In [1]:

```
import os
os.getcwd()
```

Out[1]:

```
'C:\\Users\\saima\\Downloads'
```

In [2]:

```
os.chdir("C:/Users/saima/Desktop/Datascience")
```

In [3]:

```
os.getcwd()
```

Out[3]:

```
'C:\\Users\\saima\\Desktop\\Datascience'
```

In [4]:

```
import pandas as pd
import numpy as np
flights=pd.read_csv("DelayedFlights.csv")
flights
```

Out[4]:

| | Unnamed: 0 | Year | Month | DayofMonth | DayOfWeek | DepTime | CRSDepTime | ArrTime |
|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 2008 | 1 | 3 | 4 | 2003 | 1955 | 2211.0 |
| **1** | 1 | 2008 | 1 | 3 | 4 | 754 | 735 | 1002.0 |
| **2** | 2 | 2008 | 1 | 3 | 4 | 628 | 620 | 804.0 |
| **3** | 4 | 2008 | 1 | 3 | 4 | 1829 | 1755 | 1959.0 |
| **4** | 5 | 2008 | 1 | 3 | 4 | 1940 | 1915 | 2121.0 |
| **5** | 6 | 2008 | 1 | 3 | 4 | 1937 | 1830 | 2037.0 |
| **6** | 10 | 2008 | 1 | 3 | 4 | 706 | 700 | 916.0 |
| **7** | 11 | 2008 | 1 | 3 | 4 | 1644 | 1510 | 1845.0 |
| **8** | 15 | 2008 | 1 | 3 | 4 | 1029 | 1020 | 1021.0 |
| **9** | 16 | 2008 | 1 | 3 | 4 | 1452 | 1425 | 1640.0 |
| **10** | 17 | 2008 | 1 | 3 | 4 | 754 | 745 | 940.0 |
| **11** | 18 | 2008 | 1 | 3 | 4 | 1323 | 1255 | 1526.0 |
| **12** | 19 | 2008 | 1 | 3 | 4 | 1416 | 1325 | 1512.0 |
| **13** | 21 | 2008 | 1 | 3 | 4 | 1657 | 1625 | 1754.0 |
| **14** | 22 | 2008 | 1 | 3 | 4 | 1900 | 1840 | 1956.0 |
| **15** | 23 | 2008 | 1 | 3 | 4 | 1039 | 1030 | 1133.0 |
| **16** | 25 | 2008 | 1 | 3 | 4 | 1520 | 1455 | 1619.0 |
| **17** | 26 | 2008 | 1 | 3 | 4 | 1422 | 1255 | 1657.0 |
| **18** | 27 | 2008 | 1 | 3 | 4 | 1954 | 1925 | 2239.0 |
| **19** | 30 | 2008 | 1 | 3 | 4 | 2107 | 1945 | 2334.0 |
| **20** | 33 | 2008 | 1 | 3 | 4 | 1312 | 1300 | 1546.0 |
| **21** | 34 | 2008 | 1 | 3 | 4 | 1449 | 1430 | 1715.0 |
| **22** | 35 | 2008 | 1 | 3 | 4 | 1634 | 1555 | 1859.0 |
| **23** | 37 | 2008 | 1 | 3 | 4 | 1812 | 1650 | 1927.0 |
| **24** | 38 | 2008 | 1 | 3 | 4 | 1127 | 1105 | 1235.0 |
| **25** | 39 | 2008 | 1 | 3 | 4 | 1424 | 1355 | 1531.0 |
| **26** | 40 | 2008 | 1 | 3 | 4 | 1326 | 1230 | 1559.0 |
| **27** | 41 | 2008 | 1 | 3 | 4 | 1749 | 1725 | 2019.0 |
| **28** | 42 | 2008 | 1 | 3 | 4 | 726 | 720 | 958.0 |
| **29** | 43 | 2008 | 1 | 3 | 4 | 646 | 640 | 929.0 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... |
| **1048545** | 3504962 | 2008 | 6 | 29 | 7 | 1310 | 1300 | 1552.0 |
| **1048546** | 3504963 | 2008 | 6 | 30 | 1 | 1337 | 1300 | 1624.0 |
| **1048547** | 3504994 | 2008 | 6 | 1 | 7 | 1340 | 1245 | 1720.0 |
| **1048548** | 3504995 | 2008 | 6 | 2 | 1 | 1330 | 1245 | 1729.0 |
| **1048549** | 3504998 | 2008 | 6 | 5 | 4 | 1357 | 1245 | 1753.0 |

| | Unnamed: 0 | Year | Month | DayofMonth | DayOfWeek | DepTime | CRSDepTime | ArrTime |
|---|---|---|---|---|---|---|---|---|
| **1048550** | 3505003 | 2008 | 6 | 10 | 2 | 1254 | 1245 | 1635.0 |
| **1048551** | 3505004 | 2008 | 6 | 11 | 3 | 1304 | 1245 | 1654.0 |
| **1048552** | 3505006 | 2008 | 6 | 13 | 5 | 1255 | 1245 | 1639.0 |
| **1048553** | 3505010 | 2008 | 6 | 17 | 2 | 1424 | 1245 | 1854.0 |
| **1048554** | 3505011 | 2008 | 6 | 18 | 3 | 1255 | 1245 | 1657.0 |
| **1048555** | 3505012 | 2008 | 6 | 19 | 4 | 1259 | 1245 | 1644.0 |
| **1048556** | 3505014 | 2008 | 6 | 21 | 6 | 1301 | 1245 | 1649.0 |
| **1048557** | 3505015 | 2008 | 6 | 22 | 7 | 1305 | 1245 | 1712.0 |
| **1048558** | 3505018 | 2008 | 6 | 25 | 3 | 1259 | 1245 | 1705.0 |
| **1048559** | 3505021 | 2008 | 6 | 28 | 6 | 1308 | 1245 | 1657.0 |
| **1048560** | 3505022 | 2008 | 6 | 29 | 7 | 1308 | 1245 | 1715.0 |
| **1048561** | 3505024 | 2008 | 6 | 1 | 7 | 1559 | 1535 | 1921.0 |
| **1048562** | 3505027 | 2008 | 6 | 4 | 3 | 1617 | 1535 | 1945.0 |
| **1048563** | 3505030 | 2008 | 6 | 7 | 6 | 1543 | 1535 | 1912.0 |
| **1048564** | 3505031 | 2008 | 6 | 8 | 7 | 1623 | 1535 | 1957.0 |
| **1048565** | 3505033 | 2008 | 6 | 10 | 2 | 1623 | 1535 | 2003.0 |
| **1048566** | 3505035 | 2008 | 6 | 12 | 4 | 1545 | 1535 | 1944.0 |
| **1048567** | 3505036 | 2008 | 6 | 13 | 5 | 1609 | 1535 | 1942.0 |
| **1048568** | 3505037 | 2008 | 6 | 14 | 6 | 1616 | 1535 | 1954.0 |
| **1048569** | 3505040 | 2008 | 6 | 17 | 2 | 1617 | 1535 | 2002.0 |
| **1048570** | 3505042 | 2008 | 6 | 19 | 4 | 1551 | 1535 | 1923.0 |
| **1048571** | 3505043 | 2008 | 6 | 20 | 5 | 1555 | 1535 | 1927.0 |
| **1048572** | 3505044 | 2008 | 6 | 21 | 6 | 1555 | 1535 | 1917.0 |
| **1048573** | 3505045 | 2008 | 6 | 22 | 7 | 1607 | 1535 | 1941.0 |
| **1048574** | 3505046 | 2008 | 6 | 23 | 1 | 1608 | 1535 | 1933.0 |

1048575 rows × 30 columns

In [5]:

```
flights.apply(lambda x:
             sum(x.isnull()),axis=0)
```

Out[5]:

```
Unnamed: 0                0
Year                      0
Month                     0
DayofMonth                0
DayOfWeek                 0
DepTime                   0
CRSDepTime                0
ArrTime                3896
CRSArrTime                0
UniqueCarrier             0
FlightNum                 0
TailNum                   4
ActualElapsedTime      3896
CRSElapsedTime          157
AirTime                3896
ArrDelay               3896
DepDelay                  0
Origin                    0
Dest                      0
Distance                  0
TaxiIn                 3896
TaxiOut                   0
Cancelled                 0
CancellationCode          0
Diverted                  0
CarrierDelay         362841
WeatherDelay         362841
NASDelay             362841
SecurityDelay        362841
LateAircraftDelay    362841
dtype: int64
```

In [6]:

```
flights.columns
```

Out[6]:

```
Index(['Unnamed: 0', 'Year', 'Month', 'DayofMonth', 'DayOfWeek', 'DepTim
e',
       'CRSDepTime', 'ArrTime', 'CRSArrTime', 'UniqueCarrier', 'FlightNu
m',
       'TailNum', 'ActualElapsedTime', 'CRSElapsedTime', 'AirTime', 'ArrDe
lay',
       'DepDelay', 'Origin', 'Dest', 'Distance', 'TaxiIn', 'TaxiOut',
       'Cancelled', 'CancellationCode', 'Diverted', 'CarrierDelay',
       'WeatherDelay', 'NASDelay', 'SecurityDelay', 'LateAircraftDelay'],
      dtype='object')
```

In [13]:

```
flights.dtypes
```

Out[13]:

```
Unnamed: 0             int64
Year                   int64
Month                  int64
DayofMonth             int64
DayOfWeek              int64
DepTime                int64
CRSDepTime             int64
ArrTime              float64
CRSArrTime             int64
UniqueCarrier         object
FlightNum              int64
TailNum               object
ActualElapsedTime    float64
CRSElapsedTime       float64
AirTime              float64
ArrDelay             float64
DepDelay               int64
Origin                object
Dest                  object
Distance               int64
TaxiIn               float64
TaxiOut                int64
Cancelled              int64
CancellationCode      object
Diverted               int64
CarrierDelay         float64
WeatherDelay         float64
NASDelay             float64
SecurityDelay        float64
LateAircraftDelay    float64
dtype: object
```

In [14]:

```
flights.head()
```

Out[14]:

| | Unnamed: 0 | Year | Month | DayofMonth | DayOfWeek | DepTime | CRSDepTime | ArrTime | CRS/ |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 2008 | 1 | 3 | 4 | 2003 | 1955 | 2211.0 | |
| **1** | 1 | 2008 | 1 | 3 | 4 | 754 | 735 | 1002.0 | |
| **2** | 2 | 2008 | 1 | 3 | 4 | 628 | 620 | 804.0 | |
| **3** | 4 | 2008 | 1 | 3 | 4 | 1829 | 1755 | 1959.0 | |
| **4** | 5 | 2008 | 1 | 3 | 4 | 1940 | 1915 | 2121.0 | |

5 rows × 30 columns

In [15]:

```
a=flights['ArrTime'].mean()
a
```

Out[15]:

```
1610.7425285663223
```

In [16]:

```
flights['ArrTime'].fillna(a,inplace=True)
```

In [17]:

```
sum(flights['ArrTime'].isnull())
```

Out[17]:

```
0
```

In [18]:

```
a1=flights['ActualElapsedTime'].mean()
a1
```

Out[18]:

```
131.6941902728015
```

In [19]:

```
flights['ActualElapsedTime'].fillna(a1,inplace=True)
```

In [20]:

```
a2=flights['CRSElapsedTime'].mean()
a2
```

Out[20]:

```
132.30098109723411
```

In [21]:

```
flights['CRSElapsedTime'].fillna(a2,inplace=True)
```

In [22]:

```
a3=flights['AirTime'].mean()
a3
```

Out[22]:

```
107.02498375098953
```

In [23]:

```
a4=flights['ArrDelay'].mean()
a4
```

Out[23]:

```
42.18256804243217
```

In [24]:

```
a5=flights['TaxiIn'].mean()
a5
```

Out[24]:

```
6.683526710118611
```

In [25]:

```
a6=flights['CarrierDelay'].mean()
a6
```

Out[25]:

```
18.870359352168624
```

In [26]:

```
a7=flights['WeatherDelay'].mean()
a7
```

Out[26]:

```
3.5680322107406077
```

In [27]:

```
a8=flights['NASDelay'].mean()
a8
```

Out[27]:

```
14.429618481801981
```

In [28]:

```
a9=flights['SecurityDelay'].mean()
a9
```

Out[28]:

```
0.093283984752105 05
```

In [29]:

```
a10=flights['LateAircraftDelay'].mean()
a10
```

Out[29]:

```
25.334310388576327
```

In [30]:

```
s=flights['TailNum'].mode()
```

In [31]:

```
s
```

Out[31]:

```
0    N325SW
dtype: object
```

In [32]:

```
flights['ActualElapsedTime'].fillna(a1,inplace=True)
```

In [33]:

```
flights['CRSElapsedTime'].fillna(a2,inplace=True)
```

In [34]:

```
flights['AirTime'].fillna(a3,inplace=True)
```

In [35]:

```
flights['ArrDelay'].fillna(a4,inplace=True)
```

In [36]:

```
flights['TaxiIn'].fillna(a5,inplace=True)
```

In [37]:

```
flights['CarrierDelay'].fillna(a6,inplace=True)
```

In [38]:

```
flights['WeatherDelay'].fillna(a7,inplace=True)
```

In [39]:

```
flights['NASDelay'].fillna(a8,inplace=True)
```

In [40]:

```
flights['SecurityDelay'].fillna(a9,inplace=True)
```

In [41]:

```
flights['SecurityDelay'].fillna(a9,inplace=True)
```

In [42]:

```
flights['TailNum'].fillna('N325SW',inplace=True)
```

In [43]:

```
flights.apply(lambda x:
             sum(x.isnull()),axis=0)
```

Out[43]:

```
Unnamed: 0                0
Year                      0
Month                     0
DayofMonth                0
DayOfWeek                 0
DepTime                   0
CRSDepTime                0
ArrTime                   0
CRSArrTime                0
UniqueCarrier             0
FlightNum                 0
TailNum                   0
ActualElapsedTime         0
CRSElapsedTime            0
AirTime                   0
ArrDelay                  0
DepDelay                  0
Origin                    0
Dest                      0
Distance                  0
TaxiIn                    0
TaxiOut                   0
Cancelled                 0
CancellationCode          0
Diverted                  0
CarrierDelay              0
WeatherDelay              0
NASDelay                  0
SecurityDelay             0
LateAircraftDelay    362841
dtype: int64
```

In [44]:

```
import matplotlib.pyplot as plt
%matplotlib inline
```

In [45]:

```
weather_delay=flights['WeatherDelay']
```

In [46]:

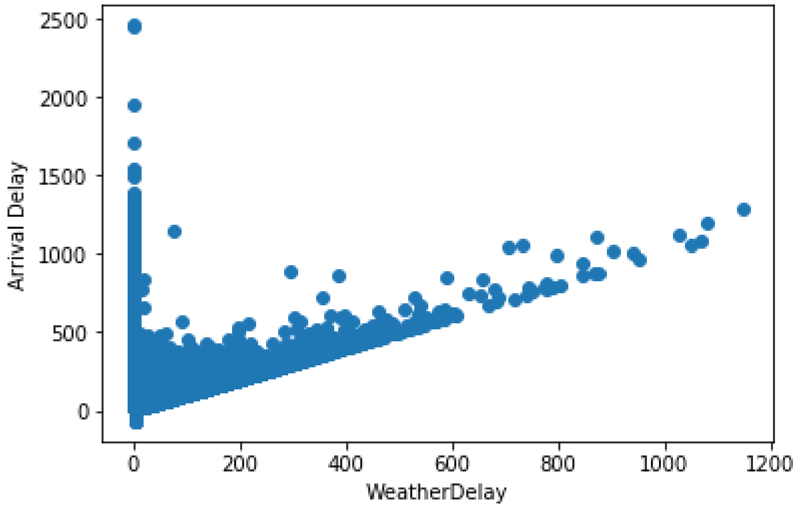
```
arr_delay=flights['ArrDelay']
```

In [47]:

```
plt.plot(weather_delay,arr_delay,'o')
plt.ylabel("Arrival Delay")
plt.xlabel("WeatherDelay")
```

Out[47]:

```
Text(0.5, 0, 'WeatherDelay')
```



In [48]:

```
flights.corr()
```

Out[48]:

| | Unnamed: 0 | Year | Month | DayofMonth | DayOfWeek | DepTime | CRSDep |
|---|---|---|---|---|---|---|---|
| **Unnamed: 0** | 1.000000 | NaN | 0.985386 | 0.015790 | 0.021755 | 0.020686 | 0.02 |
| **Year** | NaN | NaN | NaN | NaN | NaN | NaN | |
| **Month** | 0.985386 | NaN | 1.000000 | 0.020957 | 0.022139 | 0.025462 | 0.03 |
| **DayofMonth** | 0.015790 | NaN | 0.020957 | 1.000000 | -0.021198 | 0.011964 | 0.01 |
| **DayOfWeek** | 0.021755 | NaN | 0.022139 | -0.021198 | 1.000000 | 0.018711 | 0.02 |
| **DepTime** | 0.020686 | NaN | 0.025462 | 0.011964 | 0.018711 | 1.000000 | 0.88 |
| **CRSDepTime** | 0.027435 | NaN | 0.031677 | 0.012325 | 0.026140 | 0.884626 | 1.00 |
| **ArrTime** | 0.002834 | NaN | 0.005400 | 0.008353 | 0.010332 | 0.461435 | 0.40 |
| **CRSArrTime** | 0.023048 | NaN | 0.023018 | 0.010266 | 0.014525 | 0.717095 | 0.71 |
| **FlightNum** | -0.025390 | NaN | 0.003662 | -0.001148 | -0.013528 | -0.026200 | -0.05 |
| **ActualElapsedTime** | 0.026882 | NaN | -0.017787 | -0.003318 | 0.003884 | -0.047771 | -0.03 |
| **CRSElapsedTime** | 0.024039 | NaN | -0.018523 | -0.001647 | 0.008289 | -0.045085 | -0.02 |
| **AirTime** | 0.023091 | NaN | -0.015089 | -0.001849 | 0.007621 | -0.052720 | -0.03 |
| **ArrDelay** | 0.000629 | NaN | -0.008045 | -0.007086 | -0.009428 | 0.132455 | 0.04 |
| **DepDelay** | -0.004318 | NaN | -0.009907 | -0.005631 | -0.004744 | 0.145651 | 0.05 |
| **Distance** | 0.029773 | NaN | -0.008466 | -0.000555 | 0.012396 | -0.054752 | -0.02 |
| **TaxiIn** | 0.010134 | NaN | -0.020889 | -0.016088 | 0.007194 | -0.012480 | -0.03 |
| **TaxiOut** | 0.021345 | NaN | -0.009624 | -0.002460 | -0.020000 | 0.017966 | -0.00 |
| **Cancelled** | NaN | NaN | NaN | NaN | NaN | NaN | |
| **Diverted** | 0.000323 | NaN | -0.000087 | -0.004739 | -0.000888 | -0.006941 | -0.01 |
| **CarrierDelay** | 0.000100 | NaN | -0.004936 | -0.004616 | 0.011942 | -0.041657 | -0.08 |
| **WeatherDelay** | -0.002649 | NaN | -0.002151 | -0.005664 | -0.000542 | 0.006591 | -0.01 |
| **NASDelay** | 0.019101 | NaN | 0.009732 | 0.011199 | -0.024074 | 0.013669 | -0.03 |
| **SecurityDelay** | -0.004215 | NaN | -0.002441 | -0.002882 | 0.006640 | -0.011345 | -0.01 |
| **LateAircraftDelay** | -0.015905 | NaN | -0.009964 | -0.003010 | 0.000107 | 0.180777 | 0.16 |

25 rows × 25 columns

In [49]:

```
mean_arr_delay=flights['ArrDelay'].mean()
```

In [50]:

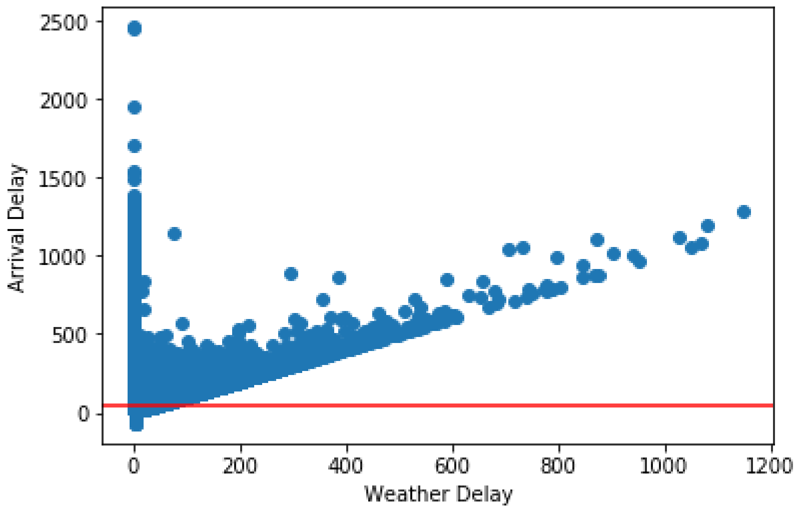
```
mean_arr_delay
```

Out[50]:

```
42.182568042433864
```

In [51]:

```
plt.plot(weather_delay,arr_delay,'o')
plt.ylabel("Arrival Delay")
plt.xlabel("Weather Delay")
plt.axhline(mean_arr_delay,color='r',linestyle='-')
plt.show()
```




In [52]:

```
import statsmodels.api as sm
model=sm.OLS(arr_delay,weather_delay).fit()
```

In [53]:

```
model.summary()
```

Out[53]:

OLS Regression Results

| | | | |
|---|---|---|---|
| **Dep. Variable:** | ArrDelay | **R-squared:** | 0.089 |
| **Model:** | OLS | **Adj. R-squared:** | 0.089 |
| **Method:** | Least Squares | **F-statistic:** | 1.025e+05 |
| **Date:** | Thu, 06 Jun 2019 | **Prob (F-statistic):** | 0.00 |
| **Time:** | 12:54:09 | **Log-Likelihood:** | -5.8915e+06 |
| **No. Observations:** | 1048575 | **AIC:** | 1.178e+07 |
| **Df Residuals:** | 1048574 | **BIC:** | 1.178e+07 |
| **Df Model:** | 1 | | |
| **Covariance Type:** | nonrobust | | |

| | coef | std err | t | P>\|t\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| **WeatherDelay** | 1.2164 | 0.004 | 320.150 | 0.000 | 1.209 | 1.224 |

| | | | |
|---|---|---|---|
| **Omnibus:** | 950040.080 | **Durbin-Watson:** | 1.180 |
| **Prob(Omnibus):** | 0.000 | **Jarque-Bera (JB):** | 94429327.373 |
| **Skew:** | 3.998 | **Prob(JB):** | 0.00 |
| **Kurtosis:** | 48.797 | **Cond. No.** | 1.00 |

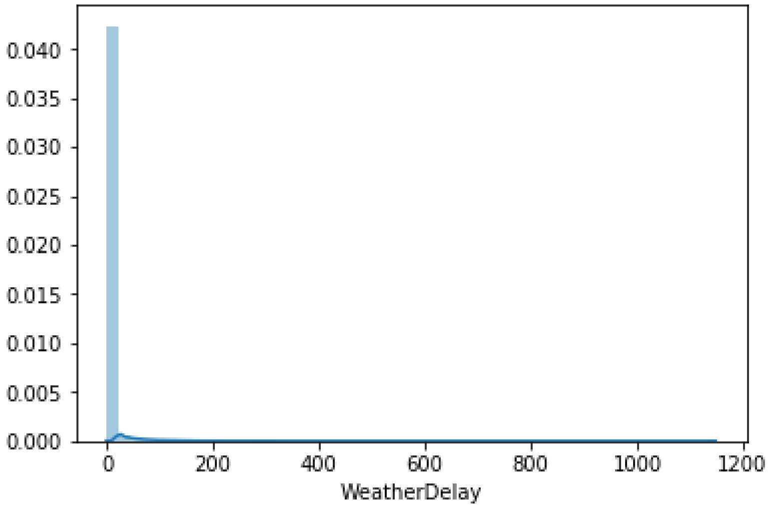
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

In [48]:

```
import seaborn as sns
sns.distplot(flights['WeatherDelay'])
```

Out[48]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1df1dbe2f98>
```

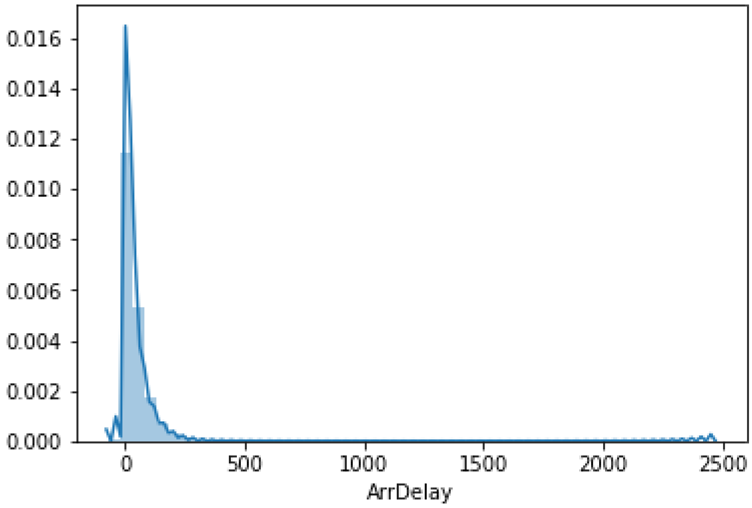


In [49]:

```
sns.distplot(flights['ArrDelay'])
```

Out[49]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1df1dd289b0>
```




In [50]:

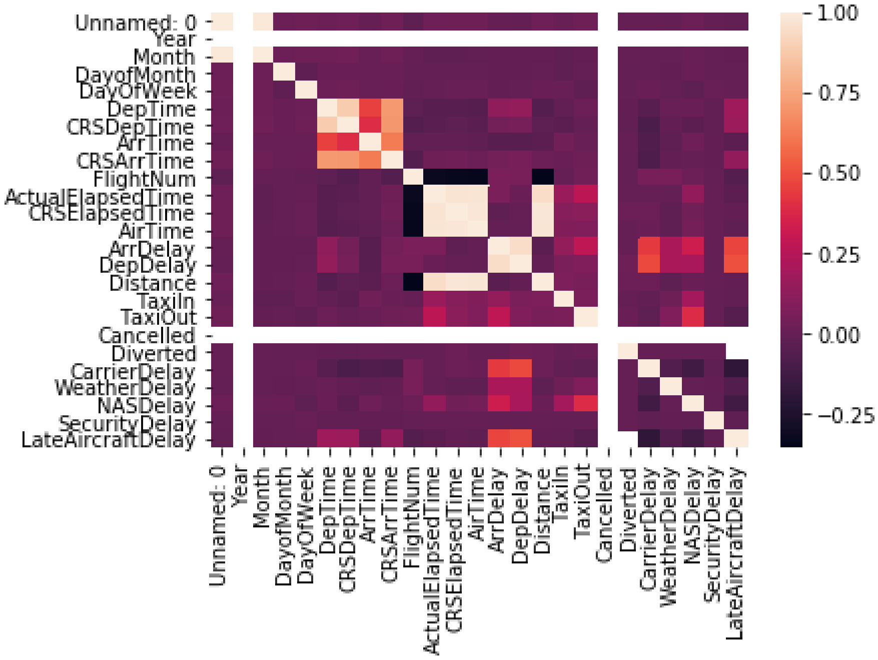
```
corr=flights.corr()
```

In [51]:

```
sns.heatmap(corr,xticklabels=corr.columns,yticklabels=corr.columns)
```

Out[51]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1df1de4bef0>
```

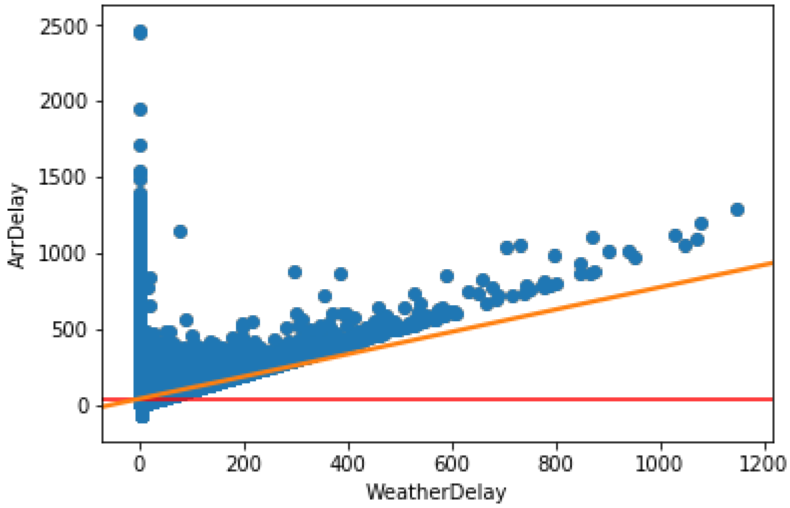


In [52]:

```
plt.plot(weather_delay,arr_delay,'o')
plt.ylabel("Arrival Delay")
plt.xlabel("Weather Delay")
plt.axhline(mean_arr_delay,color='r',linestyle='-')
sns.regplot(x='WeatherDelay',y='ArrDelay',data=flights)
```

Out[52]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1df1df688d0>
```




In [54]:

```
X=flights['WeatherDelay']
y=flights['ArrDelay']
```

In [55]:

```
X
```

Out[55]:

```
0           3.568032
1           3.568032
2           3.568032
3           0.000000
4           3.568032
5           0.000000
6           3.568032
7           0.000000
8           3.568032
9           0.000000
10          3.568032
11          0.000000
12          0.000000
13          0.000000
14          3.568032
15          3.568032
16          3.568032
17          0.000000
18          3.568032
19          0.000000
20          3.568032
21          3.568032
22          3.568032
23          0.000000
24          3.568032
25          3.568032
26          0.000000
27          3.568032
28          3.568032
29          3.568032
              ...
1048545     3.568032
1048546     3.000000
1048547     0.000000
1048548     0.000000
1048549     0.000000
1048550     3.568032
1048551     3.568032
1048552     3.568032
1048553    79.000000
1048554     0.000000
1048555     3.568032
1048556     3.568032
1048557     0.000000
1048558     0.000000
1048559     0.000000
1048560     0.000000
1048561     3.568032
1048562     0.000000
1048563     3.568032
1048564     0.000000
1048565     0.000000
1048566     0.000000
1048567     0.000000
1048568     0.000000
1048569    22.000000
1048570     3.568032
1048571     3.568032
1048572     3.568032
```

```
1048573     0.000000
1048574     0.000000
Name: WeatherDelay, Length: 1048575, dtype: float64
```

In [56]:

```
y
```

Out[56]:

```
0          -14.0
1            2.0
2           14.0
3           34.0
4           11.0
5           57.0
6            1.0
7           80.0
8           11.0
9           15.0
10         -15.0
11          16.0
12          37.0
13          19.0
14           6.0
15          -7.0
16          14.0
17          47.0
18           4.0
19          64.0
20          -4.0
21          -5.0
22          14.0
23          72.0
24           5.0
25          11.0
26          29.0
27         -11.0
28         -22.0
29         -26.0
           ...
1048545     -3.0
1048546     29.0
1048547     40.0
1048548     49.0
1048549     73.0
1048550     -5.0
1048551     14.0
1048552     -1.0
1048553    134.0
1048554     17.0
1048555      4.0
1048556      9.0
1048557     32.0
1048558     25.0
1048559     17.0
1048560     35.0
1048561      6.0
1048562     30.0
1048563     -3.0
1048564     42.0
1048565     48.0
1048566     29.0
1048567     27.0
1048568     39.0
1048569     47.0
1048570      8.0
1048571     12.0
1048572      2.0
```

```
1048573     26.0
1048574     18.0
Name: ArrDelay, Length: 1048575, dtype: float64
```

In [57]:

```
x=X.values.reshape(-1,1)
```

In [58]:

```
x
```

Out[58]:

```
array([[3.56803221],
       [3.56803221],
       [3.56803221],
       ...,
       [3.56803221],
       [0.        ],
       [0.        ]])
```

In [59]:

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=1)
```

In [60]:

```
X_train
```

Out[60]:

```
656253      0.000000
329191      0.000000
557768      3.568032
93060       3.568032
558339      0.000000
650854      3.568032
932168      0.000000
422028      0.000000
812644      3.568032
48833       0.000000
728220      0.000000
910961      3.568032
470296      0.000000
568087      3.568032
846017      3.568032
70352       3.568032
422744      0.000000
570938      0.000000
72418       0.000000
163585      0.000000
105023      3.568032
672639      3.568032
503666      0.000000
16396       0.000000
1023015     0.000000
516494      3.568032
458056      0.000000
942354      3.568032
1031805     0.000000
830602      0.000000
              ...
1005966     3.568032
188317      3.568032
365212      0.000000
806378      0.000000
401660      0.000000
457611      0.000000
575956      3.568032
691090      0.000000
176485      0.000000
21758       3.568032
513300      0.000000
1041586     0.000000
1015065     0.000000
167302      0.000000
293372      3.568032
436973      0.000000
925255      0.000000
966604      0.000000
413825      3.568032
229520      0.000000
21440       0.000000
117583     41.000000
73349       0.000000
371403      3.568032
836489      3.568032
491263      3.568032
791624      0.000000
470924      0.000000
```

```
491755      0.000000
128037      3.568032
Name: WeatherDelay, Length: 838860, dtype: float64
```

In [61]:

```
y_train
```

Out[61]:

```
656253      27.0
329191     100.0
557768      -1.0
93060        1.0
558339      29.0
650854       2.0
932168      62.0
422028      22.0
812644       6.0
48833       57.0
728220      18.0
910961       7.0
470296      16.0
568087       9.0
846017      -5.0
70352        1.0
422744      43.0
570938      19.0
72418       68.0
163585     109.0
105023       2.0
672639       4.0
503666      42.0
16396       29.0
1023015     26.0
516494      10.0
458056      22.0
942354      -7.0
1031805     35.0
830602      42.0
           ...
1005966      7.0
188317       9.0
365212      35.0
806378     226.0
401660      29.0
457611     104.0
575956       4.0
691090      37.0
176485      32.0
21758      -10.0
513300      35.0
1041586     38.0
1015065     49.0
167302     100.0
293372      10.0
436973      66.0
925255     143.0
966604     132.0
413825       8.0
229520     185.0
21440       15.0
117583      41.0
73349      145.0
371403       9.0
836489       8.0
491263      12.0
791624      69.0
470924      24.0
```

```
491755      61.0
128037       4.0
Name: ArrDelay, Length: 838860, dtype: float64
```

In [62]:

```
X_train=X_train.values.reshape((-1,1))
X_train
```

Out[62]:

```
array([[0.        ],
       [0.        ],
       [3.56803221],
       ...,
       [0.        ],
       [0.        ],
       [3.56803221]])
```

In [63]:

```
from sklearn import linear_model as lm
model=lm.LinearRegression()
results=model.fit(X_train,y_train)
```

In [64]:

```
results
```

Out[64]:

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None,
         normalize=False)
```

In [65]:

```
accuracy=model.score(X_train,y_train)
print('Accuracy of the model: ',accuracy)
```

```
Accuracy of the model:  0.049706223189891596
```

In [66]:

```
print('intercept:',model.intercept_)
print('slope:',model.coef_)
```

```
intercept: 39.56487686309654
slope: [0.73925553]
```

In [67]:

```
X_test=X_test.values.reshape((-1,1))
```

In [68]:

```
X_test
```

Out[68]:

```
array([[20.        ],
       [ 0.        ],
       [ 0.        ],
       ...,
       [ 3.56803221],
       [ 0.        ],
       [ 0.        ]])
```

In [69]:

```
predictions=model.predict(X_test)
print('predicted Arrival delays:',predictions,sep='\n')
```

```
predicted Arrival delays:
[54.34998754 39.56487686 39.56487686 ... 42.20256442 39.56487686
 39.56487686]
```

In [70]:

```
predictions
```

Out[70]:

```
array([54.34998754, 39.56487686, 39.56487686, ..., 42.20256442,
       39.56487686, 39.56487686])
```

In [71]:

```
predictions[100]
```

Out[71]:

```
42.20256441990047
```

In [72]:

```
flights.columns
```

Out[72]:

```
Index(['Unnamed: 0', 'Year', 'Month', 'DayofMonth', 'DayOfWeek', 'DepTim
e',
       'CRSDepTime', 'ArrTime', 'CRSArrTime', 'UniqueCarrier', 'FlightNu
m',
       'TailNum', 'ActualElapsedTime', 'CRSElapsedTime', 'AirTime', 'ArrDe
lay',
       'DepDelay', 'Origin', 'Dest', 'Distance', 'TaxiIn', 'TaxiOut',
       'Cancelled', 'CancellationCode', 'Diverted', 'CarrierDelay',
       'WeatherDelay', 'NASDelay', 'SecurityDelay', 'LateAircraftDelay'],
      dtype='object')
```

In [73]:

```
flights.dtypes
```

Out[73]:

```
Unnamed: 0             int64
Year                   int64
Month                  int64
DayofMonth             int64
DayOfWeek              int64
DepTime                int64
CRSDepTime             int64
ArrTime              float64
CRSArrTime             int64
UniqueCarrier         object
FlightNum              int64
TailNum               object
ActualElapsedTime    float64
CRSElapsedTime       float64
AirTime              float64
ArrDelay             float64
DepDelay               int64
Origin                object
Dest                  object
Distance               int64
TaxiIn               float64
TaxiOut                int64
Cancelled              int64
CancellationCode      object
Diverted               int64
CarrierDelay         float64
WeatherDelay         float64
NASDelay             float64
SecurityDelay        float64
LateAircraftDelay    float64
dtype: object
```

In [74]:

```
X=flights[['Month','DayofMonth','DayOfWeek','DepTime','CRSDepTime','ArrTime','CRSArrTim
e','FlightNum','ActualElapsedTime','CRSElapsedTime', 'AirTime',
       'DepDelay','Distance', 'TaxiIn', 'TaxiOut',
       'Cancelled','Diverted', 'CarrierDelay',
       'WeatherDelay', 'NASDelay', 'SecurityDelay']]
```

In [75]:

```
y=flights["ArrDelay"]
```

In [76]:

```
flights.apply(lambda x:
             sum(x.isnull()),axis=0)
```

Out[76]:

```
Unnamed: 0                0
Year                      0
Month                     0
DayofMonth                0
DayOfWeek                 0
DepTime                   0
CRSDepTime                0
ArrTime                   0
CRSArrTime                0
UniqueCarrier             0
FlightNum                 0
TailNum                   0
ActualElapsedTime         0
CRSElapsedTime            0
AirTime                   0
ArrDelay                  0
DepDelay                  0
Origin                    0
Dest                      0
Distance                  0
TaxiIn                    0
TaxiOut                   0
Cancelled                 0
CancellationCode          0
Diverted                  0
CarrierDelay              0
WeatherDelay              0
NASDelay                  0
SecurityDelay             0
LateAircraftDelay    362841
dtype: int64
```

In [77]:

```
a11=flights['LateAircraftDelay'].mean()
flights['LateAircraftDelay'].fillna(a11,inplace=True)
```

In [78]:

```
flights.apply(lambda x:
            sum(x.isnull()),axis=0)
```

Out[78]:

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

In [79]:

```
X
```

Out[79]:

| | Month | DayofMonth | DayOfWeek | DepTime | CRSDepTime | ArrTime | CRSArrTime | Fli |
|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 3 | 4 | 2003 | 1955 | 2211.0 | 2225 | |
| **1** | 1 | 3 | 4 | 754 | 735 | 1002.0 | 1000 | |
| **2** | 1 | 3 | 4 | 628 | 620 | 804.0 | 750 | |
| **3** | 1 | 3 | 4 | 1829 | 1755 | 1959.0 | 1925 | |
| **4** | 1 | 3 | 4 | 1940 | 1915 | 2121.0 | 2110 | |
| **5** | 1 | 3 | 4 | 1937 | 1830 | 2037.0 | 1940 | |
| **6** | 1 | 3 | 4 | 706 | 700 | 916.0 | 915 | |
| **7** | 1 | 3 | 4 | 1644 | 1510 | 1845.0 | 1725 | |
| **8** | 1 | 3 | 4 | 1029 | 1020 | 1021.0 | 1010 | |
| **9** | 1 | 3 | 4 | 1452 | 1425 | 1640.0 | 1625 | |
| **10** | 1 | 3 | 4 | 754 | 745 | 940.0 | 955 | |
| **11** | 1 | 3 | 4 | 1323 | 1255 | 1526.0 | 1510 | |
| **12** | 1 | 3 | 4 | 1416 | 1325 | 1512.0 | 1435 | |
| **13** | 1 | 3 | 4 | 1657 | 1625 | 1754.0 | 1735 | |
| **14** | 1 | 3 | 4 | 1900 | 1840 | 1956.0 | 1950 | |
| **15** | 1 | 3 | 4 | 1039 | 1030 | 1133.0 | 1140 | |
| **16** | 1 | 3 | 4 | 1520 | 1455 | 1619.0 | 1605 | |
| **17** | 1 | 3 | 4 | 1422 | 1255 | 1657.0 | 1610 | |
| **18** | 1 | 3 | 4 | 1954 | 1925 | 2239.0 | 2235 | |
| **19** | 1 | 3 | 4 | 2107 | 1945 | 2334.0 | 2230 | |
| **20** | 1 | 3 | 4 | 1312 | 1300 | 1546.0 | 1550 | |
| **21** | 1 | 3 | 4 | 1449 | 1430 | 1715.0 | 1720 | |
| **22** | 1 | 3 | 4 | 1634 | 1555 | 1859.0 | 1845 | |
| **23** | 1 | 3 | 4 | 1812 | 1650 | 1927.0 | 1815 | |
| **24** | 1 | 3 | 4 | 1127 | 1105 | 1235.0 | 1230 | |
| **25** | 1 | 3 | 4 | 1424 | 1355 | 1531.0 | 1520 | |
| **26** | 1 | 3 | 4 | 1326 | 1230 | 1559.0 | 1530 | |
| **27** | 1 | 3 | 4 | 1749 | 1725 | 2019.0 | 2030 | |
| **28** | 1 | 3 | 4 | 726 | 720 | 958.0 | 1020 | |
| **29** | 1 | 3 | 4 | 646 | 640 | 929.0 | 955 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | |
| **1048545** | 6 | 29 | 7 | 1310 | 1300 | 1552.0 | 1555 | |
| **1048546** | 6 | 30 | 1 | 1337 | 1300 | 1624.0 | 1555 | |
| **1048547** | 6 | 1 | 7 | 1340 | 1245 | 1720.0 | 1640 | |
| **1048548** | 6 | 2 | 1 | 1330 | 1245 | 1729.0 | 1640 | |
| **1048549** | 6 | 5 | 4 | 1357 | 1245 | 1753.0 | 1640 | |
| **1048550** | 6 | 10 | 2 | 1254 | 1245 | 1635.0 | 1640 | |

| | Month | DayofMonth | DayOfWeek | DepTime | CRSDepTime | ArrTime | CRSArrTime | Fli |
|---|---|---|---|---|---|---|---|---|
| **1048551** | 6 | 11 | 3 | 1304 | 1245 | 1654.0 | 1640 | |
| **1048552** | 6 | 13 | 5 | 1255 | 1245 | 1639.0 | 1640 | |
| **1048553** | 6 | 17 | 2 | 1424 | 1245 | 1854.0 | 1640 | |
| **1048554** | 6 | 18 | 3 | 1255 | 1245 | 1657.0 | 1640 | |
| **1048555** | 6 | 19 | 4 | 1259 | 1245 | 1644.0 | 1640 | |
| **1048556** | 6 | 21 | 6 | 1301 | 1245 | 1649.0 | 1640 | |
| **1048557** | 6 | 22 | 7 | 1305 | 1245 | 1712.0 | 1640 | |
| **1048558** | 6 | 25 | 3 | 1259 | 1245 | 1705.0 | 1640 | |
| **1048559** | 6 | 28 | 6 | 1308 | 1245 | 1657.0 | 1640 | |
| **1048560** | 6 | 29 | 7 | 1308 | 1245 | 1715.0 | 1640 | |
| **1048561** | 6 | 1 | 7 | 1559 | 1535 | 1921.0 | 1915 | |
| **1048562** | 6 | 4 | 3 | 1617 | 1535 | 1945.0 | 1915 | |
| **1048563** | 6 | 7 | 6 | 1543 | 1535 | 1912.0 | 1915 | |
| **1048564** | 6 | 8 | 7 | 1623 | 1535 | 1957.0 | 1915 | |
| **1048565** | 6 | 10 | 2 | 1623 | 1535 | 2003.0 | 1915 | |
| **1048566** | 6 | 12 | 4 | 1545 | 1535 | 1944.0 | 1915 | |
| **1048567** | 6 | 13 | 5 | 1609 | 1535 | 1942.0 | 1915 | |
| **1048568** | 6 | 14 | 6 | 1616 | 1535 | 1954.0 | 1915 | |
| **1048569** | 6 | 17 | 2 | 1617 | 1535 | 2002.0 | 1915 | |
| **1048570** | 6 | 19 | 4 | 1551 | 1535 | 1923.0 | 1915 | |
| **1048571** | 6 | 20 | 5 | 1555 | 1535 | 1927.0 | 1915 | |
| **1048572** | 6 | 21 | 6 | 1555 | 1535 | 1917.0 | 1915 | |
| **1048573** | 6 | 22 | 7 | 1607 | 1535 | 1941.0 | 1915 | |
| **1048574** | 6 | 23 | 1 | 1608 | 1535 | 1933.0 | 1915 | |

1048575 rows × 21 columns

In [80]:

```
y
```

Out[80]:

```
0          -14.0
1            2.0
2           14.0
3           34.0
4           11.0
5           57.0
6            1.0
7           80.0
8           11.0
9           15.0
10         -15.0
11          16.0
12          37.0
13          19.0
14           6.0
15          -7.0
16          14.0
17          47.0
18           4.0
19          64.0
20          -4.0
21          -5.0
22          14.0
23          72.0
24           5.0
25          11.0
26          29.0
27         -11.0
28         -22.0
29         -26.0
           ...
1048545     -3.0
1048546     29.0
1048547     40.0
1048548     49.0
1048549     73.0
1048550     -5.0
1048551     14.0
1048552     -1.0
1048553    134.0
1048554     17.0
1048555      4.0
1048556      9.0
1048557     32.0
1048558     25.0
1048559     17.0
1048560     35.0
1048561      6.0
1048562     30.0
1048563     -3.0
1048564     42.0
1048565     48.0
1048566     29.0
1048567     27.0
1048568     39.0
1048569     47.0
1048570      8.0
1048571     12.0
1048572      2.0
```

```
1048573     26.0
1048574     18.0
Name: ArrDelay, Length: 1048575, dtype: float64
```

In [81]:

```
x=X.values.reshape(-1,1)
```

In [82]:

```
x
```

Out[82]:

```
array([[1.],
       [3.],
       [4.],
       ...,
       [0.],
       [0.],
       [0.]])
```

In [83]:

```
#split into training and test data
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.4,random_state=1) #give
 same random data to all machines 0.3,0.1,0.25.....
```

In [84]:

```
X_train
```

Out[84]:

| | Month | DayofMonth | DayOfWeek | DepTime | CRSDepTime | ArrTime | CRSArrTime | Fli |
|---|---|---|---|---|---|---|---|---|
| **109881** | 1 | 19 | 6 | 2009 | 1945 | 2114.0 | 2056 | |
| **893552** | 6 | 11 | 3 | 1414 | 1340 | 1538.0 | 1430 | |
| **695917** | 4 | 27 | 7 | 1512 | 1455 | 1634.0 | 1625 | |
| **652905** | 4 | 28 | 1 | 2003 | 1745 | 2203.0 | 2005 | |
| **666397** | 4 | 28 | 1 | 1030 | 1016 | 1154.0 | 1140 | |
| **394973** | 3 | 19 | 3 | 1053 | 1040 | 1324.0 | 1325 | |
| **599286** | 4 | 29 | 2 | 1227 | 1125 | 1400.0 | 1315 | |
| **746605** | 5 | 20 | 2 | 2025 | 1910 | 2114.0 | 2005 | |
| **827928** | 5 | 14 | 3 | 1413 | 1250 | 1631.0 | 1455 | |
| **176314** | 1 | 22 | 2 | 1259 | 1235 | 1603.0 | 1521 | |
| **788152** | 5 | 20 | 2 | 1649 | 1525 | 1932.0 | 1820 | |
| **193069** | 2 | 9 | 6 | 1656 | 1650 | 2057.0 | 2100 | |
| **817926** | 5 | 26 | 1 | 1332 | 1314 | 1615.0 | 1600 | |
| **562633** | 3 | 17 | 1 | 1104 | 1030 | 1503.0 | 1502 | |
| **496269** | 3 | 24 | 1 | 935 | 915 | 1045.0 | 1020 | |
| **692581** | 4 | 21 | 1 | 2257 | 2105 | 741.0 | 520 | |
| **242349** | 2 | 4 | 1 | 1144 | 1115 | 1316.0 | 1230 | |
| **989277** | 6 | 29 | 7 | 1811 | 1749 | 1949.0 | 1848 | |
| **565880** | 3 | 22 | 6 | 1155 | 1145 | 1500.0 | 1501 | |
| **301455** | 2 | 25 | 1 | 1719 | 1710 | 1834.0 | 1829 | |
| **445079** | 3 | 26 | 3 | 1510 | 1455 | 1608.0 | 1542 | |
| **725979** | 4 | 6 | 7 | 1829 | 1700 | 2132.0 | 2028 | |
| **551345** | 3 | 5 | 3 | 1936 | 1910 | 2157.0 | 2150 | |
| **94293** | 1 | 2 | 3 | 1701 | 1605 | 1744.0 | 1655 | |
| **991032** | 6 | 30 | 1 | 2204 | 2118 | 2155.0 | 2116 | |
| **411125** | 3 | 31 | 1 | 1828 | 1745 | 1843.0 | 1800 | |
| **98609** | 1 | 24 | 4 | 1137 | 1115 | 1329.0 | 1255 | |
| **704189** | 4 | 17 | 4 | 1959 | 1929 | 2241.0 | 2229 | |
| **335617** | 2 | 1 | 5 | 1829 | 1755 | 2034.0 | 1950 | |
| **523592** | 3 | 20 | 4 | 1242 | 1219 | 1248.0 | 1250 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | |
| **1005966** | 6 | 13 | 5 | 1335 | 1325 | 2152.0 | 2145 | |
| **188317** | 2 | 6 | 3 | 1410 | 1355 | 1649.0 | 1640 | |
| **365212** | 2 | 14 | 4 | 1847 | 1800 | 2149.0 | 2114 | |
| **806378** | 5 | 1 | 4 | 1931 | 1610 | 206.0 | 2220 | |
| **401660** | 3 | 24 | 1 | 2156 | 2125 | 2259.0 | 2230 | |
| **457611** | 3 | 25 | 2 | 1923 | 1743 | 2057.0 | 1913 | |

| | Month | DayofMonth | DayOfWeek | DepTime | CRSDepTime | ArrTime | CRSArrTime | Fli |
|---|---|---|---|---|---|---|---|---|
| **575956** | 4 | 4 | 5 | 919 | 900 | 1144.0 | 1140 | |
| **691090** | 4 | 8 | 2 | 1345 | 1325 | 1542.0 | 1505 | |
| **176485** | 1 | 1 | 2 | 1813 | 1725 | 1947.0 | 1915 | |
| **21758** | 1 | 24 | 4 | 856 | 850 | 1140.0 | 1150 | |
| **513300** | 3 | 3 | 1 | 902 | 840 | 1632.0 | 1557 | |
| **1041586** | 6 | 9 | 1 | 1803 | 1755 | 2158.0 | 2120 | |
| **1015065** | 6 | 25 | 3 | 2054 | 2000 | 2329.0 | 2240 | |
| **167302** | 1 | 24 | 4 | 2002 | 1800 | 2218.0 | 2038 | |
| **293372** | 2 | 11 | 1 | 1519 | 1510 | 1824.0 | 1814 | |
| **436973** | 3 | 5 | 3 | 1903 | 1745 | 2021.0 | 1915 | |
| **925255** | 6 | 16 | 1 | 2221 | 1940 | 210.0 | 2347 | |
| **966604** | 6 | 30 | 1 | 2245 | 2037 | 211.0 | 2359 | |
| **413825** | 3 | 24 | 1 | 845 | 835 | 1000.0 | 952 | |
| **229520** | 2 | 3 | 7 | 1915 | 1616 | 2055.0 | 1750 | |
| **21440** | 1 | 23 | 3 | 1705 | 1650 | 1825.0 | 1810 | |
| **117583** | 1 | 29 | 2 | 1547 | 1505 | 1644.0 | 1603 | |
| **73349** | 1 | 3 | 4 | 1655 | 1407 | 1953.0 | 1728 | |
| **371403** | 2 | 8 | 5 | 1537 | 1520 | 1722.0 | 1713 | |
| **836489** | 5 | 1 | 4 | 1931 | 1924 | 2202.0 | 2154 | |
| **491263** | 3 | 6 | 4 | 1539 | 1530 | 1706.0 | 1654 | |
| **791624** | 5 | 22 | 4 | 1448 | 1341 | 1745.0 | 1636 | |
| **470924** | 3 | 1 | 6 | 1840 | 1815 | 1944.0 | 1920 | |
| **491755** | 3 | 7 | 5 | 1527 | 1409 | 1736.0 | 1635 | |
| **128037** | 1 | 9 | 3 | 1129 | 1110 | 1404.0 | 1400 | |

629145 rows × 21 columns

In [85]:

```
y_train
```

Out[85]:

```
109881       18.0
893552       68.0
695917        9.0
652905      118.0
666397       14.0
394973       -1.0
599286       45.0
746605       69.0
827928       96.0
176314       42.0
788152       72.0
193069       -3.0
817926       15.0
562633        1.0
496269       25.0
692581      141.0
242349       46.0
989277       61.0
565880       -1.0
301455        5.0
445079       26.0
725979       64.0
551345        7.0
94293        49.0
991032       39.0
411125       43.0
98609        34.0
704189       12.0
335617       44.0
523592       -2.0
            ...
1005966       7.0
188317        9.0
365212       35.0
806378      226.0
401660       29.0
457611      104.0
575956        4.0
691090       37.0
176485       32.0
21758       -10.0
513300       35.0
1041586      38.0
1015065      49.0
167302      100.0
293372       10.0
436973       66.0
925255      143.0
966604      132.0
413825        8.0
229520      185.0
21440        15.0
117583       41.0
73349       145.0
371403        9.0
836489        8.0
491263       12.0
791624       69.0
470924       24.0
```

```
491755      61.0
128037       4.0
Name: ArrDelay, Length: 629145, dtype: float64
```

In [ ]:

In [86]:

```
model=sm.OLS(y,X).fit()
predictions=model.predict(X)
model.summary()
```

Out[86]:

OLS Regression Results

| | | | |
|---|---|---|---|
| **Dep. Variable:** | ArrDelay | **R-squared:** | 0.991 |
| **Model:** | OLS | **Adj. R-squared:** | 0.991 |
| **Method:** | Least Squares | **F-statistic:** | 5.790e+06 |
| **Date:** | Thu, 06 Jun 2019 | **Prob (F-statistic):** | 0.00 |
| **Time:** | 12:54:50 | **Log-Likelihood:** | -3.4693e+06 |
| **No. Observations:** | 1048575 | **AIC:** | 6.939e+06 |
| **Df Residuals:** | 1048555 | **BIC:** | 6.939e+06 |
| **Df Model:** | 20 | | |
| **Covariance Type:** | nonrobust | | |

| | **coef** | **std err** | **t** | **P>\|t\|** | **[0.025** | **0.975]** |
|---|---|---|---|---|---|---|
| **Month** | -0.0394 | 0.004 | -10.863 | 0.000 | -0.047 | -0.032 |
| **DayofMonth** | -0.0099 | 0.001 | -14.102 | 0.000 | -0.011 | -0.009 |
| **DayOfWeek** | -0.0466 | 0.003 | -14.950 | 0.000 | -0.053 | -0.041 |
| **DepTime** | 0.0002 | 3.33e-05 | 6.169 | 0.000 | 0.000 | 0.000 |
| **CRSDepTime** | -0.0005 | 3.41e-05 | -13.625 | 0.000 | -0.001 | -0.000 |
| **ArrTime** | -0.0003 | 1.53e-05 | -16.621 | 0.000 | -0.000 | -0.000 |
| **CRSArrTime** | -5.325e-06 | 2.37e-05 | -0.225 | 0.822 | -5.18e-05 | 4.12e-05 |
| **FlightNum** | -6.847e-05 | 3.5e-06 | -19.550 | 0.000 | -7.53e-05 | -6.16e-05 |
| **ActualElapsedTime** | 0.7399 | 0.005 | 135.564 | 0.000 | 0.729 | 0.751 |
| **CRSElapsedTime** | -0.8561 | 0.001 | -1407.871 | 0.000 | -0.857 | -0.855 |
| **AirTime** | 0.0706 | 0.005 | 12.841 | 0.000 | 0.060 | 0.081 |
| **DepDelay** | 0.9832 | 0.000 | 6097.348 | 0.000 | 0.983 | 0.983 |
| **Distance** | 0.0051 | 6.28e-05 | 80.525 | 0.000 | 0.005 | 0.005 |
| **TaxiIn** | 0.2097 | 0.006 | 37.284 | 0.000 | 0.199 | 0.221 |
| **TaxiOut** | 0.2218 | 0.005 | 40.802 | 0.000 | 0.211 | 0.232 |
| **Cancelled** | -9.197e-16 | 1.64e-17 | -56.136 | 0.000 | -9.52e-16 | -8.88e-16 |
| **Diverted** | -5.1679 | 0.108 | -47.675 | 0.000 | -5.380 | -4.955 |
| **CarrierDelay** | 0.0111 | 0.000 | 49.095 | 0.000 | 0.011 | 0.012 |
| **WeatherDelay** | 0.0156 | 0.000 | 38.281 | 0.000 | 0.015 | 0.016 |
| **NASDelay** | 0.0271 | 0.000 | 92.555 | 0.000 | 0.027 | 0.028 |
| **SecurityDelay** | 0.0095 | 0.004 | 2.374 | 0.018 | 0.002 | 0.017 |

| | | | |
|---|---|---|---|
| **Omnibus:** | 2083333.030 | **Durbin-Watson:** | 1.915 |
| **Prob(Omnibus):** | 0.000 | **Jarque-Bera (JB):** | 237612370350.050 |
| **Skew:** | -14.672 | **Prob(JB):** | 0.00 |
| **Kurtosis:** | 2334.879 | **Cond. No.** | 1.06e+19 |

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.57e-25. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

In [ ]:

```

```

In [87]:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y,test_size = 0.4, random_state=
1)
```

In [88]:

```
X_train
```

Out[88]:

| | Month | DayofMonth | DayOfWeek | DepTime | CRSDepTime | ArrTime | CRSArrTime | Fli |
|---|---|---|---|---|---|---|---|---|
| **109881** | 1 | 19 | 6 | 2009 | 1945 | 2114.0 | 2056 | |
| **893552** | 6 | 11 | 3 | 1414 | 1340 | 1538.0 | 1430 | |
| **695917** | 4 | 27 | 7 | 1512 | 1455 | 1634.0 | 1625 | |
| **652905** | 4 | 28 | 1 | 2003 | 1745 | 2203.0 | 2005 | |
| **666397** | 4 | 28 | 1 | 1030 | 1016 | 1154.0 | 1140 | |
| **394973** | 3 | 19 | 3 | 1053 | 1040 | 1324.0 | 1325 | |
| **599286** | 4 | 29 | 2 | 1227 | 1125 | 1400.0 | 1315 | |
| **746605** | 5 | 20 | 2 | 2025 | 1910 | 2114.0 | 2005 | |
| **827928** | 5 | 14 | 3 | 1413 | 1250 | 1631.0 | 1455 | |
| **176314** | 1 | 22 | 2 | 1259 | 1235 | 1603.0 | 1521 | |
| **788152** | 5 | 20 | 2 | 1649 | 1525 | 1932.0 | 1820 | |
| **193069** | 2 | 9 | 6 | 1656 | 1650 | 2057.0 | 2100 | |
| **817926** | 5 | 26 | 1 | 1332 | 1314 | 1615.0 | 1600 | |
| **562633** | 3 | 17 | 1 | 1104 | 1030 | 1503.0 | 1502 | |
| **496269** | 3 | 24 | 1 | 935 | 915 | 1045.0 | 1020 | |
| **692581** | 4 | 21 | 1 | 2257 | 2105 | 741.0 | 520 | |
| **242349** | 2 | 4 | 1 | 1144 | 1115 | 1316.0 | 1230 | |
| **989277** | 6 | 29 | 7 | 1811 | 1749 | 1949.0 | 1848 | |
| **565880** | 3 | 22 | 6 | 1155 | 1145 | 1500.0 | 1501 | |
| **301455** | 2 | 25 | 1 | 1719 | 1710 | 1834.0 | 1829 | |
| **445079** | 3 | 26 | 3 | 1510 | 1455 | 1608.0 | 1542 | |
| **725979** | 4 | 6 | 7 | 1829 | 1700 | 2132.0 | 2028 | |
| **551345** | 3 | 5 | 3 | 1936 | 1910 | 2157.0 | 2150 | |
| **94293** | 1 | 2 | 3 | 1701 | 1605 | 1744.0 | 1655 | |
| **991032** | 6 | 30 | 1 | 2204 | 2118 | 2155.0 | 2116 | |
| **411125** | 3 | 31 | 1 | 1828 | 1745 | 1843.0 | 1800 | |
| **98609** | 1 | 24 | 4 | 1137 | 1115 | 1329.0 | 1255 | |
| **704189** | 4 | 17 | 4 | 1959 | 1929 | 2241.0 | 2229 | |
| **335617** | 2 | 1 | 5 | 1829 | 1755 | 2034.0 | 1950 | |
| **523592** | 3 | 20 | 4 | 1242 | 1219 | 1248.0 | 1250 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | |
| **1005966** | 6 | 13 | 5 | 1335 | 1325 | 2152.0 | 2145 | |
| **188317** | 2 | 6 | 3 | 1410 | 1355 | 1649.0 | 1640 | |
| **365212** | 2 | 14 | 4 | 1847 | 1800 | 2149.0 | 2114 | |
| **806378** | 5 | 1 | 4 | 1931 | 1610 | 206.0 | 2220 | |
| **401660** | 3 | 24 | 1 | 2156 | 2125 | 2259.0 | 2230 | |
| **457611** | 3 | 25 | 2 | 1923 | 1743 | 2057.0 | 1913 | |

| | Month | DayofMonth | DayOfWeek | DepTime | CRSDepTime | ArrTime | CRSArrTime | Fli |
|---|---|---|---|---|---|---|---|---|
| **575956** | 4 | 4 | 5 | 919 | 900 | 1144.0 | 1140 | |
| **691090** | 4 | 8 | 2 | 1345 | 1325 | 1542.0 | 1505 | |
| **176485** | 1 | 1 | 2 | 1813 | 1725 | 1947.0 | 1915 | |
| **21758** | 1 | 24 | 4 | 856 | 850 | 1140.0 | 1150 | |
| **513300** | 3 | 3 | 1 | 902 | 840 | 1632.0 | 1557 | |
| **1041586** | 6 | 9 | 1 | 1803 | 1755 | 2158.0 | 2120 | |
| **1015065** | 6 | 25 | 3 | 2054 | 2000 | 2329.0 | 2240 | |
| **167302** | 1 | 24 | 4 | 2002 | 1800 | 2218.0 | 2038 | |
| **293372** | 2 | 11 | 1 | 1519 | 1510 | 1824.0 | 1814 | |
| **436973** | 3 | 5 | 3 | 1903 | 1745 | 2021.0 | 1915 | |
| **925255** | 6 | 16 | 1 | 2221 | 1940 | 210.0 | 2347 | |
| **966604** | 6 | 30 | 1 | 2245 | 2037 | 211.0 | 2359 | |
| **413825** | 3 | 24 | 1 | 845 | 835 | 1000.0 | 952 | |
| **229520** | 2 | 3 | 7 | 1915 | 1616 | 2055.0 | 1750 | |
| **21440** | 1 | 23 | 3 | 1705 | 1650 | 1825.0 | 1810 | |
| **117583** | 1 | 29 | 2 | 1547 | 1505 | 1644.0 | 1603 | |
| **73349** | 1 | 3 | 4 | 1655 | 1407 | 1953.0 | 1728 | |
| **371403** | 2 | 8 | 5 | 1537 | 1520 | 1722.0 | 1713 | |
| **836489** | 5 | 1 | 4 | 1931 | 1924 | 2202.0 | 2154 | |
| **491263** | 3 | 6 | 4 | 1539 | 1530 | 1706.0 | 1654 | |
| **791624** | 5 | 22 | 4 | 1448 | 1341 | 1745.0 | 1636 | |
| **470924** | 3 | 1 | 6 | 1840 | 1815 | 1944.0 | 1920 | |
| **491755** | 3 | 7 | 5 | 1527 | 1409 | 1736.0 | 1635 | |
| **128037** | 1 | 9 | 3 | 1129 | 1110 | 1404.0 | 1400 | |

629145 rows × 21 columns

In [89]:

```
y_train
```

Out[89]:

```
109881      18.0
893552      68.0
695917       9.0
652905     118.0
666397      14.0
394973      -1.0
599286      45.0
746605      69.0
827928      96.0
176314      42.0
788152      72.0
193069      -3.0
817926      15.0
562633       1.0
496269      25.0
692581     141.0
242349      46.0
989277      61.0
565880      -1.0
301455       5.0
445079      26.0
725979      64.0
551345       7.0
94293       49.0
991032      39.0
411125      43.0
98609       34.0
704189      12.0
335617      44.0
523592      -2.0
            ... 
1005966      7.0
188317       9.0
365212      35.0
806378     226.0
401660      29.0
457611     104.0
575956       4.0
691090      37.0
176485      32.0
21758      -10.0
513300      35.0
1041586     38.0
1015065     49.0
167302     100.0
293372      10.0
436973      66.0
925255     143.0
966604     132.0
413825       8.0
229520     185.0
21440       15.0
117583      41.0
73349      145.0
371403       9.0
836489       8.0
491263      12.0
791624      69.0
470924      24.0
```

```
491755      61.0
128037       4.0
Name: ArrDelay, Length: 629145, dtype: float64
```

In [90]:

```
features = X_train.iloc[:,:].values
```

In [91]:

```
features
```

Out[91]:

```
array([[ 1.        , 19.        ,  6.        , ...,  0.        ,
         3.        ,  0.        ],
       [ 6.        , 11.        ,  3.        , ...,  0.        ,
        34.        ,  0.        ],
       [ 4.        , 27.        ,  7.        , ...,  3.56803221,
        14.42961848,  0.09328398],
       ...,
       [ 3.        ,  1.        ,  6.        , ...,  0.        ,
         0.        ,  0.        ],
       [ 3.        ,  7.        ,  5.        , ...,  0.        ,
         0.        ,  0.        ],
       [ 1.        ,  9.        ,  3.        , ...,  3.56803221,
        14.42961848,  0.09328398]])
```

In [92]:

```
labels = y_train.iloc[:].values
```

In [93]:

```
labels
```

Out[93]:

```
array([18., 68.,  9., ..., 24., 61.,  4.])
```

In [94]:

```
X=features
y=labels
```

In [95]:

```
from sklearn import linear_model as lm
model=lm.LinearRegression()
results=model.fit(X,y)
```

In [96]:

```
predictions = model.predict(X)
```

In [97]:

```
accuracy=model.score(X,y)
print('Accuracy of the model:', accuracy)
```

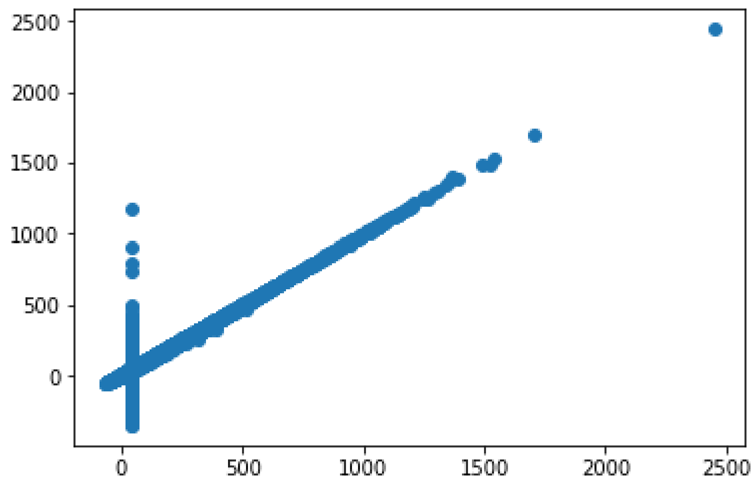
```
Accuracy of the model: 0.9856400206064125
```

In [98]:

```
plt.scatter(y, predictions)
```

Out[98]:

```
<matplotlib.collections.PathCollection at 0x21af94687b8>
```



In [99]:

```
from sklearn.metrics import mean_squared_error, r2_score

# printing values
print('Slope:' ,model.coef_)
print('Intercept:', model.intercept_)
print("\n")


import numpy as np
rmse = (np.sqrt(mean_squared_error(y,predictions)))
r2 = r2_score(y,predictions)

print("The model performance")
print("--------------------------------------")
print('RMSE is {}'.format(rmse))
print('R2 score is {}'.format(r2))
print("\n")
```

```
Slope: [ 1.52942511e-02 -1.44229489e-04  4.76216053e-03  2.19054167e-04
 -1.55594650e-04 -1.08558344e-04  2.35548098e-05 -2.65839596e-05
  7.99539145e-01 -8.42255305e-01  5.46684864e-03  9.82670871e-01
  4.20112805e-03  1.58906546e-01  1.63236749e-01  7.93809463e-15
 -4.72323860e+00  1.26105335e-02  1.66531795e-02  2.77630317e-02
  1.36361935e-02]
Intercept: -2.129090996527509


The model performance
--------------------------------------
RMSE is 6.666210639637911
R2 score is 0.9856400206064125
```

In [ ]: