01_Raw_Data_Cleaning

April 15, 2022

0.1 Predicting Airline Delays

Notebook: Irrelevant Feature Reduction

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: python-dateutil>=2.7.3 in
/opt/conda/lib/python3.7/site-packages (from pandas) (2.8.1)
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: 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 DATA SETS xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

[3]:

```
# INGEST FLIGHT DATA

s3_client = boto3.client("s3")

BUCKET='ads-508-airline'
KEY='raw/ONTIME_REPORTING_12.csv'

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

[3]:

```
   MONTH  DAY_OF_MONTH  DAY_OF_WEEK OP_UNIQUE_CARRIER TAIL_NUM  \
0     12             8            7                WN   N8651A
1     12             8            7                WN   N939WN
2     12             8            7                WN   N7741C
3     12             8            7                WN   N550WN
4     12             8            7                WN   N8319F

   OP_CARRIER_FL_NUM  ORIGIN_AIRPORT_ID ORIGIN ORIGIN_CITY_NAME  \
0               3689              15016    STL    St. Louis, MO
1               2600              15016    STL    St. Louis, MO
2               2770              15016    STL    St. Louis, MO
3               6654              15016    STL    St. Louis, MO
4               3402              15016    STL    St. Louis, MO

   DEST_AIRPORT_ID  … CRS_ELAPSED_TIME ACTUAL_ELAPSED_TIME  DISTANCE  \
0            14679  …            245.0               266.0    1557.0
1            14683  …            145.0               125.0     786.0
2            14683  …            140.0               131.0     786.0
3            14747  …            275.0               256.0    1709.0
4            14771  …            270.0               256.0    1735.0

   DISTANCE_GROUP  CARRIER_DELAY  WEATHER_DELAY NAS_DELAY  SECURITY_DELAY  \
0               7            0.0            0.0      18.0             0.0
```

```
1               4            NaN            NaN       NaN             NaN
2               4            NaN            NaN       NaN             NaN
3               7            NaN            NaN       NaN             NaN
4               7            NaN            NaN       NaN             NaN

   LATE_AIRCRAFT_DELAY  Unnamed: 32
0                  0.0          NaN
1                  NaN          NaN
2                  NaN          NaN
3                  NaN          NaN
4                  NaN          NaN

[5 rows x 33 columns]
```

[4]:

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

KEY='raw/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]:

```python
# INGEST CARRIER NAMES DICTIONARY

KEY='raw/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       21754               2PQ   21 Air LLC
2       21754               2PQ   21 Air LLC
3       20342                Q5  40-Mile Air
4       20342               WRB  40-Mile Air
```

[6]:

```python
# INGEST CARRIER EMPLOYEE / STAFFING DATA
```

```
KEY='raw/P10_EMPLOYEES.csv'

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

[6]:

```
   YEAR  AIRLINE_ID OP_UNIQUE_CARRIER  \
0  2019       21352               0WQ
1  2019       21492               1BQ
2  2019       21712               2HQ
3  2019       21974               3EQ
4  2019       20408                5V

                                 UNIQUE_CARRIER_NAME CARRIER  \
0                                  Avjet Corporation     0WQ
1        Eastern Airlines f/k/a Dynamic Airways, LLC     1BQ
2                                  Elite Airways LLC     2HQ
3             Scott Aviation, LLC  d/b/a  Silver Air     3EQ
4  Tatonduk Outfitters Limited d/b/a Everts Air A…      5V

                                        CARRIER_NAME ENTITY  GENERAL_MANAGE  \
0                                  Avjet Corporation      D               4
1        Eastern Airlines f/k/a Dynamic Airways, LLC      I              14
2                                  Elite Airways LLC      D               9
3             Scott Aviation, LLC  d/b/a  Silver Air      D               0
4  Tatonduk Outfitters Limited d/b/a Everts Air A…      D              14

   PILOTS_COPILOTS  OTHER_FLT_PERS  …  GEN_ARCFT_TRAF_HANDLING  \
0               53               6  …                        0
1               50               0  …                        0
2               32               0  …                        0
3               29               0  …                        0
4               54              11  …                        0

   AIRCRAFT_CONTROL  PASSENGER_HANDLING  CARGO_HANDLING  TRAINEES_INTRUCTOR  \
0                 0                   0               3                   1
1                 0                   0               0                   1
2                 0                   0               0                   0
3                 0                   0               0                   0
4                 0                   0               0                   4

   STATISTICAL  TRAFFIC_SOLICITERS  OTHER  TRANSPORT_RELATED  TOTAL
0           18                   0      7                  0    161
1           13                   0      3                  0    161
2            7                   0      0                  0    123
3            0                   0      0                  0     69
4           45                   5     20                  0    347
```

```
[5 rows x 23 columns]
```

[7]:

```python
# INGEST DECEMBER 2019 DAILY WEATHER OBSERVATIONS

KEY='raw/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]:

```
       STATION                                              NAME       DATE  \
0  USW00013874  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  12/1/2019
1  USW00013874  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  12/2/2019
2  USW00013874  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  12/3/2019
3  USW00013874  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  12/4/2019
4  USW00013874  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  12/5/2019

    AWND  PGTM  PRCP  SNOW  SNWD  TAVG  TMAX  …  WT08  WT09  WESD  WT10  \
0  16.11   NaN  0.04   0.0   0.0  64.0  67.0  …   NaN   NaN   NaN   NaN
1  16.78   NaN  0.00   0.0   0.0  45.0  48.0  …   NaN   NaN   NaN   NaN
2  11.18   NaN  0.00   0.0   0.0  40.0  49.0  …   NaN   NaN   NaN   NaN
3  11.18   NaN  0.00   0.0   0.0  44.0  60.0  …   NaN   NaN   NaN   NaN
4   5.82   NaN  0.00   0.0   0.0  51.0  65.0  …   NaN   NaN   NaN   NaN

   PSUN  TSUN  SN32  SX32  TOBS  WT11
0   NaN   NaN   NaN   NaN   NaN   NaN
1   NaN   NaN   NaN   NaN   NaN   NaN
2   NaN   NaN   NaN   NaN   NaN   NaN
3   NaN   NaN   NaN   NaN   NaN   NaN
4   NaN   NaN   NaN   NaN   NaN   NaN

[5 rows x 32 columns]
```

[8]:

```python
# INGEST CITY AND AIRPORT NAME DICTIONARY

KEY='raw/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…
```

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx REMOVE REDUNDANT/IRRELEVANT DEC_FLIGHT FEATURES DATA xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Rationalization for dropped features:

MONTH - MONTH OF THE YEAR - ** DROP SINCE THIS STUDY IS ONLY FOR DECEMBER ** OP_CARRIER_FL_NUM - CARRIER FLIGHT NUMBER - DROP - IRRELEVANT FOR PROJECT SCOPE ORIGIN_CITY_NAME - DEPARTURE CITY NAME, STATE - DROP - REDUNDANT AIRPORT ID DEST_AIRPORT_ID - ARRIVAL AIRPORT ID - DROP - REDUNDANT AIRPORT ID DEST_CITY_NAME - ARRIVAL CITY NAME, STATE - DROP - REDUNDANT AIRPORT ID CRS_DEP_TIME - SCHEDULED DEPARTURE TIME (local time: hhmm) - DROP - REDUNDANT - DEP_TIME_BLK DEP_TIME - ACTUAL DEPARTURE TIME (local time: hhmm) - DROP REDUNDANT - DEP-DELAY15 DEP_DELAY_NEW - NUMBER OF MINUTES DELAYED (EARLY=0) - DROP REDUNDANT - DEP-DELAY15 CRS_ARR_TIME - SCHEDULED ARRIVAL TIME (local time: hhmm) - DROP REDUNDANT - ARR_TIME_BLK ARR_TIME - ACTUAL ARRIVAL TIME (local time: hhmm) - DROP - IRRELEVANT FOR PROJECT SCOPE ARR_DELAY_NEW - NUMBER OF MINUTES ARRIVAL DELAYED - DROP - IRRELEVANT CANCELLATION_CODE - CANCELLED FLIGHT CODE - DROP - REDUNDANT - CANCELLED ACTUAL_ELAPSED_TIME - ACTUAL ELAPSED TIME - DROP - IRRELEVANT FOR PROJECT SCOPE Unnamed: 32 - BLANK ERROR CELL FROM SOURCE ** DROP ERROR **

[9]:

```
# Dropping Redundant and Irrelevant features (flight specific, redundant␣
 ↪airport ids, all actual departure data, all actual arrival data)
flight_no = ['MONTH', 'OP_CARRIER_FL_NUM', 'ORIGIN_CITY_NAME',␣
 ↪'DEST_AIRPORT_ID', 'DEST_CITY_NAME', 'CRS_DEP_TIME',
           'DEP_TIME', 'DEP_DELAY_NEW', 'CRS_ARR_TIME', 'ARR_TIME',␣
 ↪'ARR_DELAY_NEW', 'CANCELLATION_CODE', 'ACTUAL_ELAPSED_TIME', 'Unnamed: 32']
dec_flight.drop(flight_no, inplace=True, axis=1)
```

[10]:

```
# Save updated flight info to transformed folder in bucket
csv_buffer=StringIO()
dec_flight.to_csv(csv_buffer, index=False)

BUCKET_NAME = 'ads-508-airline'
FileName= 'transformed/ON_TIME_REPORTING_12.csv'

s3csv = boto3.client('s3')
```

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

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx MASSAGE NAMES DATA xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

[11]:
```
names
```

[11]:
```
      AIRLINE_ID OP_UNIQUE_CARRIER          CARRIER_NAME
0          21754               2PQ           21 Air LLC
1          21754               2PQ           21 Air LLC
2          21754               2PQ           21 Air LLC
3          20342                Q5          40-Mile Air
4          20342               WRB          40-Mile Air
…            …                 …                    …
2705       20379               ZKQ  Zantop International
2706       19771               ZAQ  Zas Airline Of Egypt
2707       21118                37             Zeal 320
2708       22069                ZG     ZIPAIR Tokyo Inc.
2709       19894               ZUQ   Zuliana De Aviacion

[2710 rows x 3 columns]
```

[12]:
```
# Drop Duplicates to retain a dictionary
names.drop_duplicates(subset='OP_UNIQUE_CARRIER', inplace=True)
names = names.reset_index(drop=True)
names
```

[12]:
```
      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]
```

[13]:
```
# Save updated carrier info to transformed folder in bucket
csv_buffer=StringIO()
```

```
names.to_csv(csv_buffer, index=False)

BUCKET_NAME = 'ads-508-airline'
FileName= 'transformed/CARRIER_DECODE.csv'

s3csv = boto3.client('s3')

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

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx MASSAGE EMPLOYEES DATA xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

[14]:
```
# Combine Carrier information for different entities and retain entitiy (only␣
 ↪domestic), passenger handling (flight attendant), pass_gen_svc_admin (ground␣
 ↪service), pilot_copilots and maintanence
employees = employees[['OP_UNIQUE_CARRIER', 'ENTITY', 'PILOTS_COPILOTS',␣
 ↪'PASSENGER_HANDLING', 'PASS_GEN_SVC_ADMIN', 'MAINTENANCE']]
# Drop on domestic entities
employees.drop(employees[employees['ENTITY'] != 'D'].index, inplace = True)
# Combine any remaining duplicates
employees = employees.groupby('OP_UNIQUE_CARRIER').sum().reset_index()
# Drop Parcel carriers (airlines with no flight attendants)
employees.drop(employees[employees['PILOTS_COPILOTS'] == 0].index, inplace =␣
 ↪True)
employees = employees.reset_index(drop=True)
```

[15]:
```
# Save updated employee info to transformed folder in bucket
csv_buffer=StringIO()
employees.to_csv(csv_buffer, index=False)

BUCKET_NAME = 'ads-508-airline'
FileName= 'transformed/P10_EMPLOYEES.csv'

s3csv = boto3.client('s3')

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

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx MASSAGE WEATHER DATA xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

[16]:
```
# limit scope of weather metrics (date, precipitation, snow, max temp, and wind)
weather = weather_report[['DATE', 'NAME', 'PRCP', 'SNOW', 'SNWD','TMAX',␣
 ↪'AWND']]
weather
```

[16]:

```
            DATE                                               NAME  PRCP  \
0      12/1/2019  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  0.04
1      12/2/2019  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  0.00
2      12/3/2019  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  0.00
3      12/4/2019  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  0.00
4      12/5/2019  ATLANTA HARTSFIELD JACKSON INTERNATIONAL AIRPO…  0.00
…          …                                                 …    …
3281  12/27/2019                TALLAHASSEE REGIONAL AIRPORT, FL US  0.00
3282  12/28/2019                TALLAHASSEE REGIONAL AIRPORT, FL US  0.06
3283  12/29/2019                TALLAHASSEE REGIONAL AIRPORT, FL US  0.10
3284  12/30/2019                TALLAHASSEE REGIONAL AIRPORT, FL US  0.02
3285  12/31/2019                TALLAHASSEE REGIONAL AIRPORT, FL US  0.00

      SNOW  SNWD  TMAX   AWND
0      0.0   0.0  67.0  16.11
1      0.0   0.0  48.0  16.78
2      0.0   0.0  49.0  11.18
3      0.0   0.0  60.0  11.18
4      0.0   0.0  65.0   5.82
…    …   …    …    …
3281   NaN   NaN  80.0   6.04
3282   NaN   NaN  74.0   5.37
3283   NaN   NaN  74.0   7.61
3284   NaN   NaN  72.0   5.82
3285   NaN   NaN  64.0   3.58

[3286 rows x 7 columns]
```

[17]:

```python
# Save weather info to transformed folder in bucket
csv_buffer=StringIO()
weather.to_csv(csv_buffer, index=False)

BUCKET_NAME = 'ads-508-airline'
FileName= 'transformed/airport_weather_dec_2019.csv'

s3csv = boto3.client('s3')

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

[ ]:

```

```