In [23]:

```
# Generic inputs for most ML tasks
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
# This is new
from sklearn.linear_model import LogisticRegression
from sklearn.linear_model import Ridge
from sklearn.linear_model import Lasso
from sklearn.ensemble import RandomForestRegressor

pd.options.display.float_format = '{:,.2f}'.format

# setup interactive notebook mode
from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = "all"

from IPython.display import display, HTML
```

Fetching flight data

In [24]:

```
pred_data = pd.read_csv('pred_data\CIS_662 _FINAL_Predictions.csv')
```

In [25]:

```
pred_data.head()
```

Out[25]:

	DATE	DAY	FLIGHT NUMBER	ORIGIN	DEPARTURE TIME	ARRIVAL TIME	ARRIVAL STATUS	ARRIVAL STATUS_Prev_flight_early	ARRIVAL STATUS_Prev_flight_ontime
0	4/19/2024	FRIDAY	UA 1400	ORD	6:52 PM	9:47 PM	NaN	NaN	NaN
1	4/19/2024	FRIDAY	AA 3402	ORD	7:59 PM	10:52 PM	NaN	NaN	NaN
2	4/19/2024	FRIDAY	B6 116	JFK	1:34 PM	2:51 PM	NaN	NaN	NaN
3	4/19/2024	FRIDAY	DL 5182	JFK	2:55 PM	4:21 PM	NaN	NaN	NaN
4	4/19/2024	FRIDAY	WN 5285	MCO	11:35 AM	2:20 PM	NaN	NaN	NaN

In [26]:
```
pred_data.dtypes
```

Out[26]:
```
DATE                                  object
DAY                                   object
FLIGHT NUMBER                         object
ORIGIN                                object
DEPARTURE TIME                        object
ARRIVAL TIME                          object
ARRIVAL STATUS                       float64
ARRIVAL STATUS_Prev_flight_early     float64
ARRIVAL STATUS_Prev_flight_ontime    float64
ARRIVAL STATUS_Prev_flight_late      float64
dep_order                             object
dtype: object
```

In [27]:

```
pred_data['DATE'] = pd.to_datetime( pred_data['DATE'],format ="%m/%d/%Y")
pred_data['DAY'] = pred_data['DATE'].dt.day_of_week.astype('object')
pred_data['Carrier_Code'] = pred_data['FLIGHT NUMBER'].str[0:2].astype('object')
pred_data['DEPARTURE TIME'] = pd.to_datetime(pred_data['DEPARTURE TIME'], format = '%I:%M %p')
pred_data['dep_hour']= pred_data['DEPARTURE TIME'].dt.hour.astype('object')
pred_data['dep_min'] = pred_data['DEPARTURE TIME'].dt.minute.astype('object')
pred_data['ARRIVAL TIME'] = pd.to_datetime(pred_data['ARRIVAL TIME'], format = '%I:%M %p')
pred_data['arr_hour']= pred_data['ARRIVAL TIME'].dt.hour.astype('object')
pred_data['arr_min'] = pred_data['ARRIVAL TIME'].dt.minute.astype('object')
pred_data.rename(columns = {'ORIGIN' : 'Origin_Airport'},inplace =True)
pred_data.head()
```

Out[27]:

	DATE	DAY	FLIGHT NUMBER	Origin_Airport	DEPARTURE TIME	ARRIVAL TIME	ARRIVAL STATUS	ARRIVAL STATUS_Prev_flight_early	ARRIVAL STATUS_Prev_flight_ontime
0	2024-04-19	4	UA 1400	ORD	1900-01-01 18:52:00	1900-01-01 21:47:00	NaN	NaN	NaN
1	2024-04-19	4	AA 3402	ORD	1900-01-01 19:59:00	1900-01-01 22:52:00	NaN	NaN	NaN
2	2024-04-19	4	B6 116	JFK	1900-01-01 13:34:00	1900-01-01 14:51:00	NaN	NaN	NaN
3	2024-04-19	4	DL 5182	JFK	1900-01-01 14:55:00	1900-01-01 16:21:00	NaN	NaN	NaN
4	2024-04-19	4	WN 5285	MCO	1900-01-01 11:35:00	1900-01-01 14:20:00	NaN	NaN	NaN

In [28]:
```
pred_data.dtypes
```

Out[28]:
```
DATE                                   datetime64[ns]
DAY                                            object
FLIGHT NUMBER                                  object
Origin_Airport                                 object
DEPARTURE TIME                         datetime64[ns]
ARRIVAL TIME                           datetime64[ns]
ARRIVAL STATUS                                float64
ARRIVAL STATUS_Prev_flight_early              float64
ARRIVAL STATUS_Prev_flight_ontime             float64
ARRIVAL STATUS_Prev_flight_late               float64
dep_order                                      object
Carrier_Code                                   object
dep_hour                                       object
dep_min                                        object
arr_hour                                       object
arr_min                                        object
dtype: object
```

In [30]:
```
pred_data.head(30)
```

Out[30]:

	DATE	DAY	FLIGHT NUMBER	Origin_Airport	DEPARTURE TIME	ARRIVAL TIME	ARRIVAL STATUS	ARRIVAL STATUS_Prev_flight_early	ARRIVAL STATUS_Prev_flight_ontime
0	2024-04-19	4	UA 1400	ORD	1900-01-01 18:52:00	1900-01-01 21:47:00	NaN	NaN	NaN
1	2024-04-19	4	AA 3402	ORD	1900-01-01 19:59:00	1900-01-01 22:52:00	NaN	NaN	NaN
2	2024-04-19	4	B6 116	JFK	1900-01-01 13:34:00	1900-01-01 14:51:00	NaN	NaN	NaN
3	2024-04-19	4	DL 5182	JFK	1900-01-01 14:55:00	1900-01-01 16:21:00	NaN	NaN	NaN
4	2024-04-19	4	WN 5285	MCO	1900-01-01 11:35:00	1900-01-01 14:20:00	NaN	NaN	NaN
5	2024-04-19	4	B6 656	MCO	1900-01-01 13:35:00	1900-01-01 16:25:00	NaN	NaN	NaN
6	2024-04-20	5	UA 1400	ORD	1900-01-01 18:52:00	1900-01-01 21:47:00	NaN	NaN	NaN
7	2024-04-20	5	AA 3402	ORD	1900-01-01 19:59:00	1900-01-01 22:52:00	NaN	NaN	NaN
8	2024-04-20	5	B6 116	JFK	1900-01-01 13:25:00	1900-01-01 14:41:00	NaN	NaN	NaN
9	2024-04-20	5	DL 5182	JFK	1900-01-01 14:55:00	1900-01-01 16:21:00	NaN	NaN	NaN
10	2024-04-20	5	B6 656	MCO	1900-01-01 13:35:00	1900-01-01 16:25:00	NaN	NaN	NaN
11	2024-04-21	6	UA 1400	ORD	1900-01-01 18:52:00	1900-01-01 21:47:00	NaN	NaN	NaN

	DATE	DAY	FLIGHT NUMBER	Origin_Airport	DEPARTURE TIME	ARRIVAL TIME	ARRIVAL STATUS	ARRIVAL STATUS_Prev_flight_early	ARRIVAL STATUS_Prev_flight_ontime
12	2024-04-21	6	AA 3402	ORD	1900-01-01 19:59:00	1900-01-01 22:52:00	NaN	NaN	NaN
13	2024-04-21	6	B6 116	JFK	1900-01-01 13:35:00	1900-01-01 14:51:00	NaN	NaN	NaN
14	2024-04-21	6	DL 5182	JFK	1900-01-01 14:55:00	1900-01-01 16:21:00	NaN	NaN	NaN
15	2024-04-21	6	WN 5285	MCO	1900-01-01 11:05:00	1900-01-01 13:50:00	NaN	NaN	NaN
16	2024-04-21	6	B6 656	MCO	1900-01-01 13:35:00	1900-01-01 16:25:00	NaN	NaN	NaN
17	2024-04-22	0	UA 1400	ORD	1900-01-01 18:52:00	1900-01-01 21:47:00	NaN	NaN	NaN
18	2024-04-22	0	AA 3402	ORD	1900-01-01 19:59:00	1900-01-01 22:52:00	NaN	NaN	NaN
19	2024-04-22	0	B6 116	JFK	1900-01-01 13:35:00	1900-01-01 14:51:00	NaN	NaN	NaN
20	2024-04-22	0	DL 5182	JFK	1900-01-01 14:55:00	1900-01-01 16:21:00	NaN	NaN	NaN
21	2024-04-22	0	WN 5285	MCO	1900-01-01 11:35:00	1900-01-01 14:20:00	NaN	NaN	NaN
22	2024-04-22	0	B6 656	MCO	1900-01-01 13:34:00	1900-01-01 16:25:00	NaN	NaN	NaN

In [31]:

```
pred_data['dep_minutes'] = 0
pred_data['dep_minutes'] = pred_data['dep_minutes'].astype('object')
pred_data['dep_hours'] = pred_data['dep_hour'].astype('object')
pred_data['arr_minutes'] = 0
pred_data['arr_minutes'] = pred_data['arr_minutes'].astype('object')
pred_data['arr_hours'] = pred_data['arr_hour'].astype('object')
pred_data.rename(columns ={'DATE':'Date'},inplace=True)
pred_data.head()
```

Out[31]:

	Date	DAY	FLIGHT NUMBER	Origin_Airport	DEPARTURE TIME	ARRIVAL TIME	ARRIVAL STATUS	ARRIVAL STATUS_Prev_flight_early	ARRIVAL STATUS_Prev_flight_ontime
0	2024-04-19	4	UA 1400	ORD	1900-01-01 18:52:00	1900-01-01 21:47:00	NaN	NaN	NaN
1	2024-04-19	4	AA 3402	ORD	1900-01-01 19:59:00	1900-01-01 22:52:00	NaN	NaN	NaN
2	2024-04-19	4	B6 116	JFK	1900-01-01 13:34:00	1900-01-01 14:51:00	NaN	NaN	NaN
3	2024-04-19	4	DL 5182	JFK	1900-01-01 14:55:00	1900-01-01 16:21:00	NaN	NaN	NaN
4	2024-04-19	4	WN 5285	MCO	1900-01-01 11:35:00	1900-01-01 14:20:00	NaN	NaN	NaN

In [32]:
```
pred_data['Date'] = pred_data['Date'].dt.strftime('%m/%d/%Y')
pred_data.dtypes
len(pred_data)
pred_data.head(30)
```

Out[32]:
```
Date                                          object
DAY                                           object
FLIGHT NUMBER                                 object
Origin_Airport                                object
DEPARTURE TIME                        datetime64[ns]
ARRIVAL TIME                          datetime64[ns]
ARRIVAL STATUS                               float64
ARRIVAL STATUS_Prev_flight_early             float64
ARRIVAL STATUS_Prev_flight_ontime            float64
ARRIVAL STATUS_Prev_flight_late              float64
dep_order                                     object
Carrier_Code                                  object
dep_hour                                      object
dep_min                                       object
arr_hour                                      object
arr_min                                       object
dep_minutes                                   object
dep_hours                                     object
arr_minutes                                   object
arr_hours                                     object
dtype: object
```

Out[32]:
```
23
```

Out[32]:

	Date	DAY	FLIGHT NUMBER	Origin_Airport	DEPARTURE TIME	ARRIVAL TIME	ARRIVAL STATUS	ARRIVAL STATUS_Prev_flight_early	ARRI STATUS_Prev_flight_ont
0	04/19/2024	4	UA 1400	ORD	1900-01-01 18:52:00	1900-01-01 21:47:00	NaN	NaN	N
1	04/19/2024	4	AA 3402	ORD	1900-01-01 19:59:00	1900-01-01 22:52:00	NaN	NaN	N
2	04/19/2024	4	B6 116	JFK	1900-01-01 13:34:00	1900-01-01 14:51:00	NaN	NaN	N
3	04/19/2024	4	DL 5182	JFK	1900-01-01 14:55:00	1900-01-01 16:21:00	NaN	NaN	N
4	04/19/2024	4	WN 5285	MCO	1900-01-01 11:35:00	1900-01-01 14:20:00	NaN	NaN	N
5	04/19/2024	4	B6 656	MCO	1900-01-01 13:35:00	1900-01-01 16:25:00	NaN	NaN	N
6	04/20/2024	5	UA 1400	ORD	1900-01-01 18:52:00	1900-01-01 21:47:00	NaN	NaN	N
7	04/20/2024	5	AA 3402	ORD	1900-01-01 19:59:00	1900-01-01 22:52:00	NaN	NaN	N
8	04/20/2024	5	B6 116	JFK	1900-01-01 13:25:00	1900-01-01 14:41:00	NaN	NaN	N
9	04/20/2024	5	DL 5182	JFK	1900-01-01 14:55:00	1900-01-01 16:21:00	NaN	NaN	N
10	04/20/2024	5	B6 656	MCO	1900-01-01 13:35:00	1900-01-01 16:25:00	NaN	NaN	N
11	04/21/2024	6	UA 1400	ORD	1900-01-01 18:52:00	1900-01-01 21:47:00	NaN	NaN	N

	Date	DAY	FLIGHT NUMBER	Origin_Airport	DEPARTURE TIME	ARRIVAL TIME	ARRIVAL STATUS	ARRIVAL STATUS_Prev_flight_early	ARRI STATUS_Prev_flight_ont
12	04/21/2024	6	AA 3402	ORD	1900-01-01 19:59:00	1900-01-01 22:52:00	NaN	NaN	N
13	04/21/2024	6	B6 116	JFK	1900-01-01 13:35:00	1900-01-01 14:51:00	NaN	NaN	N
14	04/21/2024	6	DL 5182	JFK	1900-01-01 14:55:00	1900-01-01 16:21:00	NaN	NaN	N
15	04/21/2024	6	WN 5285	MCO	1900-01-01 11:05:00	1900-01-01 13:50:00	NaN	NaN	N
16	04/21/2024	6	B6 656	MCO	1900-01-01 13:35:00	1900-01-01 16:25:00	NaN	NaN	N
17	04/22/2024	0	UA 1400	ORD	1900-01-01 18:52:00	1900-01-01 21:47:00	NaN	NaN	N
18	04/22/2024	0	AA 3402	ORD	1900-01-01 19:59:00	1900-01-01 22:52:00	NaN	NaN	N
19	04/22/2024	0	B6 116	JFK	1900-01-01 13:35:00	1900-01-01 14:51:00	NaN	NaN	N
20	04/22/2024	0	DL 5182	JFK	1900-01-01 14:55:00	1900-01-01 16:21:00	NaN	NaN	N
21	04/22/2024	0	WN 5285	MCO	1900-01-01 11:35:00	1900-01-01 14:20:00	NaN	NaN	N
22	04/22/2024	0	B6 656	MCO	1900-01-01 13:34:00	1900-01-01 16:25:00	NaN	NaN	N

Fetching weather data and merging with flight data

In [33]:

```
# Read and process weather data files for each airport
jfk_weather_data = pd.read_csv('weather_data/JFK_weather_data_forecast_processed.csv')
syr_weather_data = pd.read_csv('weather_data/SYR_weather_data_forecast_processed.csv')
ord_weather_data = pd.read_csv('weather_data/ORD_weather_data_forecast_processed.csv')
mco_weather_data = pd.read_csv('weather_data/MCO_weather_data_forecast_processed.csv')

# Combine weather data for all airports
weather_dfs = [jfk_weather_data, ord_weather_data, mco_weather_data]
weather_data = pd.concat(weather_dfs, axis=0)
#weather_data['dep_minutes'] = weather_data['dep_minutes'].astype('object')
weather_data['dep_hours'] = weather_data['dep_hours'].astype('object')
#syr_weather_data['arr_minutes'] = syr_weather_data['arr_minutes'].astype('object')
syr_weather_data['arr_hours'] = syr_weather_data['arr_hours'].astype('object')
weather_data.head(50)
len(weather_data)
len(syr_weather_data)

# Define merging logic based on airport code
pred_data = pd.merge(pred_data, syr_weather_data, how='left', on=['Date', 'arr_hours'])
len(pred_data)
pred_data.to_csv("first.csv")

pred_data = pd.merge(pred_data, weather_data, how='left', on=['Origin_Airport', 'Date', 'dep_hours'])
pred_data.to_csv("sec.csv")
len(pred_data)


#weather_data.head()
```

Out[33]:

	dep_clouds	dep_clouds_hi	dep_clouds_low	dep_clouds_mid	dep_dewpt	dep_ozone	dep_pop	dep_precip	dep_pres	dep_rh
0	92	0	100	0	4.30	382.50	0	0.00	1,019.50	74
1	95	0	100	0	4.50	375.80	0	0.00	1,020.50	77
2	95	0	100	0	3.40	369.50	0	0.00	1,021.00	72
3	83	48	100	0	3.80	365.30	0	0.00	1,020.50	75
4	76	87	99	0	3.80	362.50	0	0.00	1,020.50	74
5	80	100	100	0	3.30	354.80	0	0.00	1,021.00	72
6	73	100	98	0	3.70	354.80	0	0.00	1,020.00	74
7	67	54	34	10	3.60	351.50	0	0.00	1,020.00	75
8	61	35	12	100	3.80	347.50	0	0.00	1,020.50	76
9	57	1	5	73	3.60	346.00	0	0.00	1,020.00	76
10	71	16	14	0	3.60	344.80	0	0.00	1,020.00	77
11	73	100	18	0	3.30	343.00	0	0.00	1,021.50	76
12	63	0	17	0	3.40	343.00	0	0.00	1,022.00	76
13	64	0	27	0	3.60	341.50	0	0.00	1,022.00	75
14	65	0	52	0	3.60	342.00	0	0.00	1,021.00	70
15	80	82	100	0	3.50	338.30	0	0.00	1,021.50	65
16	61	100	48	0	3.70	335.80	0	0.00	1,021.00	62
17	64	100	30	0	3.70	334.50	0	0.00	1,021.00	58
18	76	100	23	22	4.20	336.30	0	0.00	1,020.50	58
19	82	98	11	10	4.10	336.80	0	0.00	1,019.50	58
20	85	100	11	100	4.50	338.30	0	0.00	1,019.00	59
21	83	94	17	100	5.00	342.30	0	0.00	1,019.50	62
22	83	0	7	100	5.10	342.30	0	0.00	1,020.00	63
23	90	0	9	58	5.30	342.00	0	0.00	1,019.50	67
24	79	0	13	14	5.50	342.00	0	0.00	1,018.50	70
25	65	0	34	82	5.90	341.50	0	0.00	1,018.00	76
26	88	0	77	15	5.90	340.50	0	0.00	1,018.00	77

	dep_clouds	dep_clouds_hi	dep_clouds_low	dep_clouds_mid	dep_dewpt	dep_ozone	dep_pop	dep_precip	dep_pres	dep_rh
27	93	100	94	0	6.80	338.30	0	0.00	1,017.50	81
28	91	100	100	3	7.00	337.30	0	0.00	1,017.50	82
29	96	100	100	52	7.20	340.80	25	0.25	1,018.00	82
30	95	100	100	100	7.20	343.80	40	0.50	1,017.50	83
31	95	100	100	100	7.70	345.30	25	0.25	1,016.50	85
32	97	100	100	100	8.00	343.50	40	0.50	1,015.50	87
33	96	100	100	95	8.50	347.00	25	0.25	1,015.00	88
34	97	100	100	100	8.90	350.00	25	0.25	1,014.50	89
35	94	84	100	100	8.50	347.80	0	0.00	1,014.50	89
36	91	80	100	100	8.30	349.00	0	0.00	1,014.50	85
37	89	100	100	100	8.30	352.00	0	0.00	1,015.50	82
38	86	73	100	100	8.00	356.30	0	0.00	1,015.00	78
39	80	83	100	100	7.30	361.00	0	0.00	1,015.50	70
40	75	66	100	91	6.20	364.80	0	0.00	1,014.50	61
41	69	25	58	100	5.10	370.00	0	0.00	1,014.50	53
42	68	22	100	2	4.50	374.80	0	0.00	1,015.00	49
43	66	27	40	0	4.30	374.50	0	0.00	1,014.50	46
44	65	16	35	0	3.60	375.80	0	0.00	1,014.00	42
45	65	2	25	31	3.60	378.50	0	0.00	1,009.00	42
46	64	100	91	67	3.30	381.80	0	0.00	1,008.50	41
47	64	6	76	0	3.30	386.50	0	0.00	1,009.00	41
48	50	0	55	0	2.70	387.00	0	0.00	1,009.50	42
49	36	0	9	0	2.40	391.30	0	0.00	1,010.50	44

50 rows × 24 columns

Out[33]:
```
504
```

Out[33]: 168

Out[33]: 23

Out[33]: 23

In [34]:
```
pred_data.to_csv('pred_mid.csv')
```

In [35]:
```
pred_data['arr_day'] = pred_data['DAY']
pred_data['dep_day'] = pred_data['DAY']
pred_data.drop(columns = ['DAY','DEPARTURE TIME','ARRIVAL TIME','ARRIVAL STATUS_Prev_flight_early',
                         'ARRIVAL STATUS_Prev_flight_ontime','ARRIVAL STATUS_Prev_flight_late',
                         'FLIGHT NUMBER','Date','ARRIVAL STATUS'],inplace = True)
```

In [36]:
```
pred_data.dtypes
```

Out[36]:
```
Origin_Airport              object
dep_order                   object
Carrier_Code                object
dep_hour                    object
dep_min                     object
arr_hour                    object
arr_min                     object
dep_minutes                 object
dep_hours                   object
arr_minutes                 object
arr_hours                   object
arr_clouds                   int64
arr_clouds_hi                int64
arr_clouds_low               int64
arr_clouds_mid               int64
arr_dewpt                  float64
arr_ozone                  float64
arr_pop                      int64
arr_precip                 float64
arr_pres                   float64
arr_rh                       int64
arr_snow                     int64
arr_snow_depth               int64
arr_temp                   float64
arr_vis                    float64
arr_weather.description     object
arr_weather.code             int64
arr_wind_cdir               object
arr_wind_cdir_full          object
arr_wind_dir                 int64
arr_wind_gust_spd          float64
arr_wind_spd               float64
dep_clouds                   int64
dep_clouds_hi                int64
dep_clouds_low               int64
dep_clouds_mid               int64
dep_dewpt                  float64
dep_ozone                  float64
dep_pop                      int64
dep_precip                 float64
dep_pres                   float64
dep_rh                       int64
dep_snow                     int64
```

```
dep_snow_depth               int64
dep_temp                   float64
dep_vis                    float64
dep_weather.description     object
dep_weather.code             int64
dep_wind_cdir               object
dep_wind_cdir_full          object
dep_wind_dir                 int64
dep_wind_gust_spd          float64
dep_wind_spd               float64
arr_day                     object
dep_day                     object
dtype: object
```

In [37]:

```
pred_data['dep_hour'] = pd.Categorical(pred_data['dep_hour'], categories=[i for i in range(24)])
pred_data['dep_day'] = pd.Categorical(pred_data['dep_day'], categories=[i for i in range(7)])
pred_data['dep_min'] = pd.Categorical(pred_data['dep_min'], categories=[i for i in range(60)])
pred_data['arr_hour'] = pd.Categorical(pred_data['arr_hour'], categories=[i for i in range(24)])
pred_data['arr_day'] = pd.Categorical(pred_data['arr_day'], categories=[i for i in range(7)])
pred_data['arr_min'] = pd.Categorical(pred_data['arr_min'], categories=[i for i in range(60)])
pred_data['Carrier_Code'] = pd.Categorical(pred_data['Carrier_Code'], categories=['AA', 'UA', 'DL', 'B6',
pred_data['Origin_Airport'] = pd.Categorical(pred_data['Origin_Airport'], categories=['ORD', 'JFK', 'MCO']
pred_data=pred_data[['dep_hour', 'dep_day', 'Origin_Airport', 'arr_hour', 'arr_day',
       'dep_min', 'arr_min', 'arr_clouds', 'arr_dewpt',
       'arr_precip', 'arr_pres', 'arr_rh', 'arr_snow', 'arr_temp', 'arr_vis',
       'arr_weather.code', 'arr_wind_dir', 'arr_wind_gust_spd', 'arr_wind_spd',
       'dep_clouds', 'dep_dewpt', 'dep_precip', 'dep_pres', 'dep_rh',
       'dep_snow', 'dep_temp', 'dep_vis', 'dep_weather.code', 'dep_wind_dir',
       'dep_wind_gust_spd', 'dep_wind_spd','dep_order']]
pred_data.head()
pred_data.columns
```

Out[37]:

	dep_hour	dep_day	Origin_Airport	arr_hour	arr_day	dep_min	arr_min	arr_clouds	arr_dewpt	arr_precip	...	dep_pres	dep_r
0	18	4	ORD	21	4	52	47	69	8.00	0.50	...	996.50	3
1	19	4	ORD	22	4	59	52	66	7.20	0.25	...	996.50	3
2	13	4	JFK	14	4	34	51	84	7.20	0.50	...	1,019.50	5
3	14	4	JFK	16	4	55	21	86	7.90	0.76	...	1,019.00	5
4	11	4	MCO	14	4	35	20	84	7.20	0.50	...	1,015.00	5

5 rows × 32 columns

Out[37]:

```
Index(['dep_hour', 'dep_day', 'Origin_Airport', 'arr_hour', 'arr_day',
       'dep_min', 'arr_min', 'arr_clouds', 'arr_dewpt', 'arr_precip',
       'arr_pres', 'arr_rh', 'arr_snow', 'arr_temp', 'arr_vis',
       'arr_weather.code', 'arr_wind_dir', 'arr_wind_gust_spd', 'arr_wind_spd',
       'dep_clouds', 'dep_dewpt', 'dep_precip', 'dep_pres', 'dep_rh',
       'dep_snow', 'dep_temp', 'dep_vis', 'dep_weather.code', 'dep_wind_dir',
       'dep_wind_gust_spd', 'dep_wind_spd', 'dep_order'],
      dtype='object')
```

Spliting data into latter and early flights

In [38]:

```
pred_data1 = pred_data[pred_data['dep_order'] == 'early']
pred_data2 = pred_data[pred_data['dep_order'] == 'latter']
```

In [39]:

```
pred_data1.drop(columns=['dep_order'],inplace = True)
pred_data2.drop(columns=['dep_order'],inplace = True)
```

```
C:\Users\gurud\AppData\Local\Temp\ipykernel_27312\2195326561.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  pred_data1.drop(columns=['dep_order'],inplace = True)
C:\Users\gurud\AppData\Local\Temp\ipykernel_27312\2195326561.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  pred_data2.drop(columns=['dep_order'],inplace = True)
```

In [40]:

```
pred_data1.head(22)
```

Out[40]:

	dep_hour	dep_day	Origin_Airport	arr_hour	arr_day	dep_min	arr_min	arr_clouds	arr_dewpt	arr_precip	...	dep_precip	de
0	18	4	ORD	21	4	52	47	69	8.00	0.50	...	0.00	
2	13	4	JFK	14	4	34	51	84	7.20	0.50	...	0.00	1,
4	11	4	MCO	14	4	35	20	84	7.20	0.50	...	0.00	1,
6	18	5	ORD	21	5	52	47	23	-2.40	0.00	...	0.00	
8	13	5	JFK	14	5	25	41	71	-1.20	0.00	...	0.00	1,
10	13	5	MCO	16	5	35	25	58	-1.60	0.00	...	0.00	1,
11	18	6	ORD	21	6	52	47	9	-2.90	0.00	...	0.00	
13	13	6	JFK	14	6	35	51	44	-3.10	0.00	...	0.00	1,
15	11	6	MCO	13	6	5	50	48	-2.80	0.00	...	0.00	1,
17	18	0	ORD	21	0	52	47	10	-4.70	0.00	...	0.00	
19	13	0	JFK	14	0	35	51	10	-4.80	0.00	...	0.00	1,
21	11	0	MCO	14	0	35	20	10	-4.80	0.00	...	0.12	1,

12 rows × 31 columns

In [41]:
```
pred_data2.head(20)
```

Out[41]:

	dep_hour	dep_day	Origin_Airport	arr_hour	arr_day	dep_min	arr_min	arr_clouds	arr_dewpt	arr_precip	...	dep_precip	de
1	19	4	ORD	22	4	59	52	66	7.20	0.25	...	0.00	
3	14	4	JFK	16	4	55	21	86	7.90	0.76	...	0.00	1,
5	13	4	MCO	16	4	35	25	86	7.90	0.76	...	0.00	1,
7	19	5	ORD	22	5	59	52	23	-2.40	0.00	...	0.00	
9	14	5	JFK	16	5	55	21	58	-1.60	0.00	...	0.00	1,
12	19	6	ORD	22	6	59	52	9	-2.90	0.00	...	0.00	
14	14	6	JFK	16	6	55	21	32	-3.20	0.00	...	0.00	1,
16	13	6	MCO	16	6	35	25	32	-3.20	0.00	...	0.00	1,
18	19	0	ORD	22	0	59	52	13	-4.30	0.00	...	0.00	
20	14	0	JFK	16	0	55	21	9	-4.60	0.00	...	0.00	1,
22	13	0	MCO	16	0	34	25	9	-4.60	0.00	...	0.20	1,

11 rows × 31 columns

In [42]:
```
pred_data1.to_csv('pred_data1.csv')
```

In [43]:
```
len(pred_data1)
```

Out[43]: 12

In [44]:
```
pred_data1.head(20)
```

Out[44]:

	dep_hour	dep_day	Origin_Airport	arr_hour	arr_day	dep_min	arr_min	arr_clouds	arr_dewpt	arr_precip	...	dep_precip	de
0	18	4	ORD	21	4	52	47	69	8.00	0.50	...	0.00	
2	13	4	JFK	14	4	34	51	84	7.20	0.50	...	0.00	1,
4	11	4	MCO	14	4	35	20	84	7.20	0.50	...	0.00	1,
6	18	5	ORD	21	5	52	47	23	-2.40	0.00	...	0.00	
8	13	5	JFK	14	5	25	41	71	-1.20	0.00	...	0.00	1,
10	13	5	MCO	16	5	35	25	58	-1.60	0.00	...	0.00	1,
11	18	6	ORD	21	6	52	47	9	-2.90	0.00	...	0.00	
13	13	6	JFK	14	6	35	51	44	-3.10	0.00	...	0.00	1,
15	11	6	MCO	13	6	5	50	48	-2.80	0.00	...	0.00	1,
17	18	0	ORD	21	0	52	47	10	-4.70	0.00	...	0.00	
19	13	0	JFK	14	0	35	51	10	-4.80	0.00	...	0.00	1,
21	11	0	MCO	14	0	35	20	10	-4.80	0.00	...	0.12	1,

12 rows × 31 columns

In [45]:
```
pred_data2.to_csv('pred_data2.csv')
```

In [46]:
```
len(pred_data2)
```

Out[46]: 11

In [ ]: