNWD - Daily snow depth value TMAX - Daily max temperature AWND - Daily average wind speed

[21]:

```
# Drop flights with missing target variables
Dec_merged.drop(Dec_merged.loc[Dec_merged['DEP_DEL15'].isna()].index, axis=0,␣
 ↪inplace=True)

# Drop cancelled flights
Dec_merged.drop(Dec_merged[Dec_merged['CANCELLED'] == 1].index, inplace = True)
```

[22]:

```
Dec_merged.isna().sum()
```

[22]:

```
DAY_OF_MONTH                0
DAY_OF_WEEK                 0
OP_UNIQUE_CARRIER           0
TAIL_NUM                    0
ORIGIN_AIRPORT_ID           0
ORIGIN                      0
DEST                        0
DEP_DEL15                   0
DEP_TIME_BLK                0
ARR_TIME_BLK                0
CANCELLED                   0
CRS_ELAPSED_TIME            0
DISTANCE                    0
DISTANCE_GROUP              0
CARRIER_DELAY          493025
WEATHER_DELAY          493025
NAS_DELAY              493025
SECURITY_DELAY         493025
LATE_AIRCRAFT_DELAY    493025
MANUFACTURE_YEAR        23498
NUMBER_OF_SEATS         23498
AIRLINE_ID                  0
CARRIER_NAME                0
PILOTS_COPILOTS             0
PASSENGER_HANDLING          0
PASS_GEN_SVC_ADMIN          0
```

```
MAINTENANCE                0
PRCP                   61808
SNOW                   61808
SNWD                   61808
TMAX                   61944
AWND                   61808
dtype: int64
```

[23]:

```
# Drop observations with missing TMAX values
Dec_merged.drop(Dec_merged.loc[Dec_merged['TMAX'].isna()].index, axis=0,␣
 ↪inplace=True)
```

[24]:

```
Dec_merged.isna().sum()
```

[24]:

```
DAY_OF_MONTH               0
DAY_OF_WEEK                0
OP_UNIQUE_CARRIER          0
TAIL_NUM                   0
ORIGIN_AIRPORT_ID          0
ORIGIN                     0
DEST                       0
DEP_DEL15                  0
DEP_TIME_BLK               0
ARR_TIME_BLK               0
CANCELLED                  0
CRS_ELAPSED_TIME           0
DISTANCE                   0
DISTANCE_GROUP             0
CARRIER_DELAY         443393
WEATHER_DELAY         443393
NAS_DELAY             443393
SECURITY_DELAY        443393
LATE_AIRCRAFT_DELAY   443393
MANUFACTURE_YEAR       15206
NUMBER_OF_SEATS        15206
AIRLINE_ID                 0
CARRIER_NAME               0
PILOTS_COPILOTS            0
PASSENGER_HANDLING         0
PASS_GEN_SVC_ADMIN         0
MAINTENANCE                0
PRCP                       0
SNOW                       0
SNWD                       0
TMAX                       0
AWND                       0
dtype: int64
```

[25]:

```
# Save merged data for EDA to merged folder in bucket
csv_buffer=StringIO()
Dec_merged.to_csv(csv_buffer, index=False)

BUCKET_NAME = 'ads-508-airline'
FileName= 'merged/Dec_EDA.csv'

s3csv = boto3.client('s3')

response=s3csv.put_object(Body=csv_buffer.getvalue(),
                           Bucket=BUCKET_NAME,
                           Key=FileName)
```

[26]:

```
# Drop redundant / irrelevant features

merged_no = ['CARRIER_DELAY','WEATHER_DELAY', 'NAS_DELAY', 'SECURITY_DELAY',␣
 ↪'LATE_AIRCRAFT_DELAY', 'MANUFACTURE_YEAR', 'NUMBER_OF_SEATS']
Dec_merged.drop(merged_no, inplace=True, axis=1)
```

[27]:

```
Dec_merged.head()
```

[27]:

```
   DAY_OF_MONTH  DAY_OF_WEEK OP_UNIQUE_CARRIER TAIL_NUM  ORIGIN_AIRPORT_ID  \
0             8            7                WN   N8651A              15016
1             8            7                WN   N939WN              15016
2             8            7                WN   N7741C              15016
3             8            7                WN   N550WN              15016
4             8            7                WN   N8319F              15016

  ORIGIN DEST  DEP_DEL15 DEP_TIME_BLK ARR_TIME_BLK  …  \
0    STL  SAN        0.0    1100-1159    1300-1359  …
1    STL  SAT        0.0    1200-1259    1400-1459  …
2    STL  SAT        0.0    2100-2159    0001-0559  …
3    STL  SEA        0.0    0900-0959    1200-1259  …
4    STL  SFO        1.0    1800-1859    2000-2059  …

             CARRIER_NAME  PILOTS_COPILOTS  PASSENGER_HANDLING  \
0  Southwest Airlines Co.             8989                9668
1  Southwest Airlines Co.             8989                9668
2  Southwest Airlines Co.             8989                9668
3  Southwest Airlines Co.             8989                9668
4  Southwest Airlines Co.             8989                9668

   PASS_GEN_SVC_ADMIN  MAINTENANCE  PRCP  SNOW  SNWD  TMAX  AWND
0               15475         2482  0.02   0.0   0.0  58.0  9.84
1               15475         2482  0.02   0.0   0.0  58.0  9.84
2               15475         2482  0.02   0.0   0.0  58.0  9.84
3               15475         2482  0.02   0.0   0.0  58.0  9.84
```

```
4               15475         2482  0.02    0.0   0.0  58.0  9.84

[5 rows x 25 columns]
```

[28]:

```
# Save merged data for modeling to merged folder in bucket
csv_buffer=StringIO()
Dec_merged.to_csv(csv_buffer, index=False)

BUCKET_NAME = 'ads-508-airline'
FileName= 'merged/Dec_modeling.csv'

s3csv = boto3.client('s3')

response=s3csv.put_object(Body=csv_buffer.getvalue(),
                           Bucket=BUCKET_NAME,
                           Key=FileName)
```

[ ]: