1 Exercici

Descarrega el data set Airlines Delay: Airline on-time statistics and delay causes (https://www.kaggle.com/giovamata/airlinedelaycauses) i carrega'l a un pandas Dataframe. Explora les dades que conté, i queda't únicament amb les columnes que consideris rellevants.

In [1]:
```
import os, datetime
import pandas as pd
import numpy as np

import warnings
warnings.simplefilter(action='ignore', category=FutureWarning)
```

executed in 410ms, finished 10:12:15 2021-04-14

In [2]:
```
rows = None
pd.options.display.max_columns = None

datasets_path = r"D:\Oscar\FORMACIO\DIGITAL\DATA SCIENCE with Python\Datasets"
datasets_path = os.getcwd()
datasets_path += os.sep

file = datasets_path + "DelayedFlights.csv"
df = pd.read_csv(file, sep=',', encoding='utf8', index_col=0, nrows=rows)
df.head(3)
```

executed in 9.01s, finished 10:06:53 2021-04-14

Out[2]:

	Year	Month	DayofMonth	DayOfWeek	DepTime	CRSDepTime	ArrTime	CRSArrTime	UniqueCarrier	FlightNum	TailNum	ActualElapsedTime	CRSElapsedTime	AirTime	ArrDelay
0	2008	1	3	4	2003.0	1955	2211.0	2225	WN	335	N712SW	128.0	150.0	116.0	-1
1	2008	1	3	4	754.0	735	1002.0	1000	WN	3231	N772SW	128.0	145.0	113.0	
2	2008	1	3	4	628.0	620	804.0	750	WN	448	N428WN	96.0	90.0	76.0	1

Documentació de la descripció de les columnes a Get the data (http://stat-computing.org/dataexpo/2009/the-data.html)

Variable descriptions

	Name	Description
1	Year	1987-2008
2	Month	1-12
3	DayofMonth	1-31
4	DayOfWeek	1 (Monday) - 7 (Sunday)
5	DepTime	actual departure time (local, hhmm)
6	CRSDepTime	scheduled departure time (local, hhmm)
7	ArrTime	actual arrival time (local, hhmm)
8	CRSArrTime	scheduled arrival time (local, hhmm)
9	UniqueCarrier	unique carrier code
10	FlightNum	flight number
11	TailNum	plane tail number
12	ActualElapsedTime	in minutes
13	CRSElapsedTime	in minutes
14	AirTime	in minutes
15	ArrDelay	arrival delay, in minutes
16	DepDelay	departure delay, in minutes
17	Origin	origin IATA airport code
18	Dest	destination IATA airport code
19	Distance	in miles
20	TaxiIn	taxi in time, in minutes
21	TaxiOut	taxi out time in minutes
22	Cancelled	was the flight cancelled?
23	CancellationCode	reason for cancellation (A = carrier, B = weather, C = NAS, D = security)
24	Diverted	1 = yes, 0 = no
25	CarrierDelay	in minutes
26	WeatherDelay	in minutes
27	NASDelay	in minutes
28	SecurityDelay	in minutes
29	LateAircraftDelay	in minutes

In [3]:
```
#esborrem les columnes que no considerem d'interès
df.drop(['FlightNum', 'TailNum', 'ArrTime', 'CRSArrTime', 'CRSElapsedTime', 'DepDelay',
         'TaxiIn', 'TaxiOut', 'DepTime', 'CRSDepTime', 'CarrierDelay',
         'WeatherDelay', 'NASDelay', 'SecurityDelay', 'LateAircraftDelay' ],
        axis = 'columns', inplace=True)

#canviem els noms de les cols que hem conservat
old_col_names = list(df.columns)
new_col_names = ["ANY", "MES", "DIA", "DIA_SET", "COD_AEROLINIA", "DURADA_TOTAL",
                 "DURADA_AIRE", "RETARD", "COD_ORIGEN", "COD_DESTI", "DISTANCIA",
                 "CANCELAT", "COD_CANCELACIO", "DESVIAT" ]

replace = dict(zip(old_col_names, new_col_names))
df.rename(columns=replace, inplace=True)
df.sample(3)
```

executed in 350ms, finished 10:06:57 2021-04-14

Out[3]:

	ANY	MES	DIA	DIA_SET	COD_AEROLINIA	DURADA_TOTAL	DURADA_AIRE	RETARD	COD_ORIGEN	COD_DESTI	DISTANCIA	CANCELAT	COD_CANCELACIO	DESVIAT
3981018	2008	7	15	2	FL	177.0	160.0	11.0	SJU	MCO	1189	0	N	0
2323551	2008	4	26	6	AS	188.0	164.0	-7.0	SEA	ANC	1449	0	N	0
3517912	2008	6	13	5	AA	51.0	39.0	-3.0	MIA	TPA	204	0	N	0

In [4]:
```
#convertirem int64 i float64 a int32 i float32 per reduir el dataframe
#definim una funcio que genera el diccionari de conversió de tipus
def dic_convert(colsint, colsfloat):
    dtype_l = []
    for i in range(0, len(colsint)): dtype_l.append("int32")
    for i in range(0, len(colsfloat)): dtype_l.append("float32")
    return dict(zip(colsint+colsfloat, dtype_l))

colsint = ["ANY", "MES", "DIA", "DIA_SET", "DISTANCIA",
           "CANCELAT", "DESVIAT"]
colsfloat = ["DURADA_TOTAL", "DURADA_AIRE", "RETARD"]
dic = dic_convert(colsint, colsfloat)
df = df.astype(dic)

df.info(memory_usage="deep")
```

executed in 1.68s, finished 10:06:59 2021-04-14

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1936758 entries, 0 to 7009727
Data columns (total 14 columns):
 #   Column          Dtype
---  ------          -----
 0   ANY             int32
 1   MES             int32
 2   DIA             int32
 3   DIA_SET         int32
 4   COD_AEROLINIA   object
 5   DURADA_TOTAL    float32
 6   DURADA_AIRE     float32
 7   RETARD          float32
 8   COD_ORIGEN      object
 9   COD_DESTI       object
 10  DISTANCIA       int32
 11  CANCELAT        int32
 12  COD_CANCELACIO  object
 13  DESVIAT         int32
dtypes: float32(3), int32(7), object(4)
memory usage: 526.4 MB
```

2 Exercici

Fes un informe complet del data set:.

2.1 Resumeix estadísticament les columnes d'interès

In [5]:
```
df[["DURADA_TOTAL", "DURADA_AIRE", "DISTANCIA"]].describe()
```

executed in 341ms, finished 10:07:01 2021-04-14

Out[5]:

	DURADA_TOTAL	DURADA_AIRE	DISTANCIA
count	1.928371e+06	1.928371e+06	1.936758e+06
mean	1.333069e+02	1.082788e+02	7.656862e+02
std	7.200964e+01	6.860229e+01	5.744797e+02
min	1.400000e+01	0.000000e+00	1.100000e+01
25%	8.000000e+01	5.800000e+01	3.380000e+02
50%	1.160000e+02	9.000000e+01	6.060000e+02
75%	1.650000e+02	1.370000e+02	9.980000e+02
max	1.114000e+03	1.091000e+03	4.962000e+03

In [6]:
```
#podem veure, en concret, el promig de duració en l'aire d'un vol
#definim funció per passar de minuts a str per mostrar dades temporals
def min_to_strtime(m):
    m_td = datetime.timedelta(minutes=m)
    hores, reste = divmod(m_td.seconds, 3600)
    minuts, segons = divmod(reste, 60)
    return ":".join([str(hores)+"h", str(minuts)+"m", str(segons)+"s"])

minuts = df["DURADA_AIRE"].mean()
min_to_strtime(minuts)
```

executed in 24ms, finished 10:07:02 2021-04-14

Out[6]: '1h:48m:16s'

In [7]:
```
#promig de distància per vol
#definim funció per passar de milles a km
def miles_to_km(m):
    km = m * 1.60934
    return round(km, 2)

dist = df["DISTANCIA"].mean()
str(miles_to_km(dist)) + "km"
```

executed in 17ms, finished 10:07:03 2021-04-14

Out[7]: '1232.25km'

In [8]:
```
#per exemple podem veure els vols per dia de la setmana
#abans podem mapejar la columna DIA_SET per ferla més llegible
dies_dict = {1:"diumenge", 2:"dilluns", 3:"dimarts", 4:"dimecres", 5:"dijous", 6:"divendres", 7:"dissabte"}
df["DIA_SET"] = df["DIA_SET"].map(dies_dict)

vols_dia = df.DIA_SET.value_counts()
vols_dia
```

executed in 448ms, finished 10:07:05 2021-04-14

Out[8]:
```
dijous       323259
diumenge     290933
dimecres     289451
dissabte     286111
dimarts      262805
dilluns      260943
divendres    223256
Name: DIA_SET, dtype: int64
```

In [9]:
```
#en percentatges els vols per dia de la setmana:
totals = len(df.DIA_SET)
vols_dia.apply(lambda x : str(round((x*100/totals), 2)) + "%")
```

executed in 11ms, finished 10:07:05 2021-04-14

Out[9]:
```
dijous       16.69%
diumenge     15.02%
dimecres     14.95%
dissabte     14.77%
dimarts      13.57%
dilluns      13.47%
divendres    11.53%
Name: DIA_SET, dtype: object
```

In [10]:
```
#analitzarem els vols cancel·lats i desviats
#convertim CANCELAT i DESVIAT a boolean per visualitzar
df = df.astype({"CANCELAT": bool, "DESVIAT": bool})

#definim funcio per retornar porcentatges de Series de tipus boolean
def perc_trues(series):
    try: trues = series.value_counts()[True]
    except: trues = 0
    totals = len(series)
    return str(round((trues * 100 / totals), 2)) + "%"
```

executed in 271ms, finished 10:07:08 2021-04-14

In [11]:
```
#observem el nombre de vols cancel·lats
df.CANCELAT.value_counts()
```

executed in 52ms, finished 10:07:09 2021-04-14

Out[11]:
```
False    1936125
True         633
Name: CANCELAT, dtype: int64
```

In [12]:
```
#percentatge de vols totals cancel·lats
perc_trues(df["CANCELAT"])
```

executed in 50ms, finished 10:07:10 2021-04-14

Out[12]:
```
'0.03%'
```

In [13]:
```
#calcularem dels cancel·lats els percentatges dels motius
df["COD_CANCELACIO"].value_counts()
```

executed in 281ms, finished 10:07:11 2021-04-14

Out[13]:
```
N    1936125
B        307
A        246
C         80
Name: COD_CANCELACIO, dtype: int64
```

In [14]:
```
#calculem percentatges
#mapejem els motius per fer-los entendibles:
motiu_cancel = {"A": "Aerolínia", "B": "Pel temps", "C": "Problema tècnic", "D": "Per seguretat", "N": "No cancelat"}
df["COD_CANCELACIO"] = df["COD_CANCELACIO"].map(motiu_cancel)

#generem un nou dataframe de cancel·lats:
cancel_df = df[df["CANCELAT"]]

motius_s = cancel_df["COD_CANCELACIO"].value_counts()
num_cancels = cancel_df["COD_CANCELACIO"].size

round(100 * motius_s / num_cancels, 2).astype(str) + "%"
```

executed in 1.35s, finished 10:07:13 2021-04-14

Out[14]:
```
Pel temps           48.5%
Aerolínia          38.86%
Problema tècnic    12.64%
Name: COD_CANCELACIO, dtype: object
```

In [15]:
```
#vols desviats
df["DESVIAT"].value_counts()
```

executed in 52ms, finished 10:07:14 2021-04-14

Out[15]:
```
False    1929004
True        7754
Name: DESVIAT, dtype: int64
```

In [16]:
```
#percentatge vols desviats
perc_trues(df["DESVIAT"])
```

executed in 60ms, finished 10:07:15 2021-04-14

Out[16]:
```
'0.4%'
```

In [17]:
```
#podem veure nombre de vols programats per aerolínia (s'inclouen els cancel·lats)
df.COD_AEROLINIA.value_counts()
```

executed in 313ms, finished 10:07:16 2021-04-14

Out[17]:
```
WN    377602
AA    191865
MQ    141920
UA    141426
OO    132433
DL    114238
XE    103663
CO    100195
US     98425
EV     81877
NW     79108
FL     71284
YV     67063
B6     55315
OH     52657
9E     51885
AS     39293
F9     28269
HA      7490
AQ       750
Name: COD_AEROLINIA, dtype: int64
```

In [18]:
```
#per exemple podriem visualitzar, per aerolínia, els km totals i km promig de distàncies
vols_x_co = df.groupby('COD_AEROLINIA')

#utilitzem la funció generada anteriorment per visualitzar
vols_x_co["DISTANCIA"].agg([np.sum, np.mean]).applymap(miles_to_km)
```

executed in 634ms, finished 10:07:18 2021-04-14

Out[18]:

COD_AEROLINIA	sum	mean
WN	3.954916e+08	1047.38
AA	3.340938e+08	1741.30
UA	2.457809e+08	1737.88
CO	1.999466e+08	1995.57
DL	1.780624e+08	1558.70
US	1.532836e+08	1557.36
NW	1.049296e+08	1326.41
XE	9.993560e+07	964.04
MQ	9.900768e+07	697.63
B6	9.862930e+07	1783.05
OO	9.389067e+07	708.97
FL	8.398570e+07	1178.18
EV	6.081013e+07	742.70
AS	5.973907e+07	1520.35
YV	4.396488e+07	655.58
OH	4.349692e+07	826.04
F9	4.081024e+07	1443.64
9E	3.878161e+07	747.45
HA	1.085447e+07	1449.19
AQ	1.074078e+06	1432.10

In [19]:
```
#o el nombre i percentatge de cancel·lats per aerolínia
def percentatge(series):
    return str(round(series.sum() * 100 / series.count(), 2)) + "%"

vols_x_co["CANCELAT"].agg([np.sum, np.size, percentatge])
```

executed in 349ms, finished 10:07:19 2021-04-14

Out[19]:

COD_AEROLINIA	sum	size	percentatge
AQ	0	750	0.0%
WN	15	377602	0.0%
F9	2	28269	0.01%
FL	7	71284	0.01%
AA	46	191865	0.02%
B6	10	55315	0.02%
DL	21	114238	0.02%
NW	16	79108	0.02%
OH	12	52657	0.02%
AS	11	39293	0.03%
UA	47	141426	0.03%
US	26	98425	0.03%
CO	38	100195	0.04%
EV	29	81877	0.04%
HA	3	7490	0.04%
XE	46	103663	0.04%
MQ	104	141920	0.07%
OO	89	132433	0.07%
YV	53	67063	0.08%
9E	58	51885	0.11%

2.2 Troba quantes dades faltants hi ha per columna

In [20]:
```
#busquem valors nulls
df.isnull().sum()
```

executed in 713ms, finished 10:07:27 2021-04-14

Out[20]:

```
ANY                   0
MES                   0
DIA                   0
DIA_SET               0
COD_AEROLINIA         0
DURADA_TOTAL       8387
DURADA_AIRE        8387
RETARD             8387
COD_ORIGEN            0
COD_DESTI             0
DISTANCIA             0
CANCELAT              0
COD_CANCELACIO        0
DESVIAT               0
dtype: int64
```

In [21]:
```
#eliminem els vols cancelats doncs ja no els necessitarem per evaluar les estadistiques dels vols realitzats
df.drop(df[df["CANCELAT"]].index, axis="index", inplace = True)

#eliminem la columna CANCELAT i la de codis doncs no nhi haurà cap
df.drop(['CANCELAT'], axis = 'columns', inplace=True)
df.drop(['COD_CANCELACIO'], axis = 'columns', inplace=True)
```

executed in 763ms, finished 10:07:31 2021-04-14

In [22]:
```
#mirem quants vols amb nulls ens queden
df.isnull().any(axis="columns").value_counts()
```

executed in 594ms, finished 10:07:32 2021-04-14

Out[22]:
```
False    1928371
True        7754
dtype: int64
```

In [23]:
```
#Confirmem que hi ha el mateix nombre de desviats que de vols amb nulls
df[df["DESVIAT"] == True]["DESVIAT"].value_counts()
```

executed in 37ms, finished 10:07:33 2021-04-14

Out[23]:
```
True    7754
Name: DESVIAT, dtype: int64
```

In [24]:
```
#Concluim que tots els nulls son desviats que no tenen dades del vol
df[df["DESVIAT"] == True].sample()
```

executed in 45ms, finished 10:07:34 2021-04-14

Out[24]:

	ANY	MES	DIA	DIA_SET	COD_AEROLINIA	DURADA_TOTAL	DURADA_AIRE	RETARD	COD_ORIGEN	COD_DESTI	DISTANCIA	DESVIAT
6029223	2008	11	30	dissabte	WN	NaN	NaN	NaN	OAK	SAN	446	True

In [25]:
```
#eliminarem els desviats que no ens aporten info, i no ens haurien de quedar nulls al dataframe
df.drop(df[df["DESVIAT"]].index, axis="index", inplace = True)

#eliminem la columna doncs no nhi haurà cap
df.drop(['DESVIAT'], axis = 'columns', inplace=True)

#comprovem que no queden valors nulls
df.isnull().any(axis="columns").value_counts()
```

executed in 1.39s, finished 10:07:37 2021-04-14

Out[25]:
```
False    1928371
dtype: int64
```

2.3 Crea columnes noves (velocitat mitjana del vol, si ha arribat tard o no...)

In [26]:
```
# per crear la velocitat mitjana primer convertirem la distància de milles a km al df
#utilitzarem la funció miles_to_km(m), generada anteriorment

df["DISTANCIA"] = df["DISTANCIA"].apply(miles_to_km)
df["VEL_MITJA"] = df["DISTANCIA"] / (df["DURADA_AIRE"] / 60)

#generem la columna booleana si ha arribat tard
#en una variable r farem la consideració dels minuts a partir dels quals decidim que un vol arriba tard.
#posarem, p.ex. 10 minuts
r = 10
df["RETARDAT"] = df["RETARD"].apply(lambda x: True if x>r else False)

df.sample(3)
```

executed in 4.74s, finished 10:07:43 2021-04-14

Out[26]:

	ANY	MES	DIA	DIA_SET	COD_AEROLINIA	DURADA_TOTAL	DURADA_AIRE	RETARD	COD_ORIGEN	COD_DESTI	DISTANCIA	VEL_MITJA	RETARDAT
247692	2008	1	22	dilluns	UA	94.0	70.0	162.0	BUR	SFO	524.64	449.691444	True
5693917	2008	10	27	diumenge	EV	105.0	85.0	19.0	ATL	MOB	486.02	343.072951	True
2218183	2008	4	12	divendres	NW	194.0	173.0	18.0	MSP	MCO	2108.24	731.181475	True

In [27]:
```
#crearem una columna nova, per expemple, per agrupar les tres columnes de la data en un camp amb el format ddmmaaaa
any_s = df["ANY"].astype(str)
mes_s = df["MES"].astype(str).str.zfill(2)
dia_s = df["DIA"].astype(str).str.zfill(2)

#esborrem les columnes DIA, MES i ANY
df.drop(columns=["ANY", "MES", "DIA", "DIA_SET"], inplace=True)

df["DATA"] = dia_s + mes_s + any_s

df.sample(3)
```

executed in 10.7s, finished 10:07:55 2021-04-14

Out[27]:

	COD_AEROLINIA	DURADA_TOTAL	DURADA_AIRE	RETARD	COD_ORIGEN	COD_DESTI	DISTANCIA	VEL_MITJA	RETARDAT	DATA
801060	OO	55.0	27.0	29.0	ASE	DEN	201.17	447.044456	True	23022008
243335	UA	123.0	88.0	124.0	ORD	PHL	1091.13	743.952257	True	10012008
1900093	XE	101.0	73.0	23.0	CLE	BDL	764.44	628.306833	True	04042008

2.4 Taula de les aerolínies amb més endarreriments acumulats

Crearem una nova columna amb el nom de l'aerolínia que mapejarem de la base de dades carriers.csv que ens donen a [Supplemental data (http://stat-computing.org/dataexpo/2009/supplemental-data.html)]

In [28]:
```
#importem l'arxiu d'aerolínies com una series, amb els codis d'index
file = datasets_path + "DelayedFlights-carriers.csv"
ap_s = pd.read_csv(file, sep=',', encoding='utf8', index_col=0, squeeze=True)
ap_s.sample(3)
```

executed in 46ms, finished 10:07:55 2021-04-14

Out[28]:
```
Code
SJA       San Juan Airlines Inc.
WL             World Air Network
GD     Transp. Aereos Ejecutivos
Name: Description, dtype: object
```

In [29]:
```
#creem la nova columna mapejant la series
df["AEROLINIA"] = df["COD_AEROLINIA"].map(ap_s)

#eliminem la columna de codi de l'aerolinia
df.drop(columns=["COD_AEROLINIA"], inplace=True)

df.sample()
```

executed in 870ms, finished 10:07:59 2021-04-14

Out[29]:

	DURADA_TOTAL	DURADA_AIRE	RETARD	COD_ORIGEN	COD_DESTI	DISTANCIA	VEL_MITJA	RETARDAT	DATA	AEROLINIA
622702	71.0	60.0	18.0	SJC	LAS	621.21	621.21	True	07022008	Southwest Airlines Co.

In [30]:
```
#agrupem per aerolínia
vols_x_co = df.groupby('AEROLINIA')

#obtenim el sumatori dels Trues o el nombre de vols retardats
#(recordem que haviem considerat que era retard si arribava a partir dels 10 min. de l'hora programada)
#(en un futur podriem reconsiderar aquest valor)
vols_x_co["RETARDAT"].sum()
```

executed in 415ms, finished 10:08:00 2021-04-14

Out[30]:

```
AEROLINIA
AirTran Airways Corporation                                          52463
Alaska Airlines Inc.                                                 27037
Aloha Airlines Inc.                                                    443
American Airlines Inc.                                              144326
American Eagle Airlines Inc.                                        107440
Atlantic Southeast Airlines                                          62458
Comair Inc.                                                          42453
Continental Air Lines Inc.                                           65637
Delta Air Lines Inc.                                                 80964
Expressjet Airlines Inc.                                             78577
Frontier Airlines Inc.                                               18882
Hawaiian Airlines Inc.                                                5293
JetBlue Airways                                                      41097
Mesa Airlines Inc.                                                   55146
Northwest Airlines Inc.                                              60499
Pinnacle Airlines Inc.                                               38939
Skywest Airlines Inc.                                                98693
Southwest Airlines Co.                                              235202
US Airways Inc. (Merged with America West 9/05. Reporting for both starting 10/07.)     66264
United Air Lines Inc.                                               103644
Name: RETARDAT, dtype: int64
```

In [31]:
```
#fem una agregacio per obtindre nombre de retardats, el total de vols per aerolínia i
#en percentatges els vols retardats per aerolínia
#(utilitzem la funcio percentatge definida anteriorment)

vols_x_co["RETARDAT"].agg([np.sum, np.size, percentatge])
```

executed in 566ms, finished 10:08:02 2021-04-14

Out[31]:

AEROLINIA	sum	size	percentatge
Aloha Airlines Inc.	443	744	59.54%
Southwest Airlines Co.	235202	376201	62.52%
Continental Air Lines Inc.	65637	99731	65.81%
Frontier Airlines Inc.	18882	28224	66.9%
US Airways Inc. (Merged with America West 9/05. Reporting for both starting 10/07.)	66264	98007	67.61%
Alaska Airlines Inc.	27037	39010	69.31%
Hawaiian Airlines Inc.	5293	7472	70.84%
Delta Air Lines Inc.	80964	113728	71.19%
United Air Lines Inc.	103644	140904	73.56%
AirTran Airways Corporation	52463	70969	73.92%
JetBlue Airways	41097	54925	74.82%
Skywest Airlines Inc.	98693	131780	74.89%
Pinnacle Airlines Inc.	38939	51569	75.51%
American Airlines Inc.	144326	190910	75.6%
American Eagle Airlines Inc.	107440	141223	76.08%
Expressjet Airlines Inc.	78577	103147	76.18%
Atlantic Southeast Airlines	62458	81762	76.39%
Northwest Airlines Inc.	60499	78843	76.73%
Comair Inc.	42453	52453	80.94%
Mesa Airlines Inc.	55146	66769	82.59%

2.5 Quins són els vols més llargs? I els més endarrerits?

In [32]:
```
#obtindrem, p.ex., en una variable n que podem canviar, els 5 vols més llargs en quant a durada
#si es repeteixen les durades els mostrem tots
n = 5
df.nlargest(n, "DURADA_TOTAL", keep='all')
```

executed in 257ms, finished 10:08:04 2021-04-14

Out[32]:

	DURADA_TOTAL	DURADA_AIRE	RETARD	COD_ORIGEN	COD_DESTI	DISTANCIA	VEL_MITJA	RETARDAT	DATA	AEROLINIA
5180146	1114.0	1091.0	1050.0	SEA	HNL	4308.20	236.931267	True	09092008	Hawaiian Airlines Inc.
6980183	790.0	634.0	162.0	EWR	HNL	7985.55	755.730288	True	19122008	Continental Air Lines Inc.
3922427	776.0	346.0	410.0	JFK	SFO	4161.75	721.690724	True	23072008	Delta Air Lines Inc.
4614554	750.0	733.0	612.0	HNL	SEA	4308.20	352.649399	True	19082008	Hawaiian Airlines Inc.
4811552	750.0	597.0	388.0	EWR	HNL	7985.55	802.567855	True	02082008	Continental Air Lines Inc.

In [33]:
```
#ara obtindrem els n=5 vols més llargs en quant a distància (buscarem distancia, i origen i destinació)
#generarem dues noves columnes per visualitzar el nom dels aeroports que mapejarem d'un arxiu de "Suplemental data"
#importem l'arxiu dels aeroports
file = datasets_path + "DelayedFlights-airports.csv"
ap_df = pd.read_csv(file, sep=',', encoding='utf8', index_col=0)
ap_df.sample(2)
```

executed in 68ms, finished 10:08:06 2021-04-14

Out[33]:

iata	airport	city	state	country	lat	long
Y93	Atlanta Municipal	Atlanta	MI	USA	45.000008	-84.133337
MIE	Delaware County	Muncie	IN	USA	40.242348	-85.395860

In [34]:
```
#generem noves columnes mapejant els codis d'origen i destinació amb el dataframe de codis per obtenir les series
#dels noms dels aeroports
aer_origen_s = df["COD_ORIGEN"].map(ap_df["airport"])
aer_desti_s = df["COD_DESTI"].map(ap_df["airport"])

#podem generar tb, del mateix mode, les series de les ciutats d'origen i destinació
city_origen_s = df["CITY_ORIGEN"] = df["COD_ORIGEN"].map(ap_df["city"])
city_desti_s = df["CITY_DESTI"] = df["COD_DESTI"].map(ap_df["city"])

#esborrem les columnes de codis d'origen i destinació
df.drop(columns=["COD_ORIGEN", "COD_DESTI"], inplace=True)

#generem noves columnes amb les series d'aeroports
df["AER_ORIGEN"] = aer_origen_s
df["AER_DESTI"] = aer_desti_s

#podem generar tb del mateix mode les ciutats d'origen i destinació
df["CITY_ORIGEN"] = city_origen_s
df["CITY_DESTI"] = city_desti_s
```

executed in 1.81s, finished 10:08:10 2021-04-14

In [35]:
```
#com les distàncies son fixes entre aeroports, agrupem per distàncies uniques, ordenant-les ascendentment,
#i seleccionem les n=5 mes grans, que seran les 5 últimes
mes_distants = list(np.sort(df["DISTANCIA"].unique())[-(n+1):-1])
mes_distants
```

executed in 81ms, finished 10:08:10 2021-04-14

Out[35]:
```
[6392.3, 6733.48, 6780.15, 6828.43, 7245.25]
```

In [36]:
```
#busquem el primer vol que trobem de cada una de les distàncies (hi haurà diversos vols per cada recorregut)
series_list = []
for distance in mes_distants:
    series_list.append(df[df["DISTANCIA"] == distance].iloc[0])

#creem un nou dataframe i visualitzem els vols amb les n=5 distàncies més llargues
mes_distants_df = pd.DataFrame(series_list)
mes_distants_df[['DISTANCIA', 'AER_ORIGEN', 'CITY_ORIGEN', 'AER_DESTI', 'CITY_DESTI']]
```

executed in 277ms, finished 10:08:13 2021-04-14

Out[36]:

	DISTANCIA	AER_ORIGEN	CITY_ORIGEN	AER_DESTI	CITY_DESTI
438603	6392.30	Honolulu International	Honolulu	Minneapolis-St Paul Intl	Minneapolis
218291	6733.48	Chicago O'Hare International	Chicago	Kahului	Kahului
218265	6780.15	Kona International At Keahole	Kailua/Kona	Chicago O'Hare International	Chicago
218178	6828.43	Chicago O'Hare International	Chicago	Honolulu International	Honolulu
305099	7245.25	Honolulu International	Honolulu	William B Hartsfield-Atlanta Intl	Atlanta

In [37]:
```
#busquem els n=5 vols més enraderits
df.nlargest(n, "RETARD", keep='all')
```

executed in 297ms, finished 10:08:17 2021-04-14

Out[37]:

	DURADA_TOTAL	DURADA_AIRE	RETARD	DISTANCIA	VEL_MITJA	RETARDAT	DATA	AEROLINIA	CITY_ORIGEN	CITY_DESTI	AER_ORIGEN	AER_DESTI
1018798	459.0	437.0	2461.0	6392.30	877.661331	True	03022008	Northwest Airlines Inc.	Honolulu	Minneapolis	Honolulu International	Minneapolis-St Paul Intl
2235378	154.0	132.0	2453.0	1496.69	680.313622	True	10042008	Northwest Airlines Inc.	Charlotte	Minneapolis	Charlotte/Douglas International	Minneapolis-St Paul Intl
2832617	172.0	145.0	1951.0	1746.13	722.536528	True	06052008	Northwest Airlines Inc.	Ft. Myers	Detroit	Southwest Florida International	Detroit Metropolitan-Wayne County
3387883	72.0	50.0	1707.0	489.24	587.088014	True	20062008	American Eagle Airlines Inc.	Little Rock	Dallas-Fort Worth	Adams	Dallas-Fort Worth International
6857047	259.0	192.0	1655.0	1808.90	565.281242	True	19122008	Northwest Airlines Inc.	Boston	Minneapolis	Gen Edw L Logan Intl	Minneapolis-St Paul Intl

In [38]:
```
#esborrem dataframes i series temporals de suport d'una mida considerable per deixar recursos
del ap_df, cancel_df
del aer_origen_s, aer_desti_s, city_origen_s, city_desti_s, any_s, mes_s, dia_s, ap_s
```

executed in 152ms, finished 10:08:19 2021-04-14

3 Exercici

Exporta el data set net i amb les noves columnes a Excel

Ens diu que no és permès un full de més de 1048576 files

In [4]:
```
df.to_excel("DelayedFlights2.xlsx", index=False, float_format="%.2f")
```

executed in 1.17s, finished 10:16:44 2021-04-14

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-4-1d0a707c3938> in <module>
----> 1 df.to_excel("DelayedFlights2.xlsx", index=False, float_format="%.2f")

~\AppData\Roaming\Python\Python38\site-packages\pandas\core\generic.py in to_excel(self, excel_writer, sheet_name, na_rep, float_format, columns, header, index, index_label, startrow, startcol, engine, merge_cells, encoding, inf_rep, verbose, freeze_panes, storage_options)
   2184             inf_rep=inf_rep,
   2185         )
-> 2186         formatter.write(
   2187             excel_writer,
   2188             sheet_name=sheet_name,

~\AppData\Roaming\Python\Python38\site-packages\pandas\io\formats\excel.py in write(self, writer, sheet_name, startrow, startcol, freeze_panes, engine, storage_options)
    801         num_rows, num_cols = self.df.shape
    802         if num_rows > self.max_rows or num_cols > self.max_cols:
--> 803             raise ValueError(
    804                 f"This sheet is too large! Your sheet size is: {num_rows}, {num_cols} "
    805                 f"Max sheet size is: {self.max_rows}, {self.max_cols}"

ValueError: This sheet is too large! Your sheet size is: 1928371, 12 Max sheet size is: 1048576, 16384
```

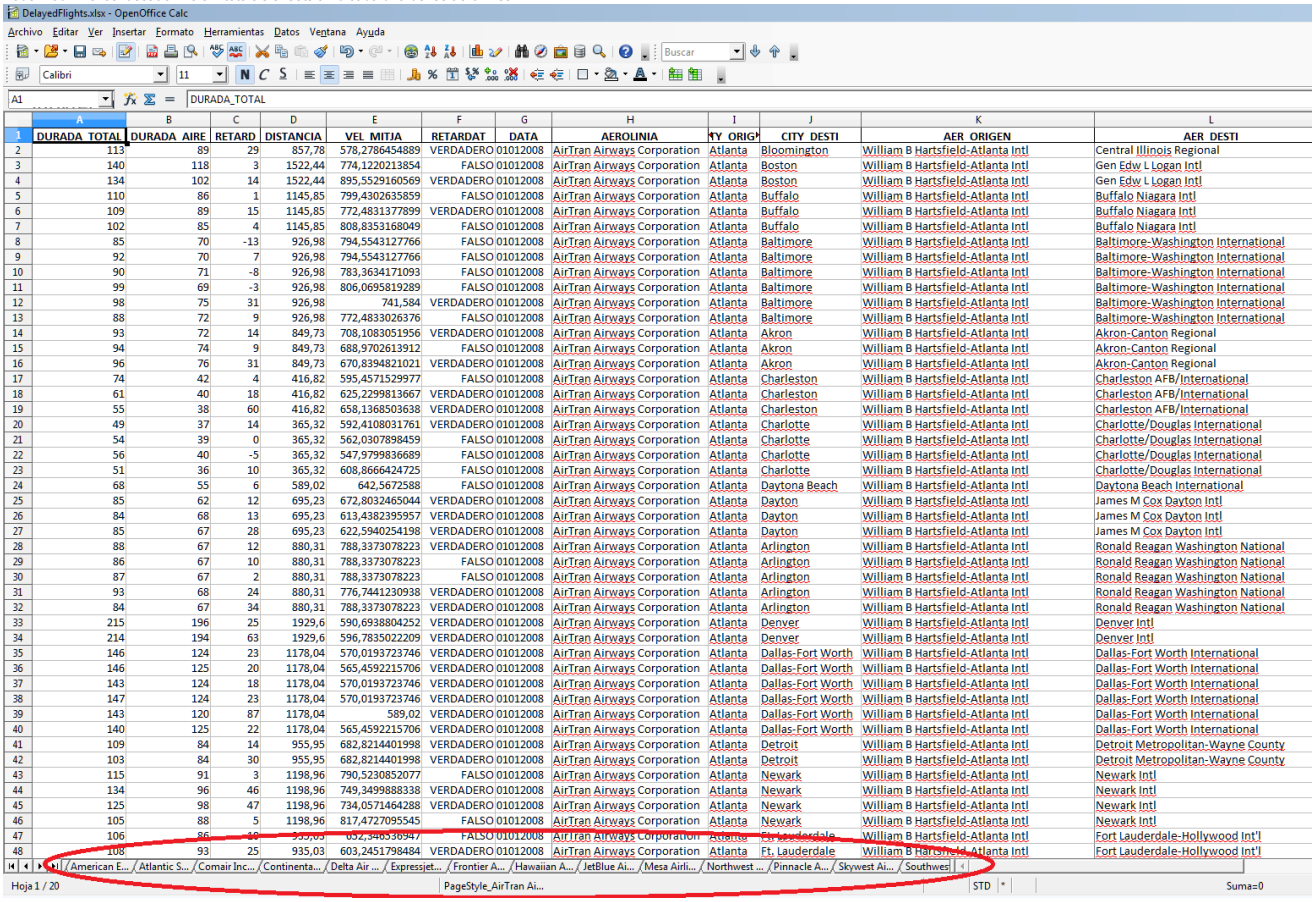
In [39]:
```
#Farem varies fulles, en el mateix excel, i dividirem l'exel en fulles, pe.ex. d'aerolínies
group = df.groupby("AEROLINIA")
group_l = list(group)
```

executed in 806ms, finished 10:21:39 2021-04-14

In [47]:
```
#posarem l'aerolínia de nom de fulla, però si passa dels deu caràcters retallarem el nom doncs si el posem sencer
#ens dona error i tampoc és pràctic (el màxim nombre permès de caràcters per nom de fulla es de 31)
with pd.ExcelWriter('DelayedFlights.xlsx') as writer:
    for i in range(0, len(group_l)):
        aerolinia = group_l[i][0]
        if len(aerolinia) > 10: aerolinia = aerolinia[:10] + "..."
        aerolinia_df = group_l[i][1]
        aerolinia_df.to_excel(writer, sheet_name=aerolinia, index=False)
```

execution queued 10:22:34 2021-04-14

Podem obrir l'excel desat on veiem tots els sheets amb cada una de les aerolínies



	A	B	C	D	E	F	G	H	I	J	K	L
1	DURADA_TOTAL	DURADA_AIRE	RETARD	DISTANCIA	VEL_MITJA	RETARDAT	DATA	AEROLINIA	CITY_ORIGI	CITY_DESTI	AER_ORIGEN	AER_DESTI
2	113	89	29	857,78	578,2786454889	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Bloomington	William B Hartsfield-Atlanta Intl	Central Illinois Regional
3	140	118	3	1522,44	774,1220213854	FALSO	01012008	AirTran Airways Corporation	Atlanta	Boston	William B Hartsfield-Atlanta Intl	Gen Edw L Logan Intl
4	134	102	14	1522,44	895,5529160569	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Boston	William B Hartsfield-Atlanta Intl	Gen Edw L Logan Intl
5	110	86	1	1145,85	799,4302635859	FALSO	01012008	AirTran Airways Corporation	Atlanta	Buffalo	William B Hartsfield-Atlanta Intl	Buffalo Niagara Intl
6	109	89	15	1145,85	772,4831377899	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Buffalo	William B Hartsfield-Atlanta Intl	Buffalo Niagara Intl
7	102	85	4	1145,85	808,8353168049	FALSO	01012008	AirTran Airways Corporation	Atlanta	Buffalo	William B Hartsfield-Atlanta Intl	Buffalo Niagara Intl
8	85	70	-13	926,98	794,5543127766	FALSO	01012008	AirTran Airways Corporation	Atlanta	Baltimore	William B Hartsfield-Atlanta Intl	Baltimore-Washington International
9	92	70	7	926,98	794,5543127766	FALSO	01012008	AirTran Airways Corporation	Atlanta	Baltimore	William B Hartsfield-Atlanta Intl	Baltimore-Washington International
10	90	71	-8	926,98	783,3634171093	FALSO	01012008	AirTran Airways Corporation	Atlanta	Baltimore	William B Hartsfield-Atlanta Intl	Baltimore-Washington International
11	99	69	-3	926,98	806,0695819289	FALSO	01012008	AirTran Airways Corporation	Atlanta	Baltimore	William B Hartsfield-Atlanta Intl	Baltimore-Washington International
12	98	75	31	926,98	741,584	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Baltimore	William B Hartsfield-Atlanta Intl	Baltimore-Washington International
13	88	72	9	926,98	772,4833026376	FALSO	01012008	AirTran Airways Corporation	Atlanta	Baltimore	William B Hartsfield-Atlanta Intl	Baltimore-Washington International
14	93	72	14	849,73	708,1083051956	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Akron	William B Hartsfield-Atlanta Intl	Akron-Canton Regional
15	94	74	9	849,73	688,9702613912	FALSO	01012008	AirTran Airways Corporation	Atlanta	Akron	William B Hartsfield-Atlanta Intl	Akron-Canton Regional
16	96	76	31	849,73	670,8394821021	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Akron	William B Hartsfield-Atlanta Intl	Akron-Canton Regional
17	74	42	4	416,82	595,4571529977	FALSO	01012008	AirTran Airways Corporation	Atlanta	Charleston	William B Hartsfield-Atlanta Intl	Charleston AFB/International
18	61	40	18	416,82	625,2299813667	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Charleston	William B Hartsfield-Atlanta Intl	Charleston AFB/International
19	55	38	60	416,82	658,1368503638	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Charleston	William B Hartsfield-Atlanta Intl	Charleston AFB/International
20	49	37	14	365,32	592,4108031761	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Charlotte	William B Hartsfield-Atlanta Intl	Charlotte/Douglas International
21	54	39	0	365,32	562,0307898459	FALSO	01012008	AirTran Airways Corporation	Atlanta	Charlotte	William B Hartsfield-Atlanta Intl	Charlotte/Douglas International
22	56	40	-5	365,32	547,9799836689	FALSO	01012008	AirTran Airways Corporation	Atlanta	Charlotte	William B Hartsfield-Atlanta Intl	Charlotte/Douglas International
23	51	36	10	365,32	608,8666424725	FALSO	01012008	AirTran Airways Corporation	Atlanta	Charlotte	William B Hartsfield-Atlanta Intl	Charlotte/Douglas International
24	68	55	6	589,02	642,5672588	FALSO	01012008	AirTran Airways Corporation	Atlanta	Daytona Beach	William B Hartsfield-Atlanta Intl	Daytona Beach International
25	85	62	12	695,23	672,8032465044	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Dayton	William B Hartsfield-Atlanta Intl	James M Cox Dayton Intl
26	84	68	13	695,23	613,4382395957	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Dayton	William B Hartsfield-Atlanta Intl	James M Cox Dayton Intl
27	85	67	28	695,23	622,5940254198	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Dayton	William B Hartsfield-Atlanta Intl	James M Cox Dayton Intl
28	88	67	12	880,31	788,3373078223	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Arlington	William B Hartsfield-Atlanta Intl	Ronald Reagan Washington National
29	86	67	10	880,31	788,3373078223	FALSO	01012008	AirTran Airways Corporation	Atlanta	Arlington	William B Hartsfield-Atlanta Intl	Ronald Reagan Washington National
30	87	67	2	880,31	788,3373078223	FALSO	01012008	AirTran Airways Corporation	Atlanta	Arlington	William B Hartsfield-Atlanta Intl	Ronald Reagan Washington National
31	93	68	24	880,31	776,7441230938	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Arlington	William B Hartsfield-Atlanta Intl	Ronald Reagan Washington National
32	84	67	34	880,31	788,3373078223	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Arlington	William B Hartsfield-Atlanta Intl	Ronald Reagan Washington National
33	215	196	25	1929,6	590,6938804252	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Denver	William B Hartsfield-Atlanta Intl	Denver Intl
34	214	194	63	1929,6	596,7835022209	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Denver	William B Hartsfield-Atlanta Intl	Denver Intl
35	146	124	23	1178,04	570,0193723746	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Dallas-Fort Worth	William B Hartsfield-Atlanta Intl	Dallas-Fort Worth International
36	146	125	20	1178,04	565,4592215706	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Dallas-Fort Worth	William B Hartsfield-Atlanta Intl	Dallas-Fort Worth International
37	143	124	18	1178,04	570,0193723746	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Dallas-Fort Worth	William B Hartsfield-Atlanta Intl	Dallas-Fort Worth International
38	147	124	23	1178,04	570,0193723746	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Dallas-Fort Worth	William B Hartsfield-Atlanta Intl	Dallas-Fort Worth International
39	143	120	87	1178,04	589,02	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Dallas-Fort Worth	William B Hartsfield-Atlanta Intl	Dallas-Fort Worth International
40	140	125	22	1178,04	565,4592215706	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Dallas-Fort Worth	William B Hartsfield-Atlanta Intl	Dallas-Fort Worth International
41	109	84	14	955,95	682,8214401998	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Detroit	William B Hartsfield-Atlanta Intl	Detroit Metropolitan-Wayne County
42	103	84	30	955,95	682,8214401998	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Detroit	William B Hartsfield-Atlanta Intl	Detroit Metropolitan-Wayne County
43	115	91	3	1198,96	790,5230852077	FALSO	01012008	AirTran Airways Corporation	Atlanta	Newark	William B Hartsfield-Atlanta Intl	Newark Intl
44	134	96	46	1198,96	749,3499888338	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Newark	William B Hartsfield-Atlanta Intl	Newark Intl
45	125	98	47	1198,96	734,0571464288	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Newark	William B Hartsfield-Atlanta Intl	Newark Intl
46	105	88	5	1198,96	817,4727095545	FALSO	01012008	AirTran Airways Corporation	Atlanta	Newark	William B Hartsfield-Atlanta Intl	Newark Intl
47	106	86	10	935,03	652,946530947	FALSO	01012008	AirTran Airways Corporation	Atlanta	Ft. Lauderdale	William B Hartsfield-Atlanta Intl	Fort Lauderdale-Hollywood Int'l
48	108	93	25	935,03	603,2451798484	VERDADERO	01012008	AirTran Airways Corporation	Atlanta	Ft. Lauderdale	William B Hartsfield-Atlanta Intl	Fort Lauderdale-Hollywood Int'l