weather-analysis

November 23, 2024

1 The Weather Analysis

[1]:
```
import pandas as pd
```

[2]:
```
df = pd.read_csv(r"C:\Users\kiran\Downloads\weather_dataset.csv")
```

[3]:
```
df.head()
```

[3]:

	Date/Time	Temp_C	Dew Point Temp_C	Rel Hum_%	Wind Speed_km/h	Visibility_km	Press_kPa	Weather
0	1/1/2012 0:00	-1.8	-3.9	86	4	8.0	101.24	Fog
1	1/1/2012 1:00	-1.8	-3.7	87	4	8.0	101.24	Fog
2	1/1/2012 2:00	-1.8	-3.4	89	7	4.0	101.26	Freezing Drizzle,Fog
3	1/1/2012 3:00	-1.5	-3.2	88	6	4.0	101.27	Freezing Drizzle,Fog
4	1/1/2012 4:00	-1.5	-3.3	88	7	4.8	101.23	Fog

2 Explore the data

[5]:
```
df.head() # top 5 data
```

[5]:

	Date/Time	Temp_C	Dew Point Temp_C	Rel Hum_%	Wind Speed_km/h	Visibility_km	Press_kPa	Weather
0	1/1/2012 0:00	-1.8	-3.9	86	4	8.0	101.24	Fog
1	1/1/2012 1:00	-1.8	-3.7	87	4	8.0	101.24	Fog
2	1/1/2012 2:00	-1.8	-3.4	89	7	4.0	101.26	Freezing Drizzle,Fog
3	1/1/2012 3:00	-1.5	-3.2	88	6	4.0	101.27	Freezing Drizzle,Fog
4	1/1/2012 4:00	-1.5	-3.3	88	7	4.8	101.23	Fog

```
2            4.0     101.26  Freezing Drizzle,Fog
3            4.0     101.27  Freezing Drizzle,Fog
4            4.8     101.23                   Fog
```

[6]:
```
df.tail() # bottom 5 data
```

[6]:
```
             Date/Time  Temp_C  Dew Point Temp_C  Rel Hum_%  Wind Speed_km/h  \
8779  12/31/2012 19:00     0.1              -2.7         81               30
8780  12/31/2012 20:00     0.2              -2.4         83               24
8781  12/31/2012 21:00    -0.5              -1.5         93               28
8782  12/31/2012 22:00    -0.2              -1.8         89               28
8783  12/31/2012 23:00     0.0              -2.1         86               30

      Visibility_km  Press_kPa Weather
8779            9.7     100.13    Snow
8780            9.7     100.03    Snow
8781            4.8      99.95    Snow
8782            9.7      99.91    Snow
8783           11.3      99.89    Snow
```

[7]:
```
print(f"No of rows: {df.shape[0]}")
print(f"No of columns:{df.shape[1]}")
```

```
No of rows: 8784
No of columns:8
```

column names

[15]:
```
print("Dataset Column/Variable Names :")
print("_"*24)
c = 1
for i in df.columns:
    print(f"{str(c)} --> {i}")
    c = c +1
```

```
Dataset Column/Variable Names :

------------------------
1 --> Date/Time
2 --> Temp_C
3 --> Dew Point Temp_C
4 --> Rel Hum_%
5 --> Wind Speed_km/h
6 --> Visibility_km
7 --> Press_kPa
8 --> Weather
```

[17]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 8784 entries, 0 to 8783
Data columns (total 8 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Date/Time         8784 non-null   object
 1   Temp_C            8784 non-null   float64
 2   Dew Point Temp_C  8784 non-null   float64
 3   Rel Hum_%         8784 non-null   int64
 4   Wind Speed_km/h   8784 non-null   int64
 5   Visibility_km     8784 non-null   float64
 6   Press_kPa         8784 non-null   float64
 7   Weather           8784 non-null   object
dtypes: float64(4), int64(2), object(2)
memory usage: 549.1+ KB
```

[19]:
```
df.dtypes
```

[19]:
```
Date/Time            object
Temp_C              float64
Dew Point Temp_C    float64
Rel Hum_%             int64
Wind Speed_km/h       int64
Visibility_km       float64
Press_kPa           float64
Weather              object
dtype: object
```

[21]:
```
# i want to check number of unique weathers

df.Weather.unique()
```

[21]:
```
array(['Fog', 'Freezing Drizzle,Fog', 'Mostly Cloudy', 'Cloudy', 'Rain',
       'Rain Showers', 'Mainly Clear', 'Snow Showers', 'Snow', 'Clear',
       'Freezing Rain,Fog', 'Freezing Rain', 'Freezing Drizzle',
       'Rain,Snow', 'Moderate Snow', 'Freezing Drizzle,Snow',
       'Freezing Rain,Snow Grains', 'Snow,Blowing Snow', 'Freezing Fog',
       'Haze', 'Rain,Fog', 'Drizzle,Fog', 'Drizzle',
       'Freezing Drizzle,Haze', 'Freezing Rain,Haze', 'Snow,Haze',
       'Snow,Fog', 'Snow,Ice Pellets', 'Rain,Haze', 'Thunderstorms,Rain',
       'Thunderstorms,Rain Showers', 'Thunderstorms,Heavy Rain Showers',
       'Thunderstorms,Rain Showers,Fog', 'Thunderstorms',
       'Thunderstorms,Rain,Fog',
       'Thunderstorms,Moderate Rain Showers,Fog', 'Rain Showers,Fog',
       'Rain Showers,Snow Showers', 'Snow Pellets', 'Rain,Snow,Fog',
       'Moderate Rain,Fog', 'Freezing Rain,Ice Pellets,Fog',
       'Drizzle,Ice Pellets,Fog', 'Drizzle,Snow', 'Rain,Ice Pellets',
       'Drizzle,Snow,Fog', 'Rain,Snow Grains', 'Rain,Snow,Ice Pellets',
```

```
       'Snow Showers,Fog', 'Moderate Snow,Blowing Snow'], dtype=object)
```

[23]:
```
df.Weather.nunique() # no of unique values from this variable
```

[23]:
```
50
```

[25]:
```
#overall dataset unique values
df.nunique()
```

[25]:
```
Date/Time           8784
Temp_C               533
Dew Point Temp_C     489
Rel Hum_%             83
Wind Speed_km/h       34
Visibility_km         24
Press_kPa            518
Weather               50
dtype: int64
```

[27]:
```
#let me check there is any NA values
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8784 entries, 0 to 8783
Data columns (total 8 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Date/Time         8784 non-null   object
 1   Temp_C            8784 non-null   float64
 2   Dew Point Temp_C  8784 non-null   float64
 3   Rel Hum_%         8784 non-null   int64
 4   Wind Speed_km/h   8784 non-null   int64
 5   Visibility_km     8784 non-null   float64
 6   Press_kPa         8784 non-null   float64
 7   Weather           8784 non-null   object
dtypes: float64(4), int64(2), object(2)
memory usage: 549.1+ KB
```

[29]:
```
#here it shows no null values but still let me use df.na :)

df.isna().sum()
```

[29]:
```
Date/Time           0
Temp_C              0
Dew Point Temp_C    0
Rel Hum_%           0
Wind Speed_km/h     0
```

```
Visibility_km       0
Press_kPa           0
Weather             0
dtype: int64
```

now i'm sure there is no null

[32]:
```
df['Date/Time'] = pd.to_datetime(df['Date/Time'])
```

3 Q) 1. Find all the unique 'Wind Speed' values in the data.

[35]:
```
# df.head()
```

[37]:
```
df['Wind Speed_km/h'].unique()
```

[37]:
```
array([ 4,  7,  6,  9, 15, 13, 20, 22, 19, 24, 30, 35, 39, 32, 33, 26, 44,
       43, 48, 37, 28, 17, 11,  0, 83, 70, 57, 46, 41, 52, 50, 63, 54,  2],
      dtype=int64)
```

4 Q) 2. Find the number of times when the 'Weather is exactly Clear'.

[40]:
```
# df.head()
# df.Weather.value_counts().head(5)
df[df.Weather == 'Clear']['Weather'].count()
```

[40]:
```
1326
```

[42]:
```
df.groupby('Weather').get_group('Clear')['Weather'].count()
```

[42]:
```
1326
```

5 Q) 3. Find the number of times when the 'Wind Speed was exactly 6 km/h'.

[45]:
```
# df.head()
x = df[df['Wind Speed_km/h'] == 6]['Wind Speed_km/h'].count()

print(f"The No of times the 'Wind Speed' for 6 km/h is : {x} times.")
```

```
The No of times the 'Wind Speed' for 6 km/h is : 609 times.
```

6 Q. 4) Find out all the Null Values in the data.

[48]:
```
df.isna().sum()
#there is no null values in dataset
```

[48]:
```
Date/Time           0
Temp_C              0
Dew Point Temp_C    0
Rel Hum_%           0
Wind Speed_km/h     0
Visibility_km       0
Press_kPa           0
Weather             0
dtype: int64
```

[ ]:

7 Q. 5) Rename the column name 'Weather' of the dataframe to 'Weather Condition'.

[52]:
```
df = df.rename(columns = {'Weather' : 'Weather_Condition'})
df.head()
```

[52]:

	Date/Time	Temp_C	Dew Point Temp_C	Rel Hum_%	Wind Speed_km/h	Visibility_km	Press_kPa	Weather_Condition
0	2012-01-01 00:00:00	-1.8	-3.9	86	4	8.0	101.24	Fog
1	2012-01-01 01:00:00	-1.8	-3.7	87	4	8.0	101.24	Fog
2	2012-01-01 02:00:00	-1.8	-3.4	89	7	4.0	101.26	Freezing Drizzle,Fog
3	2012-01-01 03:00:00	-1.5	-3.2	88	6	4.0	101.27	Freezing Drizzle,Fog
4	2012-01-01 04:00:00	-1.5	-3.3	88	7	4.8	101.23	Fog

8 Q.6) What is the mean 'Visibility' ?

[55]:
```
print(f'The Mean/Average Visibility is {df.Visibility_km.mean().round(2)}')
```

```
The Mean/Average Visibility is 27.66
```

[ ]:

9 Q. 7) What is the Standard Deviation of 'Pressure' in this data?

[59]:
```
print(f"The Standard Deviation of Pressure is {df.Press_kPa.std()}")
```

```
The Standard Deviation of Pressure is 0.8440047459486474
```

10 Q. 8) Whats is the Variance of 'Relative Humidity' in this data ?

[62]:
```
print(f"The Variance of 'Relative Humidity is {df['Rel Hum_%'].var()}")
```

```
The Variance of 'Relative Humidity is 286.2485501984998
```

11 Q. 9) Find all instances when 'Snow' was recorded.

[77]:
```
# df[df.Weather_Condition == 'Snow']
```

[75]:
```
df[df.Weather_Condition.str.contains('Snow')].head(10)
```

[75]:

	Date/Time	Temp_C	Dew Point Temp_C	Rel Hum_%	Wind Speed_km/h	Visibility_km	Press_kPa	Weather_Condition
41	2012-01-02 17:00:00	-2.1	-9.5	57	22	25.0	99.66	Snow Showers
44	2012-01-02 20:00:00	-5.6	-13.4	54	24	25.0	100.07	Snow Showers
45	2012-01-02 21:00:00	-5.8	-12.8	58	26	25.0	100.15	Snow Showers
47	2012-01-02 23:00:00	-7.4	-14.1	59	17	19.3	100.27	Snow Showers
48	2012-01-03 00:00:00	-9.0	-16.0	57	28	25.0	100.35	Snow Showers
50	2012-01-03 02:00:00	-10.5	-15.8	65	22	12.9	100.53	Snow Showers
51	2012-01-03 03:00:00	-11.3	-18.7	54	33	25.0	100.61	Snow Showers
53	2012-01-03 05:00:00	-12.9	-19.1	60	22	25.0	100.76	Snow Showers
54	2012-01-03 06:00:00	-13.3	-19.3	61	19	25.0	100.85	Snow Showers
55	2012-01-03 07:00:00	-14.0	-19.5	63	19	25.0	100.95	Snow

[ ]:
```

```

12 Q. 10) Find all instances when 'Wind Speed is above 24' and 'Visibility is 25'.

[81]:
```
df[(df['Wind Speed_km/h'] > 24) & (df['Visibility_km'] ==25)].head(10)
```

[81]:

	Date/Time	Temp_C	Dew Point Temp_C	Rel Hum_%	Wind Speed_km/h	\
23	2012-01-01 23:00:00	5.3	2.0	79	30	
24	2012-01-02 00:00:00	5.2	1.5	77	35	
25	2012-01-02 01:00:00	4.6	0.0	72	39	
26	2012-01-02 02:00:00	3.9	-0.9	71	32	
27	2012-01-02 03:00:00	3.7	-1.5	69	33	
28	2012-01-02 04:00:00	2.9	-2.3	69	32	
29	2012-01-02 05:00:00	2.6	-2.3	70	32	
30	2012-01-02 06:00:00	2.3	-2.6	70	26	
31	2012-01-02 07:00:00	2.0	-2.9	70	33	
42	2012-01-02 18:00:00	-4.1	-11.4	57	28	

	Visibility_km	Press_kPa	Weather_Condition
23	25.0	99.31	Cloudy
24	25.0	99.26	Rain Showers
25	25.0	99.26	Cloudy
26	25.0	99.26	Mostly Cloudy
27	25.0	99.30	Mostly Cloudy
28	25.0	99.26	Mostly Cloudy
29	25.0	99.21	Mostly Cloudy
30	25.0	99.18	Mostly Cloudy
31	25.0	99.14	Mostly Cloudy
42	25.0	99.86	Mostly Cloudy

[ ]:
```
```

13 Q. 11) What is the Mean value of each column against each 'Weather Conditon' ?

[83]:
```
df.groupby('Weather_Condition').mean().head(10)
```

[83]:

Weather_Condition	Date/Time	Temp_C	\
Clear	2012-06-25 18:03:12.760180992	6.825716	
Cloudy	2012-07-01 16:51:29.583333376	7.970544	
Drizzle	2012-09-17 15:00:00.000000000	7.353659	
Drizzle,Fog	2012-06-30 00:44:15.000000000	8.067500	
Drizzle,Ice Pellets,Fog	2012-12-17 09:00:00.000000000	0.400000	
Drizzle,Snow	2012-12-18 16:30:00.000000000	1.050000	
Drizzle,Snow,Fog	2012-12-19 20:56:00.000000000	0.693333	
Fog	2012-06-24 21:08:00.000000000	4.303333	

```
Freezing Drizzle        2012-03-05 23:25:42.857142784 -5.657143
Freezing Drizzle,Fog    2012-04-30 06:20:00.000000000 -2.533333

                         Dew Point Temp_C  Rel Hum_%  Wind Speed_km/h  \
Weather_Condition
Clear                            0.089367  64.497738        10.557315
Cloudy                           2.375810  69.592593        16.127315
Drizzle                          5.504878  88.243902        16.097561
Drizzle,Fog                      7.033750  93.275000        11.862500
Drizzle,Ice Pellets,Fog         -0.700000  92.000000        20.000000
Drizzle,Snow                     0.150000  93.500000        14.000000
Drizzle,Snow,Fog                 0.120000  95.866667        15.533333
Fog                              3.159333  92.286667         7.946667
Freezing Drizzle                -8.000000  83.571429        16.571429
Freezing Drizzle,Fog            -4.183333  88.500000        17.000000

                         Visibility_km   Press_kPa
Weather_Condition
Clear                        30.153243  101.587443
Cloudy                       26.625752  100.911441
Drizzle                      17.931707  100.435366
Drizzle,Fog                   5.257500  100.786625
Drizzle,Ice Pellets,Fog       4.000000  100.790000
Drizzle,Snow                 10.500000  100.890000
Drizzle,Snow,Fog              5.513333   99.281333
Fog                           6.248000  101.184067
Freezing Drizzle              9.200000  100.202857
Freezing Drizzle,Fog          5.266667  100.441667
```

[ ]:

14 Q. 12) Show all the Records where Weather Condition is Fog.

[85]:
```
df[df.Weather_Condition == 'Fog'].head()
```

[85]:
```
             Date/Time  Temp_C  Dew Point Temp_C  Rel Hum_%  Wind Speed_km/h  \
0 2012-01-01 00:00:00    -1.8              -3.9         86                4
1 2012-01-01 01:00:00    -1.8              -3.7         87                4
4 2012-01-01 04:00:00    -1.5              -3.3         88                7
5 2012-01-01 05:00:00    -1.4              -3.3         87                9
6 2012-01-01 06:00:00    -1.5              -3.1         89                7

   Visibility_km  Press_kPa Weather_Condition
0            8.0     101.24               Fog
1            8.0     101.24               Fog
4            4.8     101.23               Fog
```

```
5            6.4     101.27              Fog
6            6.4     101.29              Fog
```

15 Q. 13) Find all instances when 'Weather is Clear' or 'Visibility is above 40'.

[87]:
```
df[(df.Weather_Condition == 'Clear') | (df.Visibility_km >40)].head()
```

[87]:
```
              Date/Time  Temp_C  Dew Point Temp_C  Rel Hum_%  Wind Speed_km/h  \
67  2012-01-03 19:00:00   -16.9             -24.8         50               24
106 2012-01-05 10:00:00    -6.0             -10.0         73               17
107 2012-01-05 11:00:00    -5.6             -10.2         70               22
108 2012-01-05 12:00:00    -4.7              -9.6         69               20
109 2012-01-05 13:00:00    -4.4              -9.7         66               26

     Visibility_km  Press_kPa Weather_Condition
67            25.0     101.74             Clear
106           48.3     100.45      Mainly Clear
107           48.3     100.41      Mainly Clear
108           48.3     100.38      Mainly Clear
109           48.3     100.40      Mainly Clear
```

16 Q. 14) Find all instances when :

16.0.1 A. 'Weather is Clear' and 'Relative Humidity is greater than 50'

16.0.2 or

16.0.3 B. 'Visibility is above 40'

[89]:
```
df[((df.Weather_Condition == 'Clear') & (df['Rel Hum_%'] >50))| (df.
↪Visibility_km >40)].head()
```

[89]:
```
              Date/Time  Temp_C  Dew Point Temp_C  Rel Hum_%  Wind Speed_km/h  \
106 2012-01-05 10:00:00    -6.0             -10.0         73               17
107 2012-01-05 11:00:00    -5.6             -10.2         70               22
108 2012-01-05 12:00:00    -4.7              -9.6         69               20
109 2012-01-05 13:00:00    -4.4              -9.7         66               26
110 2012-01-05 14:00:00    -5.1             -10.7         65               22

     Visibility_km  Press_kPa Weather_Condition
106           48.3     100.45      Mainly Clear
107           48.3     100.41      Mainly Clear
108           48.3     100.38      Mainly Clear
109           48.3     100.40      Mainly Clear
110           48.3     100.46      Mainly Clear
```

16.1 Pandas Practice Done :)

- reason i used head() function all line because when i try to convert this jupyter file to Pdf it looks messy with those output

[ ]: