Part_I_exploration

March 14, 2022

1 Part I - Data Expo 2009 - Airline on-time performance

1.1 by Samiha Amroune

1.2 Introduction

Get the data

1.3 The data

The data consists of flight arrival and departure details for all commercial flights within the USA, from October 1987 to April 2008. This is a large dataset: there are nearly 120 million records in total, and takes up 1.6 gigabytes of space compressed and 12 gigabytes when uncompressed. So I choose to Analyze data of year **2001**

The data comes originally from RITA where it is described in detail.

Variable descriptions

Name

Description

1

Year

2001

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

These are the carriers currently tracked in ASQP:

IATA Code

ICAO Code

Air Carrier Name

ZW

AWI

Air Wisconsin

AS

ASA

Alaska Airlines

G4

AAY

Allegiant Air LLC

AA

AAL

American Airlines

C5

UCA

Champlain Air

CP

CPZ

Compass Airlines

DL

DAL

Delta Air Lines, Inc.

EM

CFS

Empire Airline

9E

EDV

Endeavor Air

MQ

ENY

Envoy Air

EV

ASQ

ExpressJet Airlines

F9

FFT

Frontier Airlines, Inc.

G7

GJS

GoJet Airlines

HA

HAL

Hawaiian Airlines Inc.

QX

QXE

Horizon Air

B6

JBU

Jetblue Airways Corporation

OH

JIA

Jetstream Intl

YV

ASH

Mesa Airlines, Inc.

KS

NLA

Penair

PT

PDT

Piedmont Airlines

YX

RPA

Republic Airlines

OO

SKW

Skywest Airlines

WN

SWA

Southwest Airlines

NK

NKS

Spirit Airlines, Inc.

AX

LOF

Trans State

UA

UAL

United Airlines, Inc.

1.4 The challenge

The aim of the data expo is to provide a graphical summary of important features of the data set. This is intentionally vague in order to allow different entries to focus on different aspects of the data, but here are a few ideas to get you started:

- When is the best time of day/day of week/time of year to fly to minimise delays?
- Do older planes suffer more delays?
- How does the number of people flying between different locations change over time?
- How well does weather predict plane delays?
- Can you detect cascading failures as delays in one airport create delays in others? Are there critical links in the system?
- compare flight patterns before and after 9/11, or between the pair of cities that you fly between most often, or all flights to and from a major airport like Chicago (ORD).

1.5 Preliminary Wrangling

[1]:
```
#import all packages and set plots to be embedded inline
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from scipy import stats
from IPython.display import display
from IPython.core.display import HTML
%matplotlib inline
```

[2]:
```
pd.set_option('display.max_columns', None)
pd.set_option('display.width', None)
```

UnicodeDecodeError, utf-8 invalid continuation byte

[3]:
```
#Load dataset
df_01 = pd.read_csv("2001.csv", encoding ='ISO-8859-1')
print(df_01.shape)
df_01.head()
```

```
(5967780, 29)
```

[3]:
```
   Year  Month  DayofMonth  DayOfWeek  DepTime  CRSDepTime  ArrTime  \
0  2001      1          17          3   1806.0        1810   1931.0
1  2001      1          18          4   1805.0        1810   1938.0
2  2001      1          19          5   1821.0        1810   1957.0
3  2001      1          20          6   1807.0        1810   1944.0
4  2001      1          21          7   1810.0        1810   1954.0

   CRSArrTime UniqueCarrier  FlightNum TailNum  ActualElapsedTime  \
0        1934            US        375  N700äæ               85.0
1        1934            US        375  N713äæ               93.0
2        1934            US        375  N702äæ               96.0
3        1934            US        375  N701äæ               97.0
4        1934            US        375  N768äæ              104.0

   CRSElapsedTime  AirTime  ArrDelay  DepDelay Origin Dest  Distance  TaxiIn  \
```

```
0              84     60.0      -3.0      -4.0    BWI  CLT       361       5
1              84     64.0       4.0      -5.0    BWI  CLT       361       9
2              84     80.0      23.0      11.0    BWI  CLT       361       6
3              84     66.0      10.0      -3.0    BWI  CLT       361       4
4              84     62.0      20.0       0.0    BWI  CLT       361       4

   TaxiOut  Cancelled  CancellationCode  Diverted  CarrierDelay  WeatherDelay  \
0       20          0               NaN         0           NaN           NaN
1       20          0               NaN         0           NaN           NaN
2       10          0               NaN         0           NaN           NaN
3       27          0               NaN         0           NaN           NaN
4       38          0               NaN         0           NaN           NaN

   NASDelay  SecurityDelay  LateAircraftDelay
0       NaN            NaN                NaN
1       NaN            NaN                NaN
2       NaN            NaN                NaN
3       NaN            NaN                NaN
4       NaN            NaN                NaN
```

[4]:
```
df_01.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5967780 entries, 0 to 5967779
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     int64
 13  AirTime            float64
 14  ArrDelay           float64
 15  DepDelay           float64
 16  Origin             object
 17  Dest               object
 18  Distance           int64
 19  TaxiIn             int64
 20  TaxiOut            int64
```

```
 21  Cancelled          int64
 22  CancellationCode   float64
 23  Diverted           int64
 24  CarrierDelay       float64
 25  WeatherDelay       float64
 26  NASDelay           float64
 27  SecurityDelay      float64
 28  LateAircraftDelay  float64
dtypes: float64(12), int64(13), object(4)
memory usage: 1.3+ GB
```

[5]:
```
#Check null values
df_01.isnull().sum()
```

[5]:
```
Year                       0
Month                      0
DayofMonth                 0
DayOfWeek                  0
DepTime               231198
CRSDepTime                 0
ArrTime               244107
CRSArrTime                 0
UniqueCarrier              0
FlightNum                  0
TailNum                    0
ActualElapsedTime     244107
CRSElapsedTime             0
AirTime               244107
ArrDelay              244107
DepDelay              231198
Origin                     0
Dest                       0
Distance                   0
TaxiIn                     0
TaxiOut                    0
Cancelled                  0
CancellationCode     5967780
Diverted                   0
CarrierDelay         5967780
WeatherDelay         5967780
NASDelay             5967780
SecurityDelay        5967780
LateAircraftDelay    5967780
dtype: int64
```

[6]:
```
#Check value_counts of Cancelled col
df_01.Cancelled.value_counts()
```

[6]:
```
0    5736582
1     231198
Name: Cancelled, dtype: int64
```

https://stackoverflow.com/questions/18648626/for-loop-with-two-variables

[7]:
```
# Convert dtypes of interest variables
interest = [
    'AirTime',
    'DepDelay',
    'ArrDelay',
    'TaxiIn',
    'TaxiOut'
]

interst2 = [
    'DayOfWeek',
    'Month',
    'FlightNum',
    'DayofMonth'
]

for i, n in zip(interest, interst2):
    #update NaN time values in variables of interest to 0
    df_01[i] = df_01[i].fillna(0)
    #As the values are in minutes convert dtype to int
    df_01[i] = df_01[i].astype('Int64')
    # convert cols to str for better visual analysis
    df_01[n] = df_01[n].astype('str')
```

[8]:
```
# Change format to hours
df_01['CRSDepTime'] = pd.to_datetime(df_01.CRSDepTime, format='%H',␣
 ↪exact=False).dt.hour
df_01['CRSArrTime'] = pd.to_datetime(df_01.CRSArrTime, format='%H',␣
 ↪exact=False).dt.hour
```

[9]:
```
# Drop columns that contains 5967780 null values
df_01.drop(columns=[
    'Year',# Since the dataset from year 2001 no need to keep Year col
    'CancellationCode',
    'CarrierDelay',
    'WeatherDelay',
    'NASDelay',
    'SecurityDelay',
    'LateAircraftDelay'
], inplace=True)
```

[10]:
```
#check if There are duplicates
df_01.duplicated().sum()
```

[10]: 0

[11]:
```
# Describe df_01
df_01.describe()
```

[11]:
```
            DepTime     CRSDepTime       ArrTime     CRSArrTime  \
count  5.736582e+06  5.967780e+06  5.723673e+06  5.967780e+06
mean   1.348705e+03  1.318335e+01  1.489809e+03  1.480912e+01
std    4.826860e+02  4.676512e+00  5.111805e+02  4.763519e+00
min    1.000000e+00  1.000000e+00  1.000000e+00  1.000000e+00
25%    9.300000e+02  9.000000e+00  1.110000e+03  1.100000e+01
50%    1.333000e+03  1.300000e+01  1.522000e+03  1.500000e+01
75%    1.740000e+03  1.700000e+01  1.920000e+03  1.900000e+01
max    2.400000e+03  2.300000e+01  2.400000e+03  2.300000e+01

       ActualElapsedTime  CRSElapsedTime       AirTime      ArrDelay  \
count       5.723673e+06    5.967780e+06  5.967780e+06  5.967780e+06
mean        1.250339e+02    1.274760e+02  9.900964e+01  5.302120e+00
std         7.070398e+01    7.036913e+01  6.916640e+01  3.079926e+01
min        -7.190000e+02    0.000000e+00  0.000000e+00 -1.116000e+03
25%         7.200000e+01    7.500000e+01  5.000000e+01 -9.000000e+00
50%         1.060000e+02    1.080000e+02  8.100000e+01 -1.000000e+00
75%         1.580000e+02    1.600000e+02  1.310000e+02  9.000000e+00
max         7.790000e+02    1.440000e+03  7.070000e+02  1.688000e+03

           DepDelay      Distance        TaxiIn       TaxiOut     Cancelled  \
count  5.967780e+06  5.967780e+06  5.967780e+06  5.967780e+06  5.967780e+06
mean   7.838911e+00  7.330293e+02  6.120620e+00  1.483022e+01  3.874104e-02
std    2.783844e+01  5.740716e+02  4.798693e+00  1.030761e+01  1.929771e-01
min   -2.040000e+02  2.100000e+01  0.000000e+00  0.000000e+00  0.000000e+00
25%   -3.000000e+00  3.130000e+02  3.000000e+00  9.000000e+00  0.000000e+00
50%    0.000000e+00  5.710000e+02  5.000000e+00  1.300000e+01  0.000000e+00
75%    6.000000e+00  9.800000e+02  7.000000e+00  1.800000e+01  0.000000e+00
max    1.692000e+03  4.962000e+03  3.290000e+02  5.020000e+02  1.000000e+00

           Diverted
count  5.967780e+06
mean   2.163116e-03
std    4.645898e-02
min    0.000000e+00
25%    0.000000e+00
50%    0.000000e+00
75%    0.000000e+00
max    1.000000e+00
```

https://stackoverflow.com/questions/28683216/python-int-object-has-no-attribute-sort

[12]:
```
# Get the lowest 10 values from ArrDelay to extract data
print(sorted(i ** 2 for i in df_01.ArrDelay.unique())[:10])
```

```
[0, 1, 1, 4, 4, 9, 9, 16, 16, 25]
```

[13]:
```
# Get the lowest 10 values from DepDelay to extract data
print(sorted(i ** 2 for i in df_01.DepDelay.unique())[:10])
```

```
[0, 1, 1, 4, 4, 9, 9, 16, 16, 25]
```

https://stackoverflow.com/questions/54759936/extension-dtypes-in-pandas-appear-to-have-a-bug-with-query

[14]:
```
#Get copy from our dataframe that have cancelled flights and reset index in new␣
 ↪df
df1 = df_01.query('ArrDelay >= 16 or DepDelay >=16 or Cancelled == 1',␣
 ↪engine='python')
# print shape
print(df1.shape)
#display 5 rows
df1.head()
```

```
(1448618, 22)
```

[14]:
```
    Month  DayofMonth  DayOfWeek  DepTime  CRSDepTime  ArrTime  CRSArrTime  \
2       1          19          5   1821.0          18   1957.0          19
4       1          21          7   1810.0          18   1954.0          19
15      1           1          1   1000.0           9   1112.0          10
16      1           2          2   1120.0           9   1230.0          10
20      1           6          6      NaN           9      NaN          10

   UniqueCarrier FlightNum TailNum  ActualElapsedTime  CRSElapsedTime  \
2             US       375  N702ää               96.0              84
4             US       375  N768ää              104.0              84
15            US       376  N300Aä               72.0              74
16            US       376  N375ää               70.0              74
20            US       376  äNKNOæ                NaN              74

    AirTime  ArrDelay  DepDelay Origin Dest  Distance  TaxiIn  TaxiOut  \
2        80        23        11    BWI  CLT       361       6       10
4        62        20         0    BWI  CLT       361       4       38
15       53        18        20    PHL  MHT       290       5       14
16       54        96       100    PHL  MHT       290       7        9
20        0         0         0    PHL  MHT       290       0        0

    Cancelled  Diverted
2           0         0
```

```
4           0         0
15          0         0
16          0         0
20          1         0
```

[15]:
```
#export clean dataframe to csv for later use
df1.to_csv('d_2001.csv', index = False)
```

[16]:
```
#Read df
df = pd.read_csv('d_2001.csv')
print(df.shape)
df.head()
```

```
(1448618, 22)
```

[16]:
```
   Month  DayofMonth  DayOfWeek  DepTime  CRSDepTime  ArrTime  CRSArrTime  \
0      1          19          5   1821.0          18   1957.0          19
1      1          21          7   1810.0          18   1954.0          19
2      1           1          1   1000.0           9   1112.0          10
3      1           2          2   1120.0           9   1230.0          10
4      1           6          6      NaN           9      NaN          10

  UniqueCarrier  FlightNum TailNum  ActualElapsedTime  CRSElapsedTime  \
0            US        375  N702äæ               96.0              84
1            US        375  N768äæ              104.0              84
2            US        376  N300Aä               72.0              74
3            US        376  N375ää               70.0              74
4            US        376  äNKNOæ                NaN              74

   AirTime  ArrDelay  DepDelay Origin Dest  Distance  TaxiIn  TaxiOut  \
0       80        23        11    BWI  CLT       361       6       10
1       62        20         0    BWI  CLT       361       4       38
2       53        18        20    PHL  MHT       290       5       14
3       54        96       100    PHL  MHT       290       7        9
4        0         0         0    PHL  MHT       290       0        0

   Cancelled  Diverted
0          0         0
1          0         0
2          0         0
3          0         0
4          1         0
```

1.5.1 What is the structure of your dataset?

There are **5.967.780** flights in the original dataset with **29** columns **('Year', 'Month', 'DayofMonth', 'DayOfWeek', 'DepTime', 'CRSDepTime', 'ArrTime', 'CRSArrTime', 'UniqueCarrier', 'FlightNum', 'TailNum','ActualElapsedTime',**

'CRSElapsedTime', 'AirTime', 'ArrDelay','DepDelay', 'Origin', 'Dest', 'Distance', 'TaxiIn', 'TaxiOut', 'Cancelled', 'Diverted','CancellationCode','CarrierDelay','WeatherDelay','NASDelay','SecurityDelay','LateAi Most variables are numeric in nature, but the variables **'UniqueCarrier', 'TailNum', 'Origin' and 'Dest'** are strings

1.5.2 What is/are the main feature(s) of interest in your dataset?

I'm most interested in figuring out: 1- What are the Most top 10 delays and Cancellation by: ('UniqueCarrier', 'Origin', 'Dest')? 2- What are the most week days have Delays and cancelation? 3- What are most day of Month with a flight delay or cancellation? 4- What are most Month with a flight delay or cancellation? 5- Was the delay in this month affected by the events of September 11? 6- What are the most Scheduled Departure Time with a flight delay or cancellation? 7- What is most Scheduled Time Of Arrival with a flight delay or cancellation ? 8- What are most delay or cancellation flight by Air Time? 9- What are Most 10 TailNum a flight delay or cancellation? 10- What are the most 10 FlightNum a flight delay or cancellation? 11- What is the most Distance in miles a flight delay or cancellation?

1.5.3 What features in the dataset do you think will help support your investigation into your feature(s) of interest?

I expect that the events of September 11 will have an impact on the largest number of canceled flights

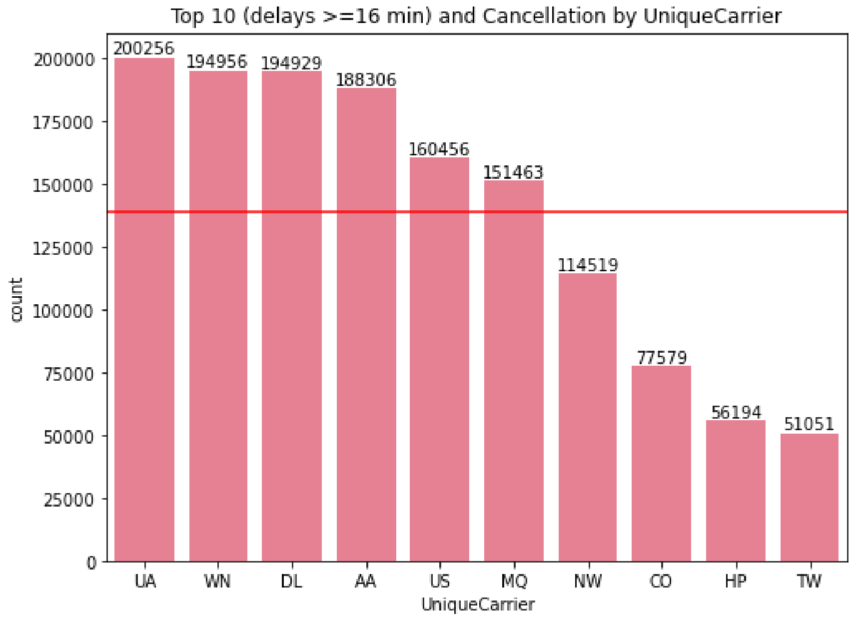
1.6 Univariate Exploration

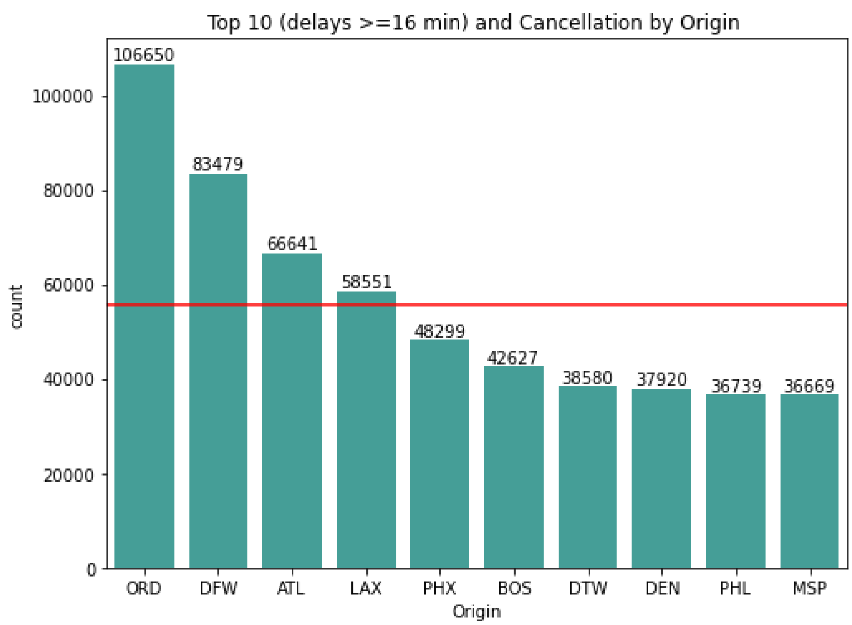
I'll start by looking at the Top 10 Delays and Cancellation by Carrier: ('UniqueCarrier', 'Origin', 'Dest')

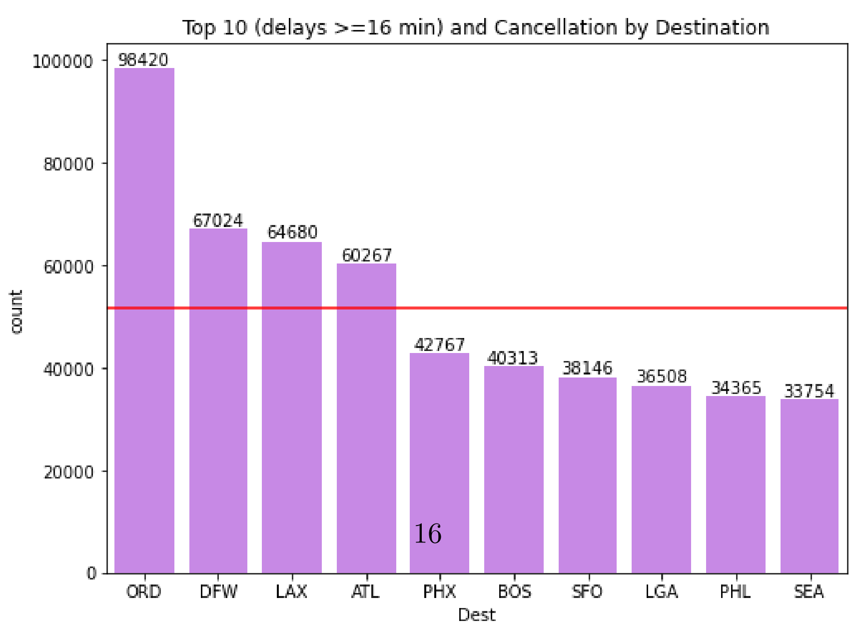
2 What are the Most top 10 delays and Cancellation by: ('UniqueCarrier', 'Origin', 'Dest')?

[17]:
```
# let's plot 'UniqueCarrier, Origin and Dest' to get an idea of the␣
 ↪distribution.
# Set plot size
fig, ax = plt.subplots(nrows=3, figsize = [8,20])
#set plot color
colors = sns.color_palette("husl",10)

# Set order to sort bars
orders0 = df['UniqueCarrier'].value_counts().head(10).index
orders1 = df['Origin'].value_counts().head(10).index
orders2 = df['Dest'].value_counts().head(10).index

# Define Plots
#Top 10 delays and Cancellation by UniqueCarrier
```

```python
ax1 = sns.countplot(
    x = 'UniqueCarrier',
    data = df,
    color=colors[0],
    order = orders0, ax = ax[0]
)
#Top 10 delays and Cancellation by Origin
ax2= sns.countplot(
    x = 'Origin',
    data = df,
    color=colors[5],
    order = orders1,
    ax = ax[1],
)
#Top 10 delays and Cancellation by Dest
ax3=sns.countplot(
    x = 'Dest',
    data = df,
    color=colors[8],
    order = orders2, ax = ax[2]
)
# Set count labels
ax1.bar_label(ax1.containers[0])
ax2.bar_label(ax2.containers[0])
ax3.bar_label(ax3.containers[0])

# Add mean line
ax1.axhline(df['UniqueCarrier'].value_counts().head(10).mean(), c='red')
ax2.axhline(df['Origin'].value_counts().head(10).mean(), c='red')
ax3.axhline(df['Dest'].value_counts().head(10).mean(), c='red')

# Set titels
ax1.set(title='Top 10 (delays >=16 min) and Cancellation by UniqueCarrier')
ax2.set(title='Top 10 (delays >=16 min) and Cancellation by Origin ')
ax3.set(title='Top 10 (delays >=16 min) and Cancellation by Destination')


plt.show();
```


Top 10 (delays >=16 min) and Cancellation by UniqueCarrier
count
UniqueCarrier
200256
194956
194929
188306
160456
151463
114519
77579
56194
51051
UA
WN
DL
AA
US
MQ
NW
CO
HP
TW


Top 10 (delays >=16 min) and Cancellation by Origin
count
Origin
106650
83479
66641
58551
48299
42627
38580
37920
36739
36669
ORD
DFW
ATL
LAX
PHX
BOS
DTW
DEN
PHL
MSP


Top 10 (delays >=16 min) and Cancellation by Destination
count
Dest
98420
67024
64680
60267
42767
40313
38146
36508
34365
33754
ORD
DFW
LAX
ATL
PHX
BOS
SFO
LGA
PHL
SEA

Airline and Location Code Search

2.1 The most top 10 Delays or cancelation are:

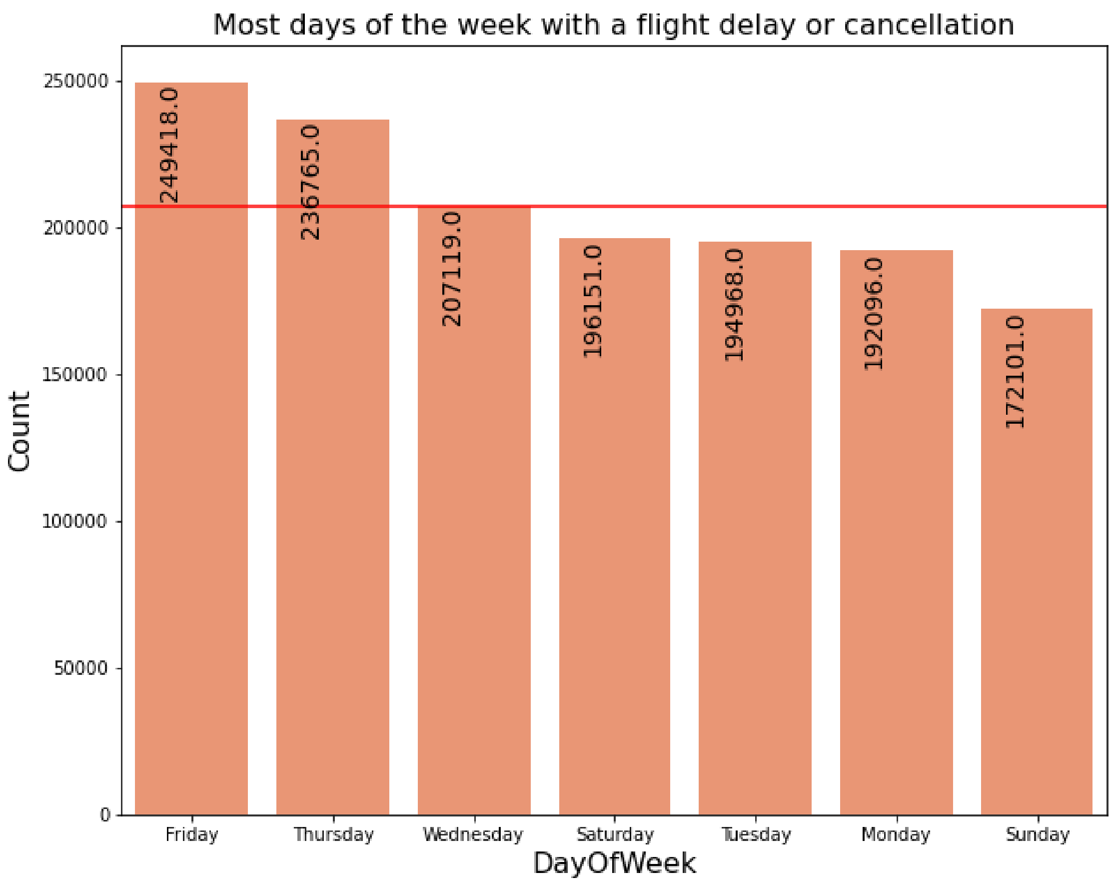
- **From plot 1 :** United Airlines, Inc.**(UA)**, followed by Southwest Airlines Co.**(WN)**, then Delta Air Lines, Inc.**(DL)**, had the most delays
- **From plot 2 :** Chicago O'Hare International Airport **(ORD)** ,Dallas/Ft Worth Intl **(DFW)**, Atlanta Hartsfield-Jackson Int
- **From plot 3 :** Chicago O'Hare International Airport **(ORD)**, Dallas/Ft Worth Intl **(DFW)**, Los Angeles **(LAX)**

2.2 What are the most week days have Delays and cancelation?

https://www.tutorialspoint.com/matplotlib-how-to-show-the-count-values-on-the-top-of-a-bar-in-a-countplot

[18]:
```
# plot 'DayOfWeek' to get an idea of the distribution.
# Set plot size
fig, ax = plt.subplots(figsize = [10,8])
# Set Plot Color
colors = sns.color_palette("Set2",10)
# Set order values
orders = df['DayOfWeek'].value_counts().index

# Using print to get orders value
#print(orders)

# Define plot
ax1 = sns.countplot(
    x = 'DayOfWeek',
    data = df,
    color=colors[1],
    order = orders
)

plt.axhline(df['DayOfWeek'].value_counts().head(10).mean(), c='red')

# Set labels day
week_day = ['Friday', 'Thursday', 'Wednesday', 'Saturday', 'Tuesday', 'Monday',␣
↪'Sunday']
ax.set_xticklabels(week_day);

# show the count values
for p in ax.patches:
```

```
    ax.annotate('{:.1f}'.format(p.get_height()), (p.get_x()+0.25, p.
 ↪get_height()+0.01),rotation = 90, horizontalalignment='center',␣
 ↪verticalalignment='top',
                size=14)

# Set labels fontsize
plt.ylabel('Count', fontsize=16);
plt.xlabel('DayOfWeek', fontsize=16);

# Set title
plt.title('Most days of the week with a flight delay or cancellation',␣
 ↪fontsize=16)
plt.show();
```

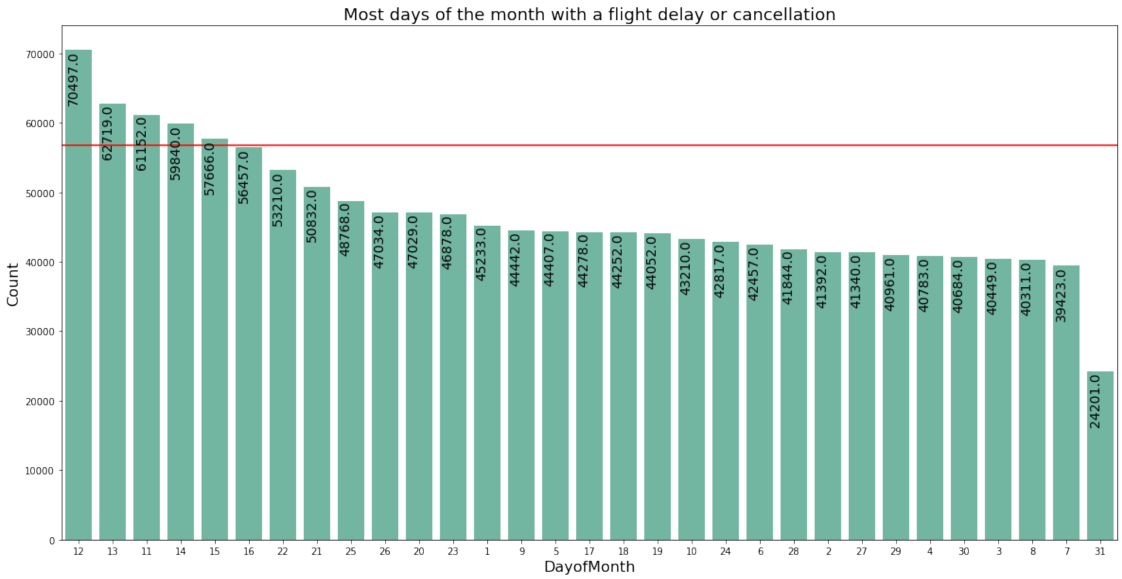


2.3 Most days of the week with a flight delay or cancellation are:

Friday, Thursday and Wednesday Saturday

2.4 What are most day of Month with a flight delay or cancellation?

[19]:
```python
# plot 'DayofMonth' to get an idea of the distribution.
# Set plot size
fig, ax = plt.subplots(figsize = [20,10])
# Set Plot Color
colors = sns.color_palette("Set2")
# Set order values
orders = df['DayofMonth'].value_counts().index

# Define plot
sns.countplot(
    x = 'DayofMonth',
    data = df,
    color=colors[0],
    order = orders,
)
plt.axhline(df['DayofMonth'].value_counts().head(10).mean(), c='red')

# show the count values
for p in ax.patches:
   ax.annotate('{:.1f}'.format(p.get_height()), (p.get_x()+0.25, p.
 ↪get_height()+0.01),rotation = 90, horizontalalignment='center',␣
 ↪verticalalignment='top',
              size=14)

# Set labels fontsize
plt.ylabel('Count', fontsize=16);
plt.xlabel('DayofMonth', fontsize=16);

# Set title
plt.title('Most days of the month with a flight delay or cancellation',␣
 ↪fontsize=18)

plt.show();
```



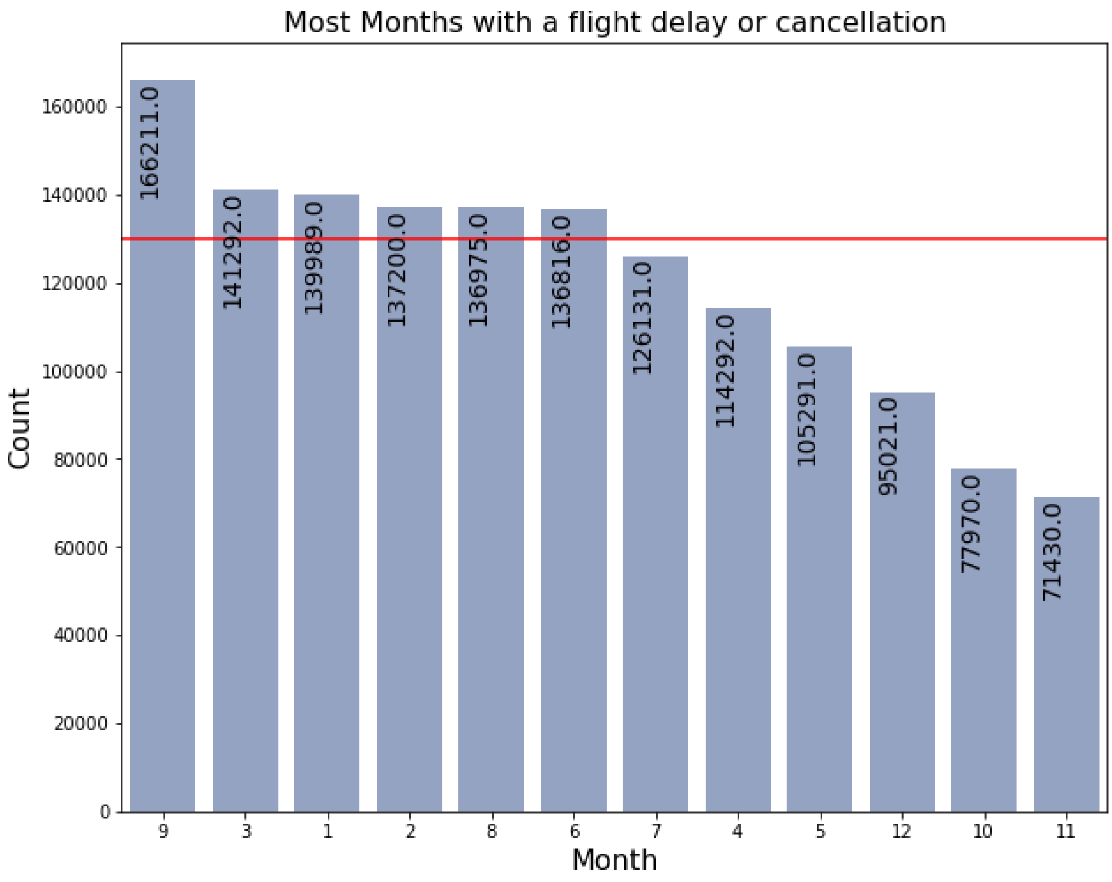
2.5 Most days of the Month with a flight delay or cancellation are:

12, 13, 11

2.6 What are most Month with a flight delay or cancellation?

[20]:
```
# plot 'Month' to get an idea of the distribution.
# Set plot size
fig, ax = plt.subplots(figsize = [10,8])
# Set Plot Color
colors = sns.color_palette("Set2")
# Set order values
orders = df['Month'].value_counts().index

# Define plot
sns.countplot(
    x = 'Month',
    data = df,
    color=colors[2],
    order = orders,
)

plt.axhline(df['Month'].value_counts().head(10).mean(), c='red')

# show the count values
for p in ax.patches:
   ax.annotate('{:.1f}'.format(p.get_height()),
               (p.get_x()+0.25, p.get_height()+0.01),
```

```
                rotation = 90,
                horizontalalignment='center',
                verticalalignment='top',
               size=14
               )

# Set labels fontsize
plt.ylabel('Count', fontsize=16);
plt.xlabel('Month', fontsize=16);

# Set title
plt.title('Most Months with a flight delay or cancellation', fontsize=16)
# Show plot
plt.show();
```

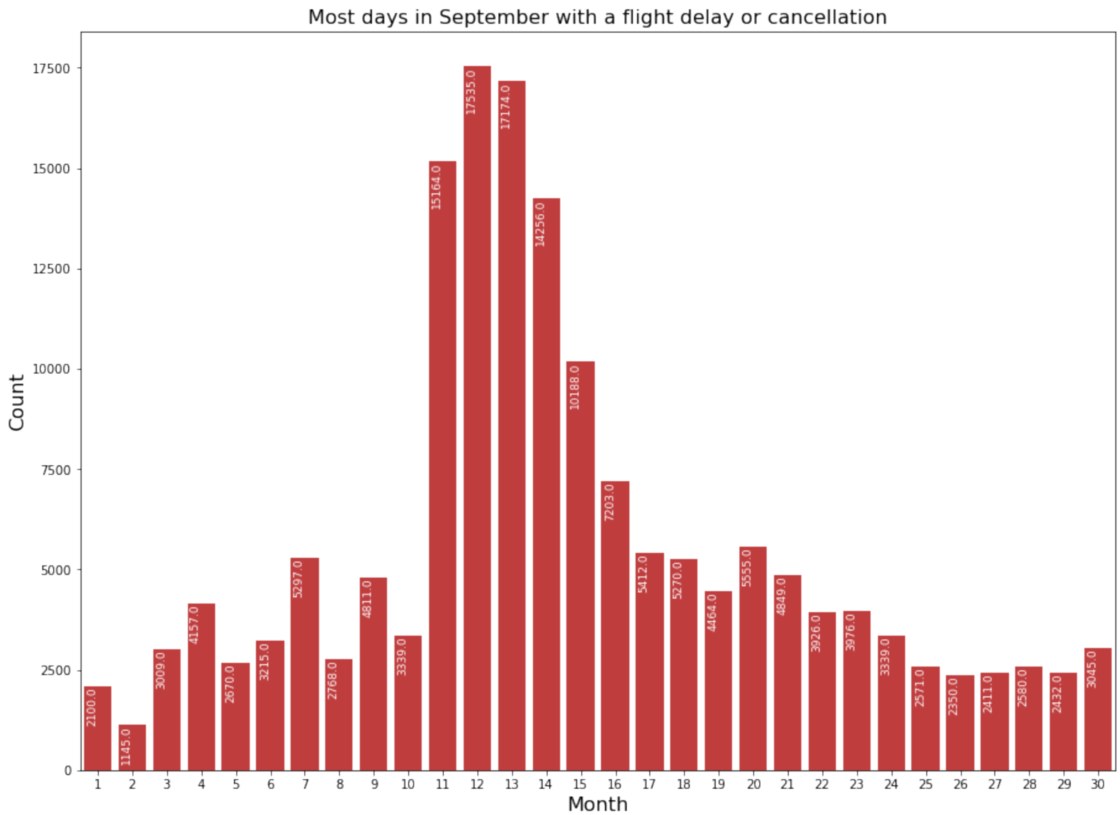


2.7 Most Month with a flight delay or cancellation is:

September had the most delay or cancellation flight; It may have something to do with the September 11 attacks

2.8 Was the delay in this month affected by the events of September 11?

[21]:
```
# plot 'DayofMonth' in September to get an idea of the distribution.
# Set plot size
fig, ax = plt.subplots(figsize = [15,11])

# Set Plot Color
colors = sns.color_palette()

# Set plot data
pl = pd.DataFrame(df.query('Month == 9'))

# Define plot
sns.countplot(
    x = 'DayofMonth',
    data = pl,
    color=colors[3],
)

# show the count values
for p in ax.patches:
   ax.annotate('{:.1f}'.format(p.get_height()),
               (p.get_x()+0.25, p.get_height()+0.01),
               rotation = 90,
               horizontalalignment='center',
               verticalalignment='top',
              size=9,
               color='White'
              )

# Set labels fontsize
plt.ylabel('Count', fontsize=16);
plt.xlabel('Month', fontsize=16);

# Set title
plt.title('Most days in September with a flight delay or cancellation',␣
 ↪fontsize=16)
# Show plot
plt.show();
```



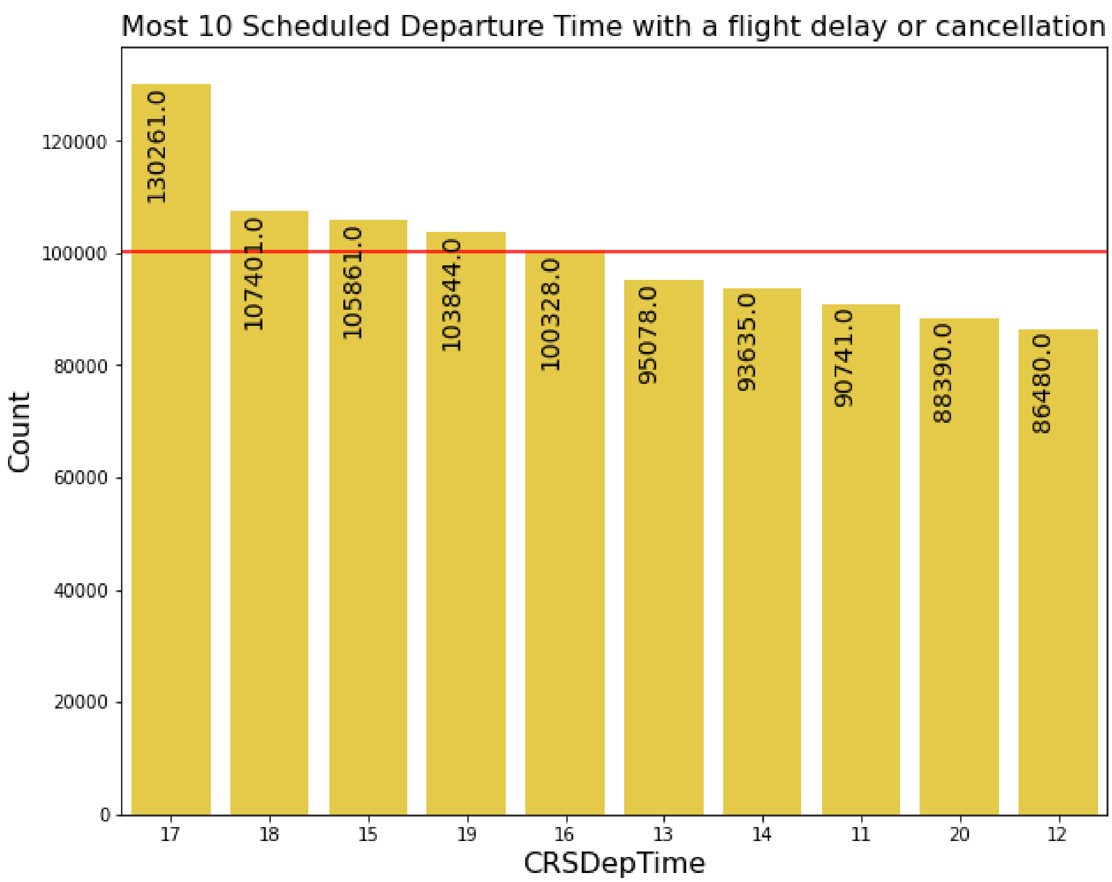
2.9 Highest days of delayed or canceled flights in September:

Days 11, 12, 13, 14 The days with the most flight delays or cancellations

2.10 What are the most Scheduled Departure Time with a flight delay or cancellation?

[22]:
```python
# plot 'CRSDepTime' to get an idea of the distribution.
# Set plot size
fig, ax = plt.subplots(figsize = [10,8])
# Set Plot Color
colors = sns.color_palette("Set2")
# Set order values
orders = df['CRSDepTime'].value_counts().head(10).index

# Define plot
sns.countplot(
    x = 'CRSDepTime',
    data = df,
    color=colors[5],
    order = orders,
```

```
)
# Set title
plt.title('Most 10 Scheduled Departure Time with a flight delay or␣
 ↪cancellation', fontsize=16);
plt.axhline(df['CRSDepTime'].value_counts().head(10).mean(), c='red')

# Set labels fontsize
plt.ylabel('Count', fontsize=16);
plt.xlabel('CRSDepTime', fontsize=16);

# show the count values
for p in ax.patches:
    ax.annotate('{:.1f}'.format(p.get_height()),
                (p.get_x()+0.25, p.get_height()+0.01),
                rotation = 90,
                horizontalalignment='center',
                verticalalignment='top',
               size=14
               )
plt.show();
```

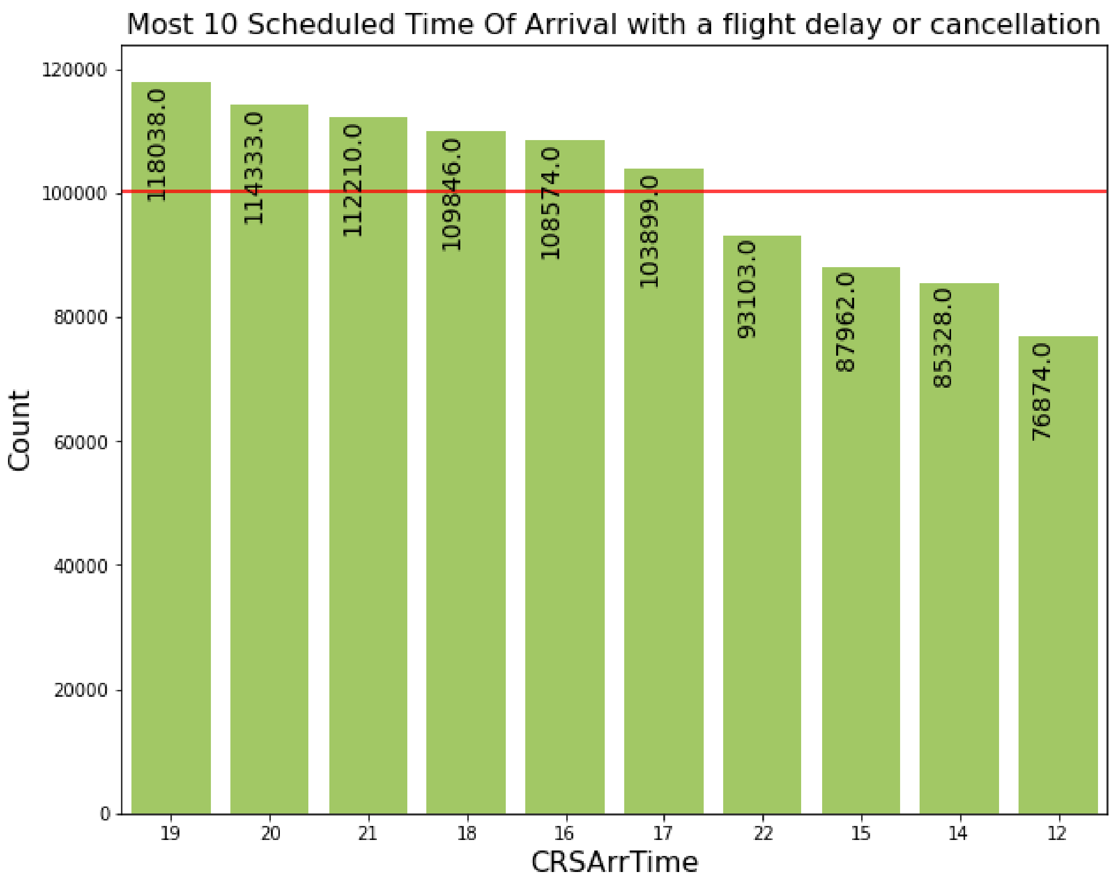


3 Scheduled Departure Time that have the most delays or cancelation are :

Between 15:00 to 17:00 (3-5 pm) Maximum delay or cancellation is at 17:00

3.1 What is most Scheduled Time Of Arrival with a flight delay or cancellation ?

[23]:
```
# plot 'CRSArrTime' to get an idea of the distribution.
# Set plot size
fig, ax = plt.subplots(figsize = [10,8])

# Set Plot Color
colors = sns.color_palette("Set2")

# Set order values
orders = df['CRSArrTime'].value_counts().head(10).index

# Define plot
sns.countplot(
    x = 'CRSArrTime',
    data = df,
    color=colors[4],
    order = orders,
)
# Set title
plt.title('Most 10 Scheduled Time Of Arrival with a flight delay or␣
 ↪cancellation', fontsize=16);
# Set labels fontsize
plt.ylabel('Count', fontsize=16);
plt.xlabel('CRSArrTime', fontsize=16);

# Add mean line
plt.axhline(df['CRSDepTime'].value_counts().head(10).mean(), c='red')

# show the count values
for p in ax.patches:
   ax.annotate('{:.1f}'.format(p.get_height()),
               (p.get_x()+0.25, p.get_height()+0.01),
               rotation = 90,
               horizontalalignment='center',
               verticalalignment='top',
              size=14
              )
plt.show();
```



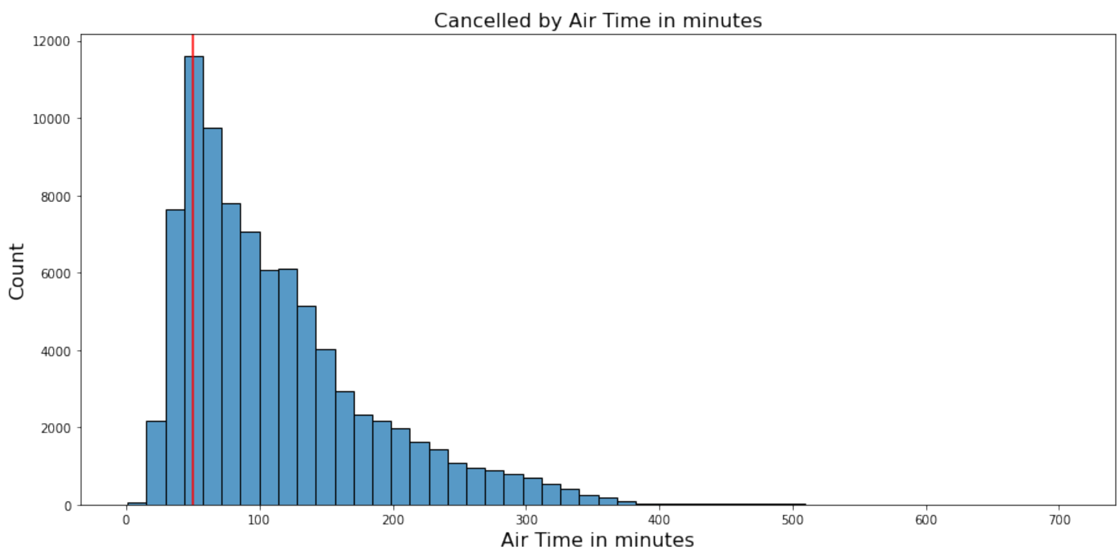
3.2 Scheduled Time Of Arrival that have the most delays or cancelation are :

Between 19:00 to 21:00 (7-9 pm) Maximum delay or cancellation is at 19:00

3.3 What are most delay or cancellation flight by Air Time?

[24]:
```
#define plot
fig, ax = plt.subplots(figsize=(15,7))

#set plot color
colors = sns.color_palette()

#generate data
ar_data = df[(df['Cancelled']== 0) & (df['AirTime']> 0)]


sns.histplot(
    data=ar_data,
    x = 'AirTime',
```

```
    bins=50,
    stat = "frequency" #show the number of observations divided by the bin width
)

#set title and axis

plt.title('Cancelled by Air Time in minutes', fontsize=16);
plt.xlabel('Air Time in minutes', fontsize=16);
plt.ylabel('Count', fontsize=16);


#plot mean line
plt.axvline(50, c='red')

#display plot
plt.show()
```



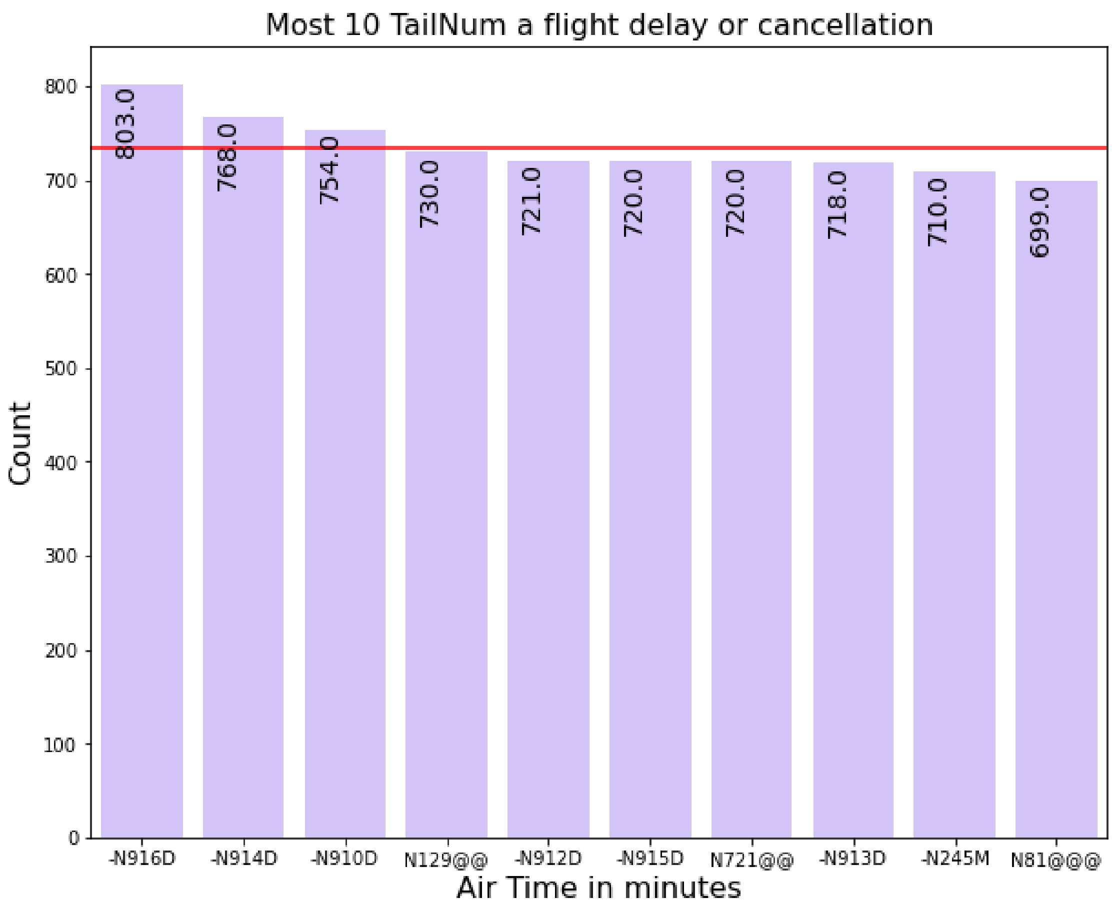
3.4 Most delay or cancellation flight by Air Time:

Airtime on short flights of 50 minutes or less has the greatest cancelled flights

3.5 What are Most 10 TailNum a flight delay or cancellation?

[25]:
```
# plot 'TailNum' to get an idea of the distribution.
# Set plot size
fig, ax = plt.subplots(figsize = [10,8])
# Set Plot Color
colors = sns.color_palette("pastel")
# Extract data
```

```
ex_data = df[df['TailNum'] != "äNKNOæ"]['TailNum'].value_counts().head(10)
# Set order values
orders = ex_data.head(10).index

# Define plot
sns.countplot(
    x = 'TailNum',
    data = df,
    color=colors[4],
    order = orders,
)
# show the count values
for p in ax.patches:
   ax.annotate('{:.1f}'.format(p.get_height()),
               (p.get_x()+0.25, p.get_height()+0.01),
               rotation = 90,
               horizontalalignment='center',
               verticalalignment='top',
              size=14
              )
# Set title
plt.title('Most 10 TailNum a flight delay or cancellation', fontsize=16);
plt.xlabel('Air Time in minutes', fontsize=16);
plt.ylabel('Count', fontsize=16);
# plot mean line
plt.axhline(ex_data.mean(), c='red')

plt.show();
```



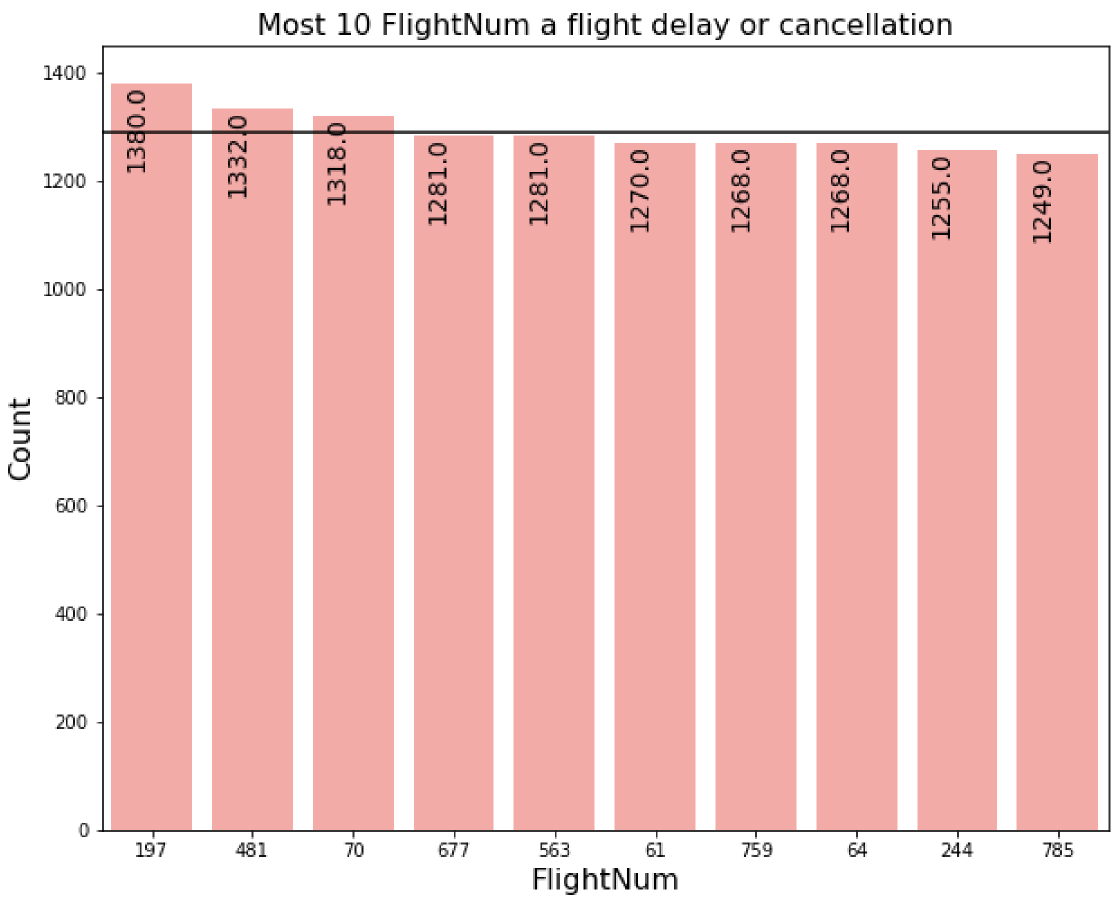
3.6 Most 10 TailNum a flight delay or cancellation are:

Plane -N916D has 803 delays and cancellations followed by plane -N914 and plane -N910

3.7 What are the most 10 FlightNum a flight delay or cancellation?

[26]:
```
# plot 'FlightNum' to get an idea of the distribution.
# Set plot size
fig, ax = plt.subplots(figsize = [10,8])
# Set Plot Color
colors = sns.color_palette("pastel")
# Extract data
ex_data = df['FlightNum'].value_counts().head(10)
# Set order values
orders = ex_data.head(10).index

# Define plot
sns.countplot(
    x = 'FlightNum',
```

```
    data = df,
    color=colors[3],
    order = orders,
)
# show the count values
for p in ax.patches:
   ax.annotate('{:.1f}'.format(p.get_height()),
               (p.get_x()+0.25, p.get_height()+0.01),
               rotation = 90,
               horizontalalignment='center',
               verticalalignment='top',
              size=14
              )
# Set title
plt.title('Most 10 FlightNum a flight delay or cancellation', fontsize=16);
plt.xlabel('FlightNum', fontsize=16);
plt.ylabel('Count', fontsize=16);

# plot mean line
plt.axhline(ex_data.mean(), c='black')

plt.show();
```



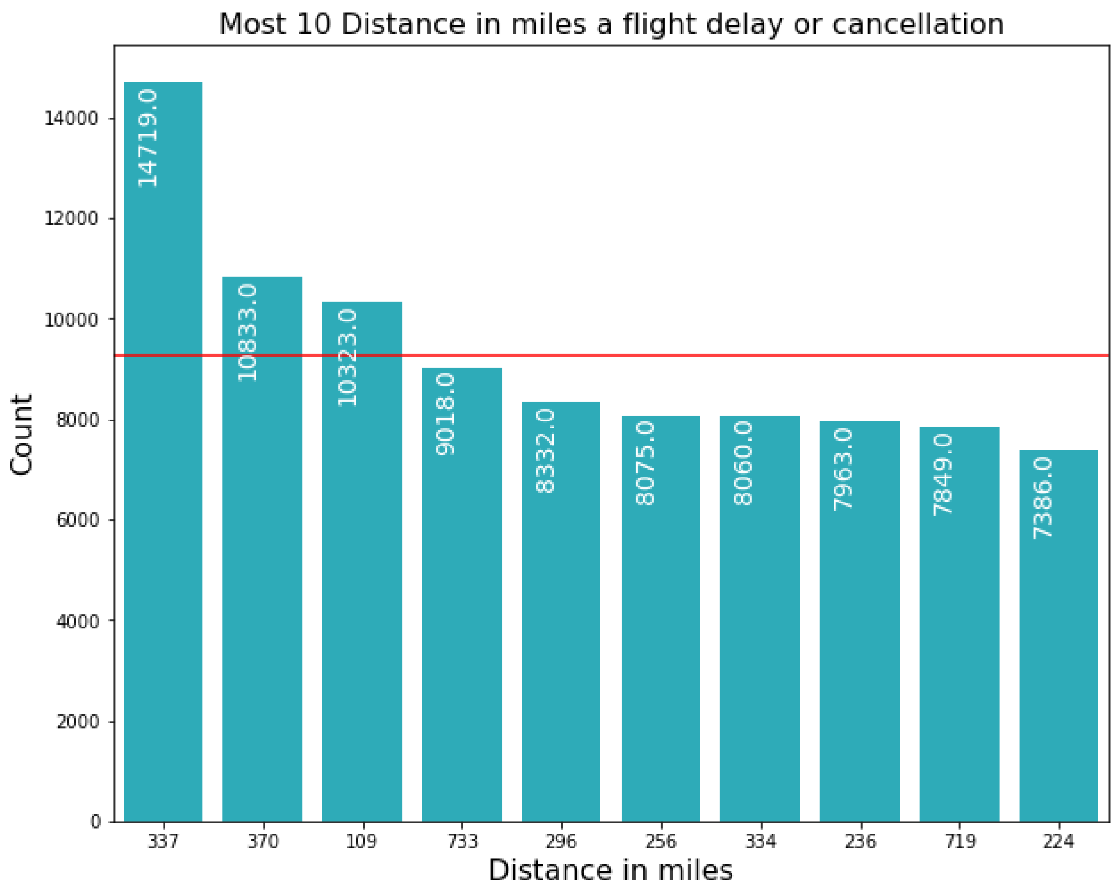
3.8 Most 10 FlightNum a flight delay or cancellation are:

Flight numbers 197, 481 and 70 are the most delayed flights

3.9 What is the most Distance in miles a flight delay or cancellation?

[27]:
```
# Set plot size
fig, ax = plt.subplots(figsize = [10,8])

# Set Plot Color
colors = sns.color_palette()

# Extract data
ex_data = df['Distance'].value_counts().head(10)

# Set order values
orders = ex_data.head(10).index

# Define plot
```

```
sns.countplot(
    x = 'Distance',
    data = df,
    color=colors[9],
    order = orders,
)

# show the count values
for p in ax.patches:
   ax.annotate('{:.1f}'.format(p.get_height()),
               (p.get_x()+0.25, p.get_height()+0.01),
               rotation = 90,
               horizontalalignment='center',
               verticalalignment='top',
               color='white',
              size=14
              )
# Set title
plt.title('Most 10 Distance in miles a flight delay or cancellation',␣
 ↪fontsize=16);

# Set labels
plt.xlabel('Distance in miles', fontsize=16);
plt.ylabel('Count', fontsize=16);

# plot mean line
plt.axhline(ex_data.mean(), c='red')

plt.show();
```



3.10 The most Distance in miles a flight delay or cancellation

Flights 337, 370, 109 are the most delayed or canceled

3.10.1 Discuss the distribution(s) of your variable(s) of interest. Were there any unusual points? Did you need to perform any transformations?

The data that was studied showed multiple types of distribution that correspond to the reality of canceled or delayed flights as it is expected to be according to the variables concerned

3.10.2 Of the features you investigated, were there any unusual distributions? Did you perform any operations on the data to tidy, adjust, or change the form of the data? If so, why did you do this?

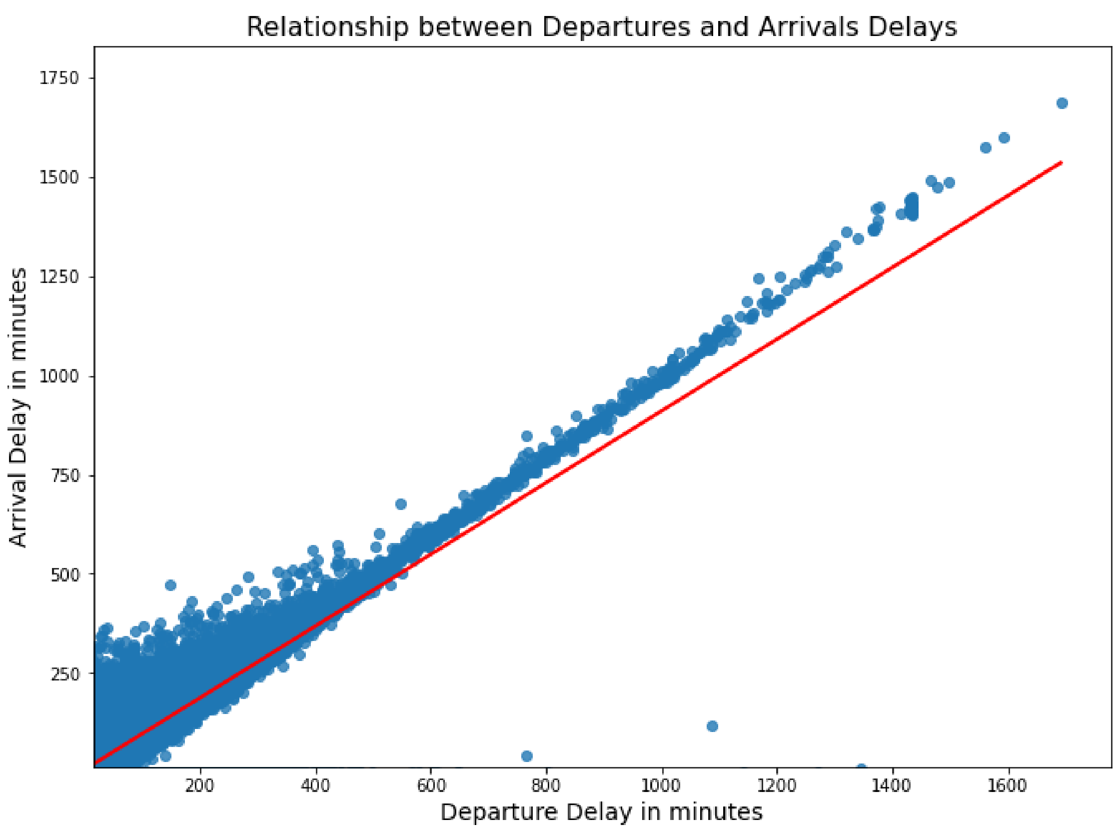
The data studied is a subset based only on delayed and canceled flights, so there were no unusual distributions or outliers, so no changes were needed

4 Bivariate Exploration

4.1 Is there any relationShip between DepDelay and ArrDelay?

https://www.adamsmith.haus/python/answers/how-to-find-the-correlation-between-two-pandas-dataframe-columns-in-python

[28]:
```
# set size of plot
f,ax = plt.subplots(figsize=(11,8));

sns.regplot(data=df, x='DepDelay', y='ArrDelay', line_kws={"color":"r"})
#Set Title
plt.title('Relationship between Departures and Arrivals Delays', fontsize=16);
#Set labels
plt.xlabel('Departure Delay in minutes', fontsize=14);
plt.ylabel('Arrival Delay in minutes', fontsize=14);

# Focus on the delays >=16
plt.xlim(15);
plt.ylim(15);

plt.show();
```

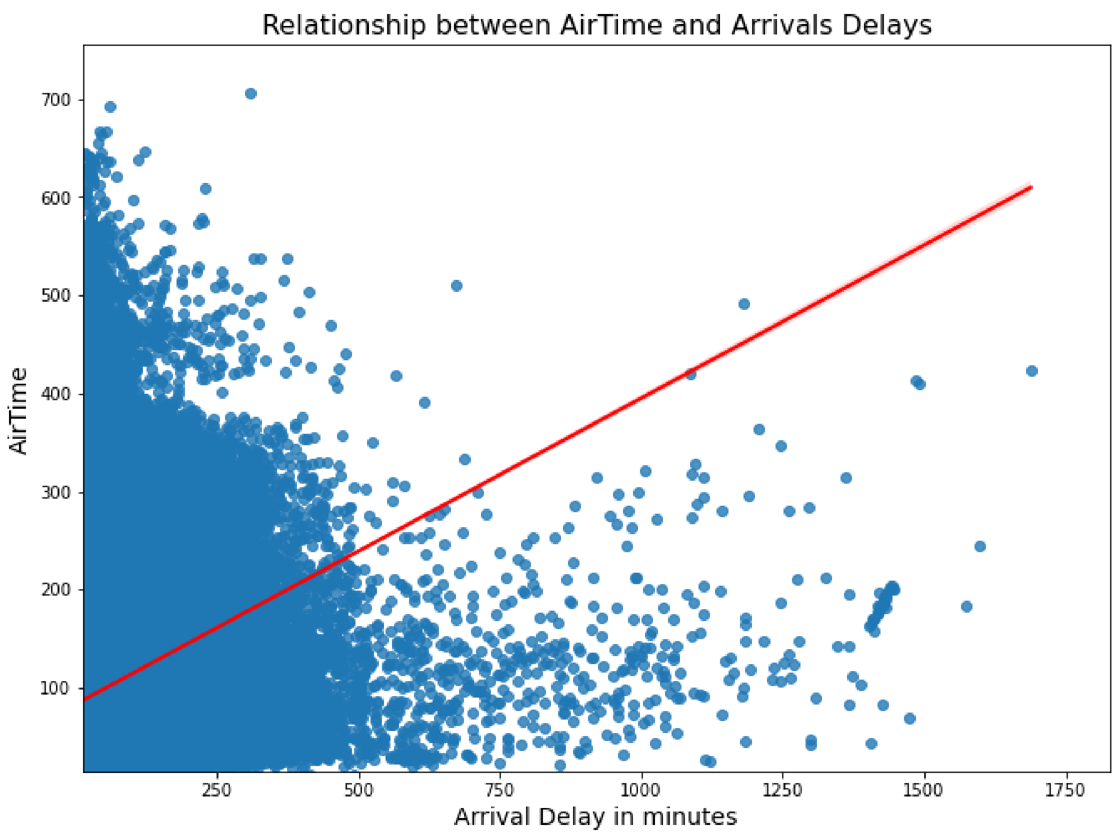


4.2 Relationship between Departure Delay and Arrival Delay is:

Very Strong positive relationship

4.3 What is the Relationship between Air Time and Arrival Delay?

[29]:
```
# set size of plot
f,ax = plt.subplots(figsize=(11, 8));

sns.regplot(data=df, x='ArrDelay', y='AirTime', line_kws={"color":"r"})
#Set Title
plt.title('Relationship between AirTime and Arrivals Delays', fontsize=16);
#Set labels
plt.xlabel('Arrival Delay in minutes', fontsize=14);
plt.ylabel('AirTime', fontsize=14);


# Focus on the delays >=16
plt.xlim(15);
plt.ylim(15);

plt.show();
```

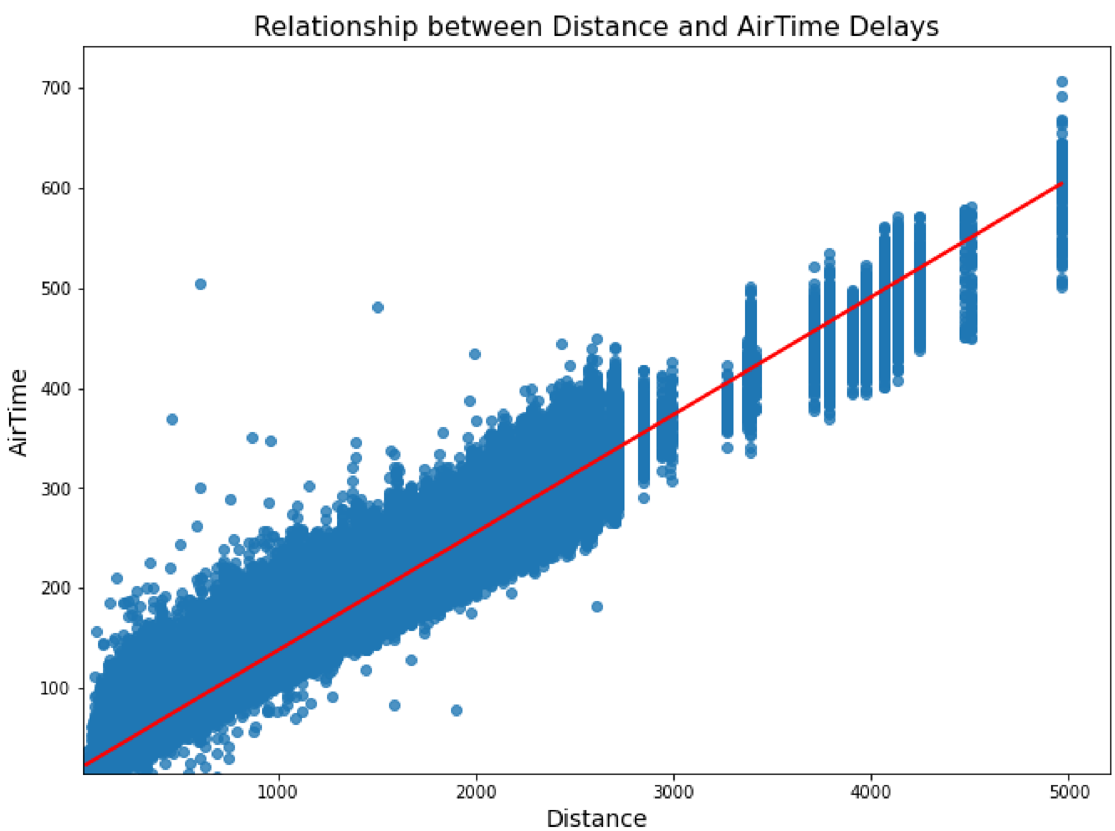


4.4 Relationship between Air Time and Arrival Delay is:

Very week positive

4.5 What is the Relationship between Distance and AirTime Delay?

[30]:
```
# set size of plot
f,ax = plt.subplots(figsize=(11, 8));
arr = df[(df['AirTime'] !=0)]
sns.regplot(data=arr, x='Distance', y='AirTime', line_kws={"color":"r"})
#Set Title
plt.title('Relationship between Distance and AirTime Delays', fontsize=16);
#Set labels
plt.xlabel('Distance', fontsize=14);
plt.ylabel('AirTime', fontsize=14);

# Focus on the delays >=16
plt.xlim(15);
plt.ylim(15);

plt.show();
```

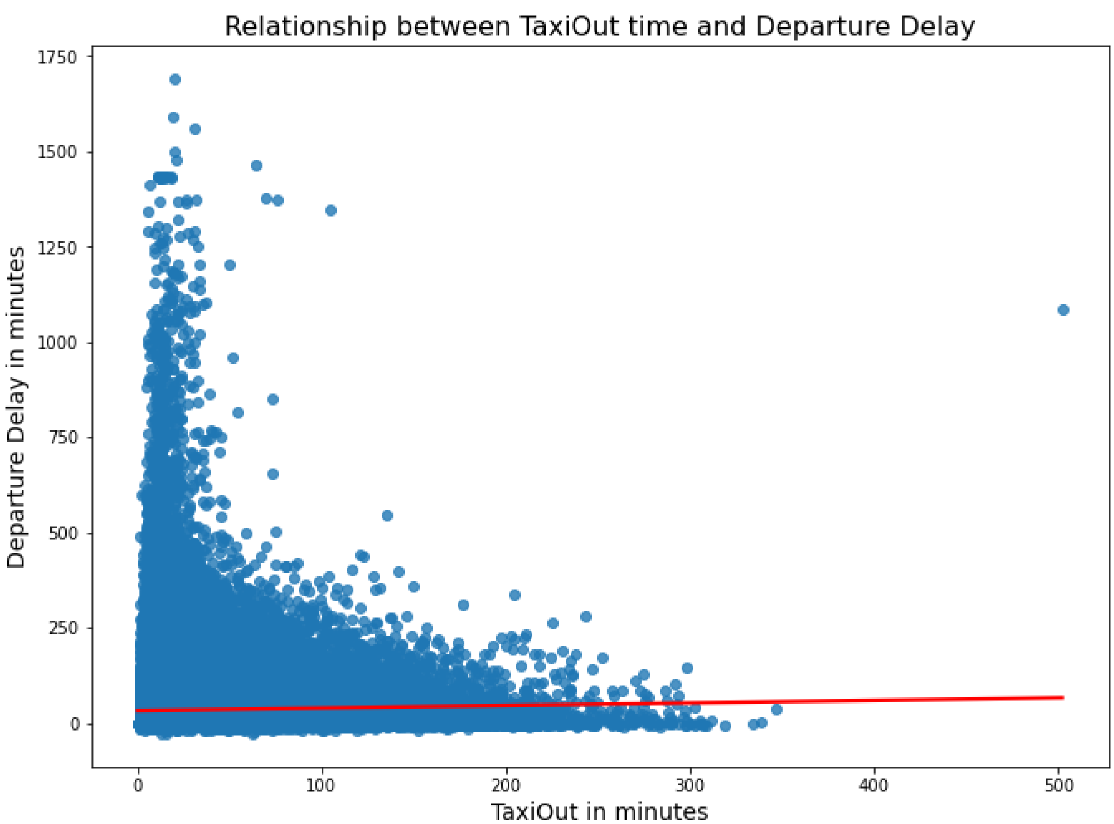


4.6 Relationship between Air Time and Distance is:

Very Strong positive relation

4.7 What is the Relationship between TaxiOut and Departure Delay?

[31]:
```
# Define plot

plt.figure(figsize=(11,8));
sns.regplot(data=df, x = 'TaxiOut',y = 'DepDelay', line_kws={'color': 'red'});

#set title and axis

plt.title('Relationship between TaxiOut time and Departure Delay', fontsize=16);
plt.xlabel('TaxiOut in minutes', fontsize=14);
plt.ylabel('Departure Delay in minutes', fontsize=14);

#show plot

plt.show();
```

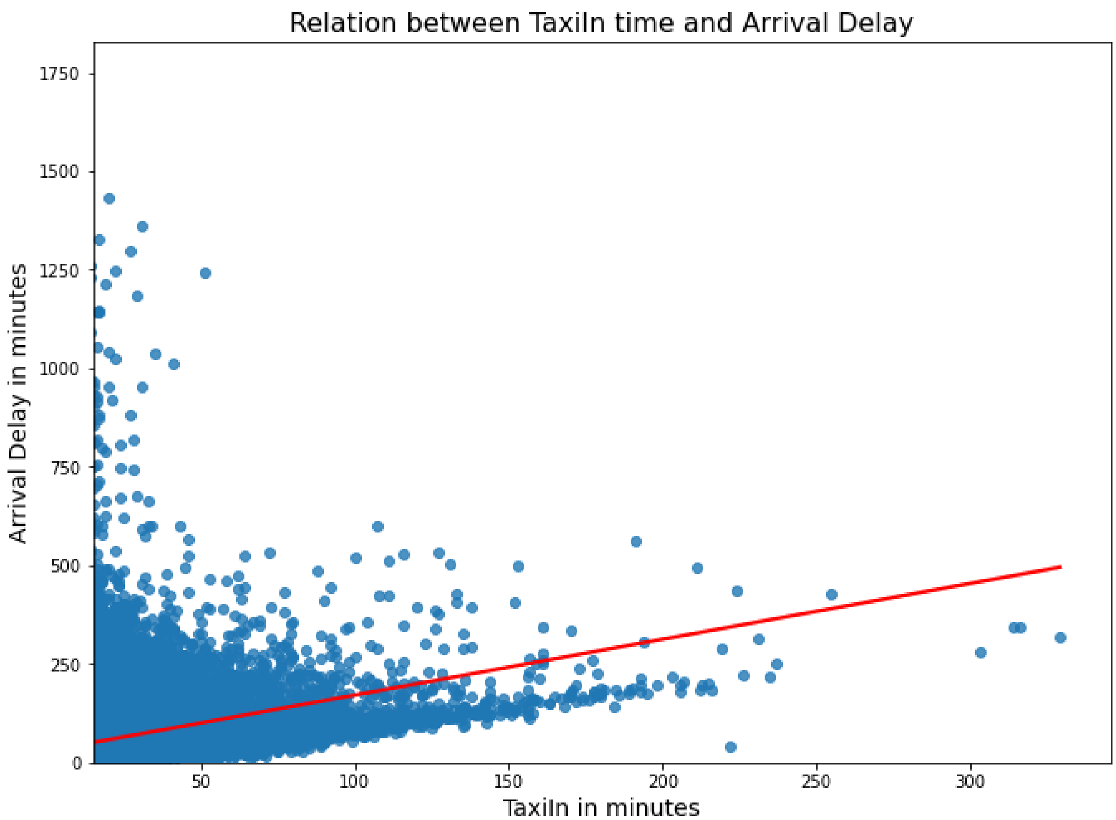


4.8 Relationship between TaxiOut and Departure Delay is:

Very week positive relation

4.9 What is the Relationship between TaxiIn and Arrival Delay?

[32]:
```
# Define plot

plt.figure(figsize=(11,8));
sns.regplot(data=df, x = 'TaxiIn',y = 'ArrDelay', line_kws={'color': 'red'});

#set title and axis

plt.title('Relation between TaxiIn time and Arrival Delay', fontsize=16);
plt.xlabel('TaxiIn in minutes', fontsize=14);
plt.ylabel('Arrival Delay in minutes', fontsize=14);

plt.ylim(0,);
plt.xlim(15,);

#show plot
plt.show();
```



4.10 Relationship between TaxiIn and Arrival Delay is:

Week positive relation

4.10.1 Talk about some of the relationships you observed in this part of the investigation. How did the feature(s) of interest vary with other features in the dataset?

It was noted that most of the relationships between the variables of interest are positive, and differ among themselves among:

the very strong relation , such as:

Relationship between Distance and AirTime Delays

Relationship between AirTime and Distance

Relationship between Departure Delay and Arrival Delay

Then Weak relation such as:

Relation between TaxiIn time and Arrival Delay

And very weak like:

Relationship between TaxiOut time and Departure Delay

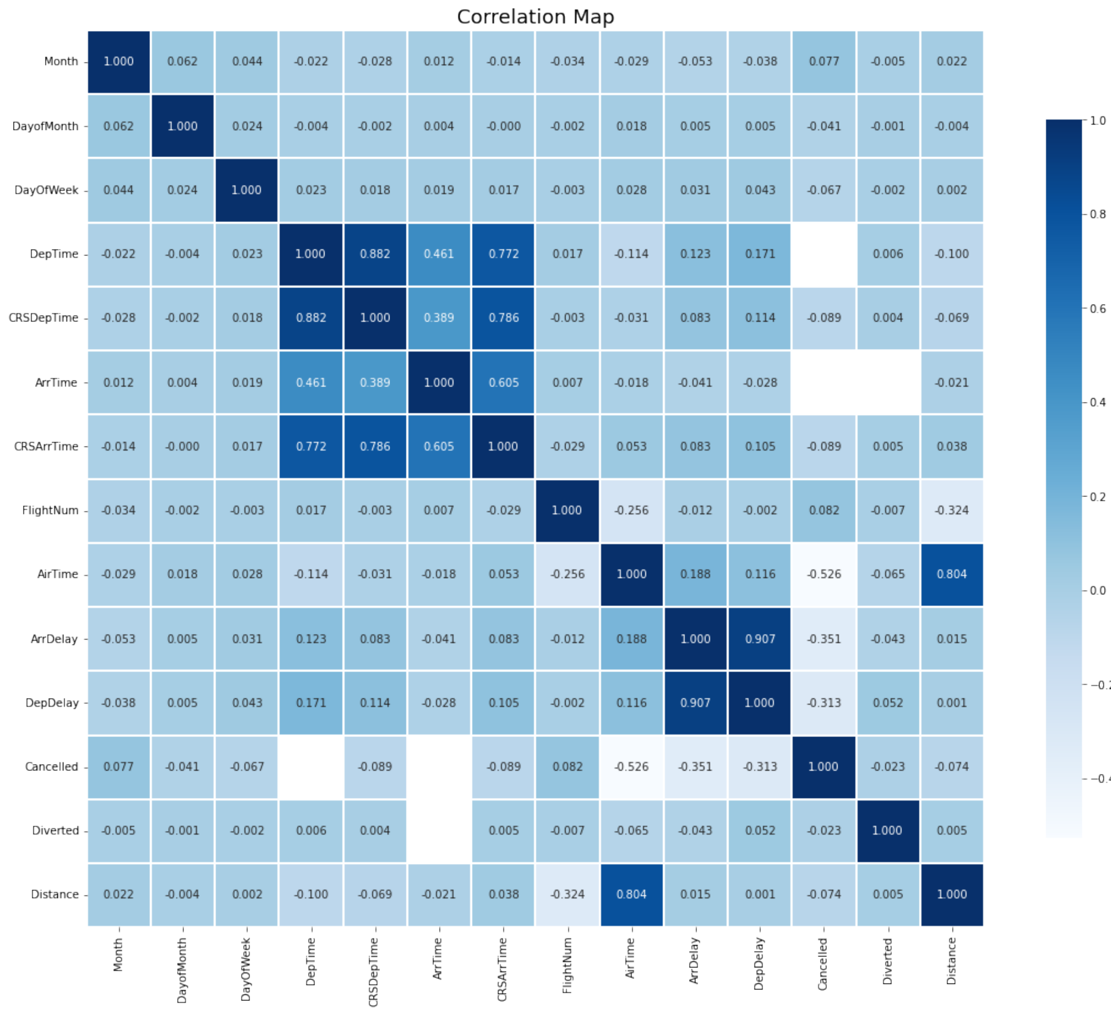
4.10.2 Did you observe any interesting relationships between the other features (not the main feature(s) of interest)?

In This dataset I did not observe any interesting relationships between other features

5 Multivariate Exploration

Create plots of three or more variables to investigate your data even further. Make sure that your investigations are justified, and follow from your work in the previous sections.

[33]:
```
# set size of plot
f,ax = plt.subplots(figsize=(20, 15));
# Define plot of all interesting variables
sns.heatmap(df[[
    'Month', 'DayofMonth', 'DayOfWeek', 'DepTime', 'CRSDepTime', 'ArrTime',
    'CRSArrTime', 'UniqueCarrier', 'FlightNum', 'TailNum','AirTime', 'ArrDelay',
    'DepDelay', 'Origin', 'Dest', 'Cancelled', 'Diverted', 'Distance'
]].corr(),
            cmap="Blues",square=True, annot=True, fmt= '.3f',ax=ax,
           linewidth=0.3, cbar_kws={"shrink": .8});
# set title
plt.title('Correlation Map', fontsize=18);
```



From the heatmap Most notable:

Very Strong positve relationship between:

ArrDelay and DepDelay with correlation coefficient = 0.907

DepTime and CRSDepTime with correlation coefficient = 0.882

Strong positve relationship between:

'CRSArrTime' and 'CRSDepTime' with correlation coefficient = 0.786

'CRSArrTime' and 'DepTime' with correlation coefficient = 0.772

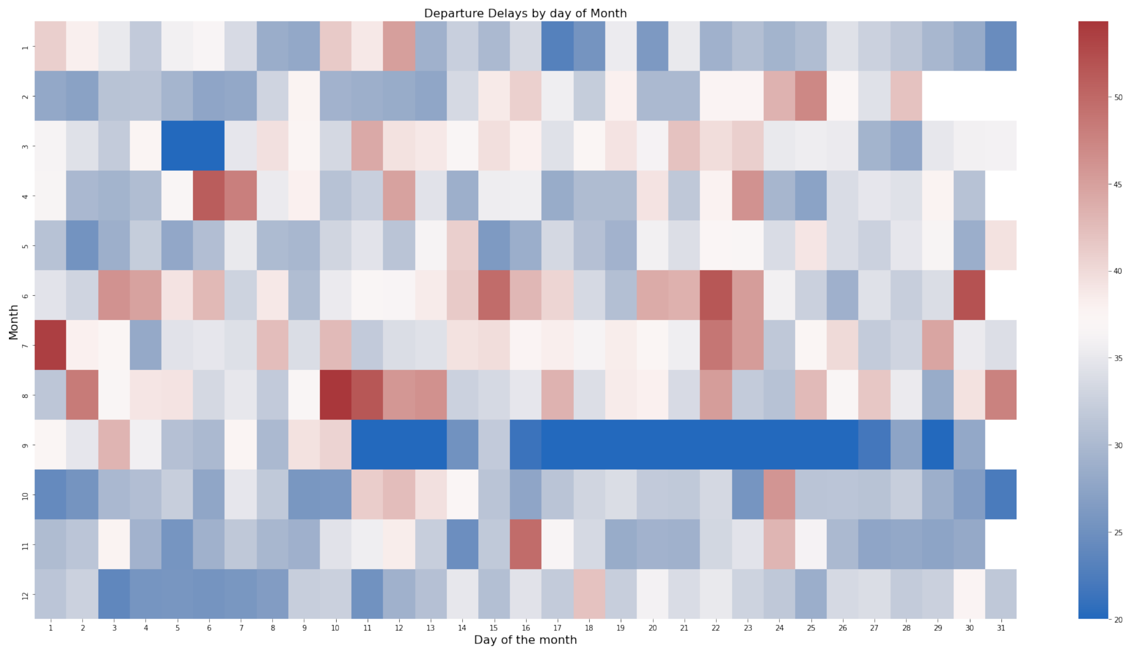
'CRSArrTime' and 'AirTime' with correlation coefficient = 0.605

Moderate positve relationship between:

'DepTime' and 'ArrTime' with correlation coefficient = 0.461

`<p> P.S: We can see also negative relationships and very week relations</p>`

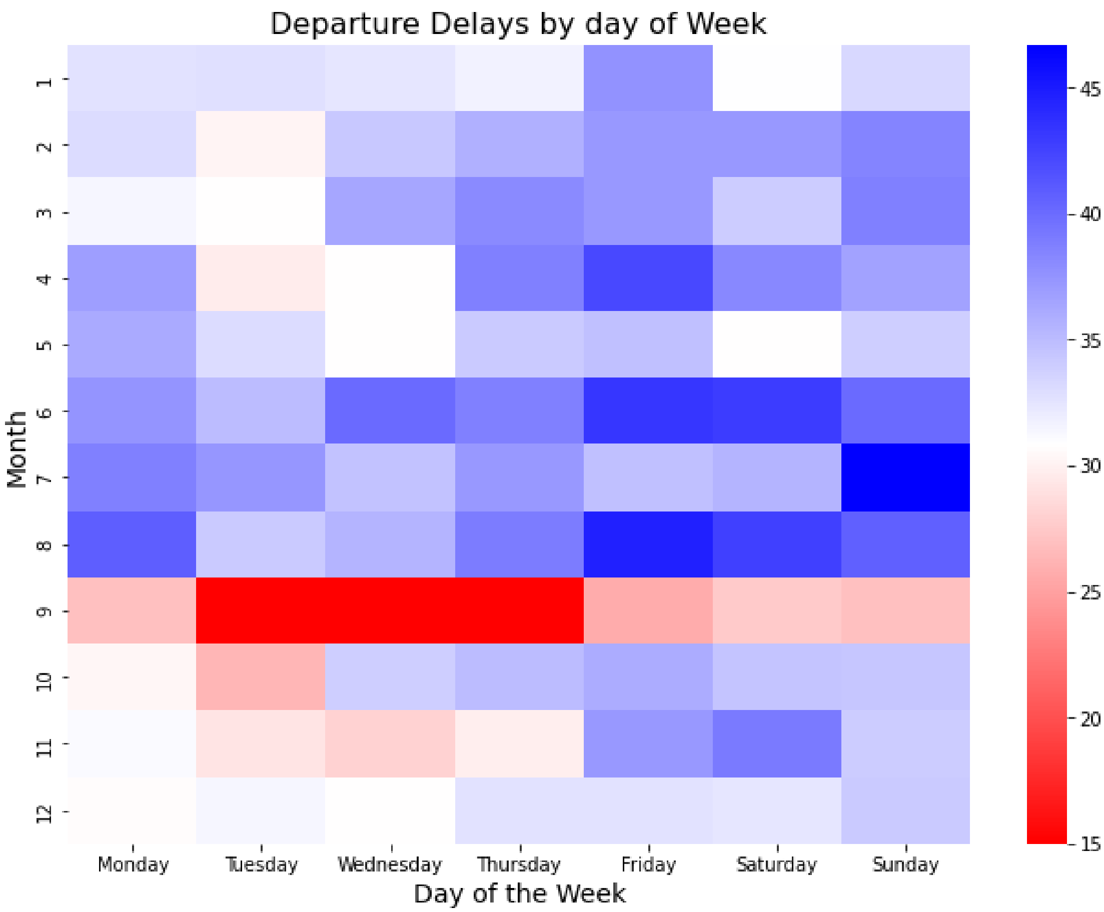
[34]:
```
#pivot variables of interest
pl = df.pivot_table(index='Month',columns='DayofMonth', values='DepDelay',␣
 ↪aggfunc='mean')

#generate plot
plt.figure(figsize=(30,15));
sns.heatmap(pl,cmap='vlag', vmin=20);

#set title and axis

plt.title('Departure Delays by day of Month', fontsize=16);
plt.xlabel('Day of the month', fontsize=16);
plt.ylabel('Month', fontsize=16);
```



July 1st and August 10th had the highest Average of Departure delays

[35]:
```
#pivot variables of interest
pl = df.pivot_table(index='Month',columns='DayOfWeek', values='DepDelay',␣
 ↪aggfunc='mean')

#generate plot
plt.figure(figsize=(11,8));
g = sns.heatmap(pl, cmap='bwr_r', vmin=15);

#set title and axis
```

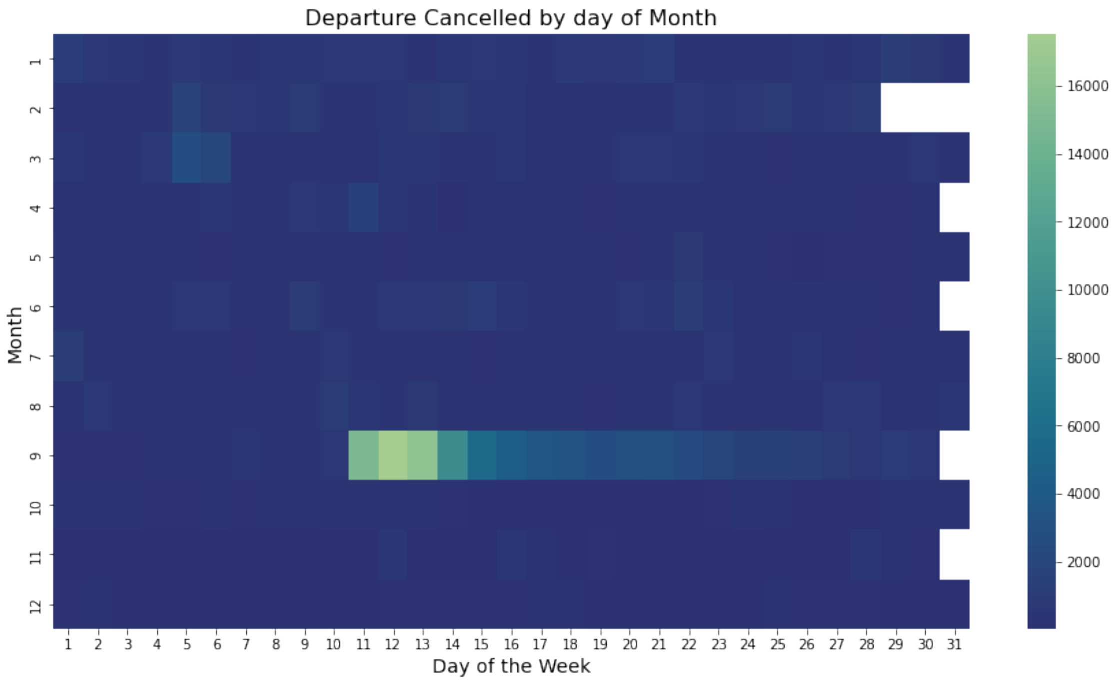
```
week_day = ['Monday','Tuesday','Wednesday','Thursday','Friday',␣
↪'Saturday','Sunday']
g.set_xticklabels(week_day);
#SET Title
plt.title('Departure Delays by day of Week', fontsize=16);
plt.xlabel('Day of the Week', fontsize=14);
plt.ylabel('Month', fontsize=14);
```



Sunday in july is the day that had the highest Average of Departure delays

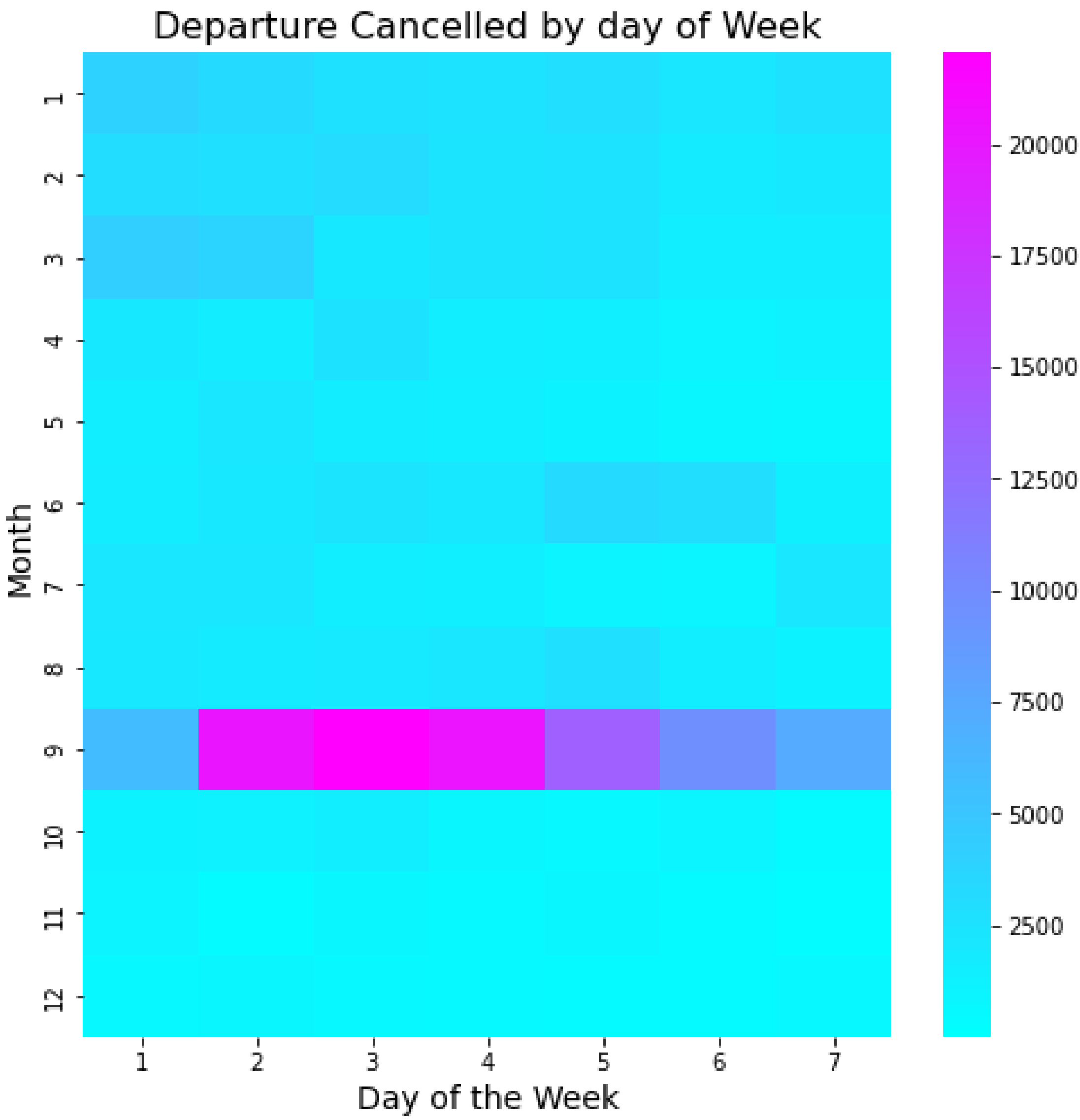
[36]:
```
#pivot variables of interest
pl = df.pivot_table(index='Month',columns='DayofMonth', values='Cancelled',␣
↪aggfunc='sum')

#generate plot
plt.figure(figsize=(15,8));
sns.heatmap(pl,cmap='crest_r', vmin=15);
```

```
#set title and axis

plt.title('Departure Cancelled by day of Month', fontsize=16);
plt.xlabel('Day of the Week', fontsize=14);
plt.ylabel('Month', fontsize=14);
```



12 September the day that have the most cancellation flights

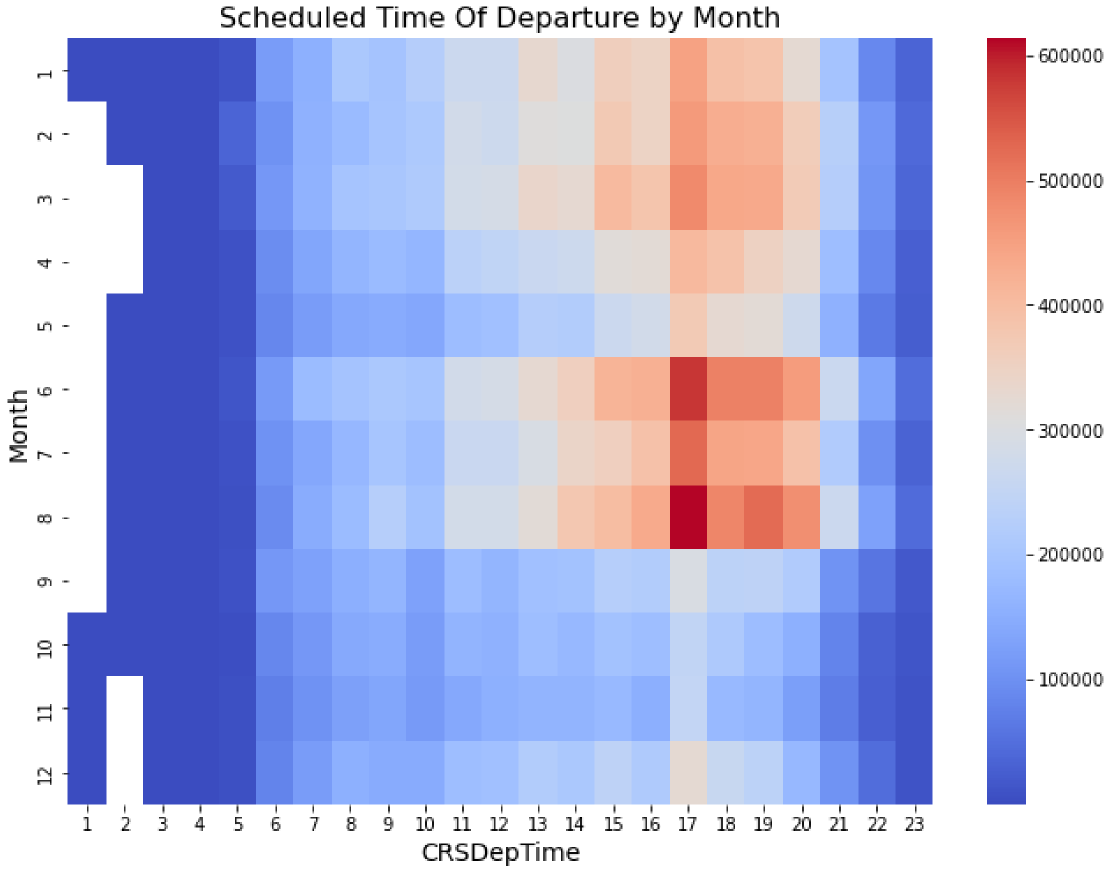
[37]:
```
#pivot variables of interest
pl = df.pivot_table(index='Month',columns='DayOfWeek', values='Cancelled',␣
↪aggfunc='sum')

#generate plot
plt.figure(figsize=(8,8));
sns.heatmap(pl, cmap='cool', vmin=15);

#set title and axis

plt.title('Departure Cancelled by day of Week', fontsize=16);
plt.xlabel('Day of the Week', fontsize=14);
plt.ylabel('Month', fontsize=14);
```



Wednesday in September had the most cancellated flghts

[38]:
```
#pivot variables of interest
pl = df.pivot_table(index = 'Month', columns = 'CRSDepTime', values='DepDelay',␣
↪aggfunc='sum')

#generate plot
plt.figure(figsize=(11,8));
sns.heatmap(pl, cmap='coolwarm', vmin=15);

#set title and axis
```

```
plt.title('Scheduled Time Of Departure by Month', fontsize=16);
plt.xlabel('CRSDepTime', fontsize=14);
plt.ylabel('Month', fontsize=14);
```



17:00 (5PM) to 20:00 (8PM) in June, July , August had the most delay time

5.0.1 Talk about some of the relationships you observed in this part of the investigation. Were there features that strengthened each other in terms of looking at your feature(s) of interest?

- Wednesday September 12th was the highest day for flights to be canceled
- Sunday in july 1st was the day that had the highest Average of Departure delays

5.0.2 Were there any interesting or surprising interactions between features?

The 9/11 attack had a huge impact in being the only month with a high rate of canceled flights

6 Conclusions:

The data used for this study from Year 2001 included only delayed and canceled flights The data was analyzed in 3 stages

6.0.1 1- Univariate Exploration

The results of this stage:

The most top 10 Delays or cancelation are: - **By UniqueCarrier :** United Airlines, Inc.**(UA)**, followed by Southwest Airlines Co.**(WN)**, then Delta Air Lines, Inc.**(DL)**, had the most delays - **By Origin :** Chicago O'Hare International Airport **(ORD)** ,Dallas/Ft Worth Intl **(DFW)**, Atlanta Hartsfield-Jackson Int - **By Destination :** Chicago O'Hare International Airport **(ORD)**, Dallas/Ft Worth Intl **(DFW)**, Los Angeles **(LAX)**

- Most days of the week with a flight delay or cancellation are: **Friday, Thursday and Wednesday**
- Most Month with a flight delay or cancellation is:**September**
- Highest days of delayed or canceled flights in September: **Days 11, 12, 13, 14**
- Scheduled Departure Time that have the most delays or cancelation are :**Between 15:00 to 17:00 (3-5 pm) Maximum delay or cancellation is at 17:00**
- Most delay or cancellation flight by Air Time:**Airtime on short flights of 50 minutes or less has the greatest cancelled flights**
- Most 10 TailNum a flight delay or cancellation are:**Plane -N916D has 803 delays and cancellations followed by plane -N914 and plane -N910**
- Most 10 FlightNum a flight delay or cancellation are:**Flight numbers 197, 481 and 70 are the most delayed flights**
- The most Distance in miles a flight delay or cancellation:**Flights 337, 370, 109 are the most delayed or canceled**

6.0.2 Bivariate Exploration

The results of this stage:

- Relationship between Departure Delay and Arrival Delay is:**Very Strong positive relationship**
- Relationship between Air Time and Arrival Delay is:**Very week positive**
- Relationship between Air Time and Distance is:**Very Strong positive relation**
- Relationship between TaxiIn and Arrival Delay is:**Week positive relation**

6.0.3 Multivariate Exploration

The results of this stage:

- From the heatmap there are: . Very Strong positve relationship between:

- ArrDelay and DepDelay with correlation coefficient = 0.907
- DepTime and CRSDepTime with correlation coefficient = 0.882 . Strong positve relationship between:
- 'CRSArrTime' and 'CRSDepTime' with correlation coefficient = 0.786
- 'CRSArrTime' and 'DepTime' with correlation coefficient = 0.772
- 'CRSArrTime' and 'AirTime' with correlation coefficient = 0.605 . Moderate positve relationship between:
- 'DepTime' and 'ArrTime' with correlation coefficient = 0.461
- July 1st and August 10th had the highest Average of Departure delays
- Sunday in july is the day that had the highest Average of Departure delays
- 12 September the day that have the most cancellation flights
- Wednesday in September had the most cancellated flights
- 17:00 (5PM) to 20:00 (8PM) in June, July , August had the most delay time