Sprint 2_Data analysis

June 7, 2022

1 IT Academy - Data Science

1.1 S02 T05: Data analysis

1.1.1 *Exercise 1*

Download the *Airlines Delay: Airline on-time statistics and delay causes* data set and upload it to a Dataframe pandas.

[1]:
```
#import requested library
import pandas as pd
import numpy as np

#import data on airline flights statistics
dataframe = pd.read_csv('DelayedFlights.csv', sep=',', encoding='utf8',␣
 ↪index_col=0, nrows=None)

#data set is very large, show small subset of rows to display (=10)
dataframe.head(10)
```

[1]:

	Year	Month	DayofMonth	DayOfWeek	DepTime	CRSDepTime	ArrTime	\
0	2008	1	3	4	2003.0	1955	2211.0	
1	2008	1	3	4	754.0	735	1002.0	
2	2008	1	3	4	628.0	620	804.0	
4	2008	1	3	4	1829.0	1755	1959.0	
5	2008	1	3	4	1940.0	1915	2121.0	
6	2008	1	3	4	1937.0	1830	2037.0	
10	2008	1	3	4	706.0	700	916.0	
11	2008	1	3	4	1644.0	1510	1845.0	
15	2008	1	3	4	1029.0	1020	1021.0	
16	2008	1	3	4	1452.0	1425	1640.0	

	CRSArrTime	UniqueCarrier	FlightNum	…	TaxiIn	TaxiOut	Cancelled	\
0	2225	WN	335	…	4.0	8.0	0	
1	1000	WN	3231	…	5.0	10.0	0	
2	750	WN	448	…	3.0	17.0	0	
4	1925	WN	3920	…	3.0	10.0	0	
5	2110	WN	378	…	4.0	10.0	0	
6	1940	WN	509	…	3.0	7.0	0	

```
10         915            WN         100  …     5.0      19.0          0
11        1725            WN        1333  …     6.0       8.0          0
15        1010            WN        2272  …     6.0       9.0          0
16        1625            WN         675  …     7.0       8.0          0

    CancellationCode  Diverted  CarrierDelay WeatherDelay NASDelay  \
0                  N         0           NaN          NaN      NaN
1                  N         0           NaN          NaN      NaN
2                  N         0           NaN          NaN      NaN
4                  N         0           2.0          0.0      0.0
5                  N         0           NaN          NaN      NaN
6                  N         0          10.0          0.0      0.0
10                 N         0           NaN          NaN      NaN
11                 N         0           8.0          0.0      0.0
15                 N         0           NaN          NaN      NaN
16                 N         0           3.0          0.0      0.0

    SecurityDelay  LateAircraftDelay
0             NaN                NaN
1             NaN                NaN
2             NaN                NaN
4             0.0               32.0
5             NaN                NaN
6             0.0               47.0
10            NaN                NaN
11            0.0               72.0
15            NaN                NaN
16            0.0               12.0

[10 rows x 29 columns]
```

[2]:
```
#print information about the data
print(dataframe.info())
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1936758 entries, 0 to 7009727
Data columns (total 29 columns):
 #   Column             Dtype
---  ------             -----
 0   Year               int64
 1   Month              int64
 2   DayofMonth         int64
 3   DayOfWeek          int64
 4   DepTime            float64
 5   CRSDepTime         int64
 6   ArrTime            float64
 7   CRSArrTime         int64
```

```
 8   UniqueCarrier      object
 9   FlightNum          int64
 10  TailNum            object
 11  ActualElapsedTime  float64
 12  CRSElapsedTime     float64
 13  AirTime            float64
 14  ArrDelay           float64
 15  DepDelay           float64
 16  Origin             object
 17  Dest               object
 18  Distance           int64
 19  TaxiIn             float64
 20  TaxiOut            float64
 21  Cancelled          int64
 22  CancellationCode   object
 23  Diverted           int64
 24  CarrierDelay       float64
 25  WeatherDelay       float64
 26  NASDelay           float64
 27  SecurityDelay      float64
 28  LateAircraftDelay  float64
dtypes: float64(14), int64(10), object(5)
memory usage: 443.3+ MB
None
```

Clean Data

Explore the data it contains, and keep only the columns that you consider relevant.

[3]:
```
dataframe.drop(['CRSArrTime', 'CRSElapsedTime','FlightNum','TailNum',␣
↪'CRSElapsedTime','TaxiIn', 'TaxiOut',
               'CarrierDelay', 'WeatherDelay', 'NASDelay',␣
↪'SecurityDelay','LateAircraftDelay' ], axis = 1, inplace=True)
```

[4]:
```
#remove all duplicates
dataframe.drop_duplicates(inplace = True)
```

1.1.2 *Exercise 2*

Make a complete report of the date set

[5]:
```
##Summarize the columns of interest statistically
dataframe[['ActualElapsedTime', 'AirTime', 'ArrDelay', 'Distance']].describe()
```

[5]:

	ActualElapsedTime	AirTime	ArrDelay	Distance
count	1.928369e+06	1.928369e+06	1.928369e+06	1.936756e+06
mean	1.333059e+02	1.082772e+02	4.219988e+01	7.656863e+02
std	7.206010e+01	6.864264e+01	5.678474e+01	5.744799e+02
min	1.400000e+01	0.000000e+00	-1.090000e+02	1.100000e+01

```
25%          8.000000e+01  5.800000e+01  9.000000e+00  3.380000e+02
50%          1.160000e+02  9.000000e+01  2.400000e+01  6.060000e+02
75%          1.650000e+02  1.370000e+02  5.600000e+01  9.980000e+02
max          1.114000e+03  1.091000e+03  2.461000e+03  4.962000e+03
```

[6]:
```
##Find missing data per column
dataframe.isnull().sum()
```

[6]:
```
Year                    0
Month                   0
DayofMonth              0
DayOfWeek               0
DepTime                 0
CRSDepTime              0
ArrTime              7110
UniqueCarrier           0
ActualElapsedTime    8387
AirTime              8387
ArrDelay             8387
DepDelay                0
Origin                  0
Dest                    0
Distance                0
Cancelled               0
CancellationCode        0
Diverted                0
dtype: int64
```

[7]:
```
#return a new Data Frame with no empty cells
df = dataframe.dropna()
```

[8]:
```
#check
df.isnull().sum()
```

[8]:
```
Year                 0
Month                0
DayofMonth           0
DayOfWeek            0
DepTime              0
CRSDepTime           0
ArrTime              0
UniqueCarrier        0
ActualElapsedTime    0
AirTime              0
ArrDelay             0
DepDelay             0
Origin               0
```

```
Dest                0
Distance            0
Cancelled           0
CancellationCode    0
Diverted            0
dtype: int64
```

[9]:
```
##Create new columns

## Show the average flight speed
# First convert the distance from miles to km
def distance_km(m):
    km = m * 1.609344
    return round(km, 2)

#get average distance in km
dist = df["Distance"].mean()
print("Average distance in km:",distance_km(dist))
```

```
Average distance in km: 1231.07
```

[10]:
```
#get average speed
df["DistanceKm"] = df["Distance"].apply(distance_km)
df["AvgSpeed"] = df["Distance"] / (df["AirTime"] / 60)

#returns a specified number of random rows to show result
df.sample(10)
```

```
/var/folders/hd/v__zth8s6xz0y6mlb6y3m2j00000gn/T/ipykernel_23252/754632412.py:2:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df["DistanceKm"] = df["Distance"].apply(distance_km)
/var/folders/hd/v__zth8s6xz0y6mlb6y3m2j00000gn/T/ipykernel_23252/754632412.py:3:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df["AvgSpeed"] = df["Distance"] / (df["AirTime"] / 60)
```

[10]:

	Year	Month	DayofMonth	DayOfWeek	DepTime	CRSDepTime	ArrTime	\
2729377	2008	5	21	3	758.0	600	1010.0	

```
6638640  2008     12          18          4   2205.0        2051   2251.0
5149228  2008      9          25          4   1100.0        1020   1310.0
2239680  2008      4          20          7   1609.0        1600   1723.0
2771790  2008      5          24          6    937.0         900   1100.0
1998814  2008      4          15          2   1516.0        1430   1600.0
3310229  2008      6          27          5    916.0         705   1051.0
6586158  2008     12          10          3    730.0         717    825.0
1625     2008      1           3          4   1938.0        1910   2049.0
5337801  2008      9          14          7   1926.0        1850   2219.0

        UniqueCarrier  ActualElapsedTime  AirTime  ArrDelay  DepDelay Origin  \
2729377            EV               72.0     53.0     113.0     118.0    JAN
6638640            OO               46.0     34.0      74.0      74.0    PDX
5149228            EV               70.0     52.0      37.0      40.0    MOB
2239680            9E              134.0    110.0      -2.0       9.0    ATL
2771790            MQ               83.0     64.0      32.0      37.0    DFW
1998814            OO               44.0     30.0      40.0      46.0    CWA
3310229            DL               95.0     74.0     114.0     131.0    IAD
6586158            YV              115.0     95.0      -2.0      13.0    CLT
1625               WN               71.0     53.0      29.0      28.0    RNO
5337801            B6              173.0    128.0      42.0      36.0    TPA

        Dest  Distance  Cancelled CancellationCode  Diverted  DistanceKm  \
2729377  ATL       341          0                N         0      548.79
6638640  RDM       116          0                N         0      186.68
5149228  ATL       302          0                N         0      486.02
2239680  HOU       696          0                N         0     1120.10
2771790  AEX       285          0                N         0      458.66
1998814  MKE       154          0                N         0      247.84
3310229  ATL       533          0                N         0      857.78
6586158  ORD       599          0                N         0      964.00
1625     LAS       345          0                N         0      555.22
5337801  JFK      1005          0                N         0     1617.39

           AvgSpeed
2729377  386.037736
6638640  204.705882
5149228  348.461538
2239680  379.636364
2771790  267.187500
1998814  308.000000
3310229  432.162162
6586158  378.315789
1625     390.566038
5337801  471.093750
```

[11]:
```
# consider var y as minimum minutes to be considered a delayed flight
y = 15
# create a boolean var
df["Delay"] = df["ArrDelay"].apply(lambda x: True if x>y else False)

#returns a specified number of random rows to show result
df.sample(10)
```

```
/var/folders/hd/v__zth8s6xz0y6mlb6y3m2j00000gn/T/ipykernel_23252/1034309438.py:4
: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df["Delay"] = df["ArrDelay"].apply(lambda x: True if x>y else False)
```

[11]:

	Year	Month	DayofMonth	DayOfWeek	DepTime	CRSDepTime	ArrTime	\
6971463	2008	12	31	3	718.0	700	1153.0	
6543778	2008	12	28	7	1945.0	1855	2043.0	
2766045	2008	5	17	6	1852.0	1835	1938.0	
6174105	2008	11	12	3	937.0	930	1150.0	
3324079	2008	6	27	5	1147.0	1110	1409.0	
2741423	2008	5	3	6	1507.0	1440	1830.0	
4624625	2008	8	20	3	1225.0	1200	1328.0	
3727522	2008	7	24	4	944.0	905	1106.0	
6535479	2008	12	25	4	2124.0	1905	2255.0	
1582104	2008	3	27	4	1421.0	1330	1615.0	

	UniqueCarrier	ActualElapsedTime	AirTime	…	DepDelay	Origin	Dest	\
6971463	CO	215.0	190.0	…	18.0	EWR	SJU	
6543778	WN	58.0	48.0	…	50.0	BWI	PVD	
2766045	HA	46.0	30.0	…	17.0	KOA	HNL	
6174105	US	253.0	233.0	…	7.0	CLT	PHX	
3324079	EV	142.0	118.0	…	37.0	ATL	SWF	
2741423	FL	203.0	182.0	…	27.0	BOS	RSW	
4624625	MQ	63.0	49.0	…	25.0	DFW	TXK	
3727522	XE	82.0	63.0	…	39.0	JAN	IAH	
6535479	WN	91.0	70.0	…	139.0	LAS	OAK	
1582104	MQ	114.0	87.0	…	51.0	LIT	ORD	

	Distance	Cancelled	CancellationCode	Diverted	DistanceKm	\
6971463	1608	0	N	0	2587.83	
6543778	328	0	N	0	527.86	
2766045	163	0	N	0	262.32	
6174105	1774	0	N	0	2854.98	
3324079	784	0	N	0	1261.73	

```
2741423     1249          0                N        0     2010.07
4624625      181          0                N        0      291.29
3727522      351          0                N        0      564.88
6535479      407          0                N        0      655.00
1582104      552          0                N        0      888.36

           AvgSpeed  Delay
6971463  507.789474  False
6543778  410.000000   True
2766045  326.000000   True
6174105  456.824034  False
3324079  398.644068   True
2741423  411.758242   True
4624625  221.632653   True
3727522  334.285714   True
6535479  348.857143   True
1582104  380.689655   True

[10 rows x 21 columns]
```

[14]:
```
##Table of airlines with the most accumulated delays
#import the carrier file as a series
carriers = pd.read_csv('carriers.csv', sep=',', encoding='utf8', index_col=0,␣
 ↪squeeze=True)
carriers.sample(10)
```

```
/var/folders/hd/v__zth8s6xz0y6mlb6y3m2j00000gn/T/ipykernel_23252/2581724392.py:3
: FutureWarning: The squeeze argument has been deprecated and will be removed in
a future version. Append .squeeze("columns") to the call to squeeze.


  carriers = pd.read_csv('carriers.csv', sep=',', encoding='utf8', index_col=0,
squeeze=True)
```

[14]:
```
Code
TPQ        Aerial Transit Company
MAX                   Maxair Inc.
WI              Tradewinds Airlines
AJ              Air Micronesia Inc.
RAQ    Arista Int'l Airlines Inc.
CC           Air Atlanta Icelandic
MET             Metroplex Airlines
VIQ           Volga-Dnepr Airlines
17               Piedmont Airlines
CAV                   Air Virginia
Name: Description, dtype: object
```

[15]:
```
#create new column using map() function
df["Carrier"] = df["UniqueCarrier"].map(carriers)
df.sample(10)
```

```
/var/folders/hd/v__zth8s6xz0y6mlb6y3m2j00000gn/T/ipykernel_23252/3919281804.py:2
: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df["Carrier"] = df["UniqueCarrier"].map(carriers)
```

[15]:

	Year	Month	DayofMonth	DayOfWeek	DepTime	CRSDepTime	ArrTime	\
3242363	2008	6	12	4	1053.0	1010	1919.0	
4494295	2008	8	1	5	1654.0	1645	2252.0	
934870	2008	2	14	4	1118.0	1056	1456.0	
1968813	2008	4	21	1	743.0	725	1223.0	
1251184	2008	3	25	2	1419.0	1400	1543.0	
2739022	2008	5	29	4	1023.0	1010	1454.0	
6133914	2008	11	21	5	1812.0	1758	2142.0	
3398325	2008	6	5	4	2040.0	1955	2212.0	
356978	2008	1	5	6	1156.0	1149	1253.0	
5326656	2008	9	19	5	1329.0	1310	1406.0	

	UniqueCarrier	ActualElapsedTime	AirTime	…	Origin	Dest	Distance	\
3242363	UA	326.0	307.0	…	SFO	JFK	2586	
4494295	US	298.0	277.0	…	ANC	LAS	2304	
934870	F9	158.0	135.0	…	SEA	DEN	1024	
1968813	OO	160.0	140.0	…	LAX	DFW	1235	
1251184	WN	84.0	71.0	…	SMF	PDX	479	
2739022	F9	151.0	135.0	…	DEN	DTW	1123	
6133914	UA	390.0	353.0	…	BOS	SFO	2704	
3398325	MQ	32.0	22.0	…	GRB	MQT	134	
356978	FL	57.0	41.0	…	ATL	CLT	227	
5326656	AS	37.0	28.0	…	CDV	ANC	160	

	Cancelled	CancellationCode	Diverted	DistanceKm	AvgSpeed	Delay	\
3242363	0	N	0	4161.76	505.407166	False	
4494295	0	N	0	3707.93	499.061372	False	
934870	0	N	0	1647.97	455.111111	True	
1968813	0	N	0	1987.54	529.285714	False	
1251184	0	N	0	770.88	404.788732	True	
2739022	0	N	0	1807.29	499.111111	False	
6133914	0	N	0	4351.67	459.603399	False	
3398325	0	N	0	215.65	365.454545	True	
356978	0	N	0	365.32	332.195122	False	

```
5326656         0                 N          0     257.50  342.857143  False

                                                    Carrier
3242363                                 United Air Lines Inc.
4494295  US Airways Inc. (Merged with America West 9/05…
934870                                Frontier Airlines Inc.
1968813                                Skywest Airlines Inc.
1251184                               Southwest Airlines Co.
2739022                               Frontier Airlines Inc.
6133914                                United Air Lines Inc.
3398325                          American Eagle Airlines Inc.
356978                            AirTran Airways Corporation
5326656                                  Alaska Airlines Inc.

[10 rows x 22 columns]
```

[22]:
```
#group flight by carrier company
CarrierFlights = df.groupby('Carrier')

#show
CarrierFlights["Delay"].sum()
CarrierFlights["Delay"].agg([np.sum, np.size])
```

[22]:

Carrier	sum	size
AirTran Airways Corporation	45738	70969
Alaska Airlines Inc.	23340	39010
Aloha Airlines Inc.	321	744
American Airlines Inc.	129401	190910
American Eagle Airlines Inc.	95126	141223
Atlantic Southeast Airlines	55483	81762
Comair Inc.	38362	52453
Continental Air Lines Inc.	57497	99731
Delta Air Lines Inc.	70260	113728
Expressjet Airlines Inc.	70455	103147
Frontier Airlines Inc.	15267	28222
Hawaiian Airlines Inc.	4124	7472
JetBlue Airways	37538	54925
Mesa Airlines Inc.	49536	66769
Northwest Airlines Inc.	53392	78843
Pinnacle Airlines Inc.	34836	51569
Skywest Airlines Inc.	86619	131780
Southwest Airlines Co.	196424	376201
US Airways Inc. (Merged with America West 9/05…	57936	98007
United Air Lines Inc.	93355	140904

1.1.3 *Exercise 3*

Export the clean date set with the new columns to Excel

[24]:
```
#error message:This sheet is too large!
#---> df.to_excel('CleanDelayedFlights.xlsx')

df.to_csv('CleanDelayedFlights.csv')
```