Weather Data Analysis (Part of Big Data Analysis) The Weather Dataset here is a time series dataset with per hour information about the weather conditions at a particular location. It records Temperature, Relative Humidity, Wind Speed, Visibility, Pressure and Conditions In [6]: # IMPORTING A FILE import pandas as pd df = pd.read\_csv("C:\\Users\\Administrator\\OneDrive\\Documents\\Weather data.csv") 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 86 101.24 8.0 Fog -3.7 1/1/2012 1:00 -1.8 101.24 Fog 1/1/2012 2:00 -1.8 -3.4 89 4.0 101.26 Freezing Drizzle,Fog 1/1/2012 3:00 -1.5 -3.2 101.27 Freezing Drizzle,Fog 1/1/2012 4:00 -1.5 -3.3 88 4.8 101.23 Fog **8779** 12/31/2012 19:00 0.1 -2.7 81 30 100.13 9.7 Snow **8780** 12/31/2012 20:00 0.2 -2.4 83 24 100.03 9.7 Snow **8781** 12/31/2012 21:00 -0.5 -1.5 93 28 99.95 Snow 4.8 **8782** 12/31/2012 22:00 -1.8 99.91 Snow **8783** 12/31/2012 23:00 0.0 -2.1 86 30 99.89 11.3 Snow 8784 rows × 8 columns In [12]: # ANALYZING THE DATA # 1) HEAD() = shows the first n rows in a data. here n = 10print(df.head(10)) Date/Time Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h \ 0 1/1/2012 0:00 -1.8 -3.9 86 1 1/1/2012 1:00 -1.8 2 1/1/2012 2:00 -1.8 -3.4 89 3 1/1/2012 3:00 -1.5 -3.2 88 88 4 1/1/2012 4:00 -1.5 -3.3 87 5 1/1/2012 5:00 -1.4 -3.3 6 1/1/2012 6:00 -1.5 -3.1 89 7 1/1/2012 7:00 -1.4 -3.6 85 85 9 8 1/1/2012 8:00 -1.4 -3.6 15 9 1/1/2012 9:00 -1.3 Visibility\_km Press\_kPa Weather 8.0 101.24 8.0 101.24 Fog 4.0 101.26 Freezing Drizzle, Fog 4.0 101.27 Freezing Drizzle, Fog 4.8 101.23 6.4 101.27 Fog 101.29 6.4 Fog 101.26 8.0 Fog 8.0 101.23 Fog In [16]: # 2) .SHAPE = shows the total number of rows and columns of the dataframe print(df.shape) (8784, 8) In [18]: # 3).INDEX = attributes provide the idex to the dataframe df.index Out[18]: RangeIndex(start=0, stop=8784, step=1) In [20]: # 4) .columns = shows the column name df.columns Out[20]: Index(['Date/Time', 'Temp\_C', 'Dew Point Temp\_C', 'Rel Hum\_%', 'Wind Speed\_km/h', 'Visibility\_km', 'Press\_kPa', 'Weather'], dtype='object') In [22]: # 5) .dtypes = shows the data types df.dtypes Out[22]: 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 In [22]: # 6) .dtypes = shows the data types df.dtypes Out[22]: Date/Time object Temp\_C float64 Dew Point Temp\_C Rel Hum\_% int64 Wind Speed\_km/h int64 Visibility\_km float64 Press\_kPa float64 Weather object dtype: object In [24]: # 7) .unique() = shows the unique values for a particular column df["Weather"].unique() Out [24]: 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) In [26]: # 8) .nunique() = shows the unique values foe each column. Can also apply on whole data frame df.nunique() Out[26]: Date/Time 8784 533  $Temp\_C$ Dew Point Temp\_C 489 Rel Hum\_% 83 34 Wind Speed\_km/h Visibility\_km 24 Press\_kPa 518 Weather 50 dtype: int64 In [28]: # 9) .count() = shows the total of non null values in each column. Can also apply on whole data frame df.count() 8784 Out[28]: Date/Time 8784  $Temp\_C$ Dew Point Temp\_C 8784 8784 Rel Hum\_% Wind Speed\_km/h 8784 Visibility\_km 8784 Press\_kPa 8784 Weather 8784 dtype: int64 In [30]: # 10) .value\_counts = shows the unique values with their count. Can only apply on a single column df.value\_counts("Press\_kPa") Out[30]: Press\_kPa 101.13 69 101.36 57 100.69 56 101.06 54 100.65 54 98.45 102.70 98.50 98.53 97.52 Name: count, Length: 518, dtype: int64 In [32]: # 11) .info() = provides teh basic info of the dataframe 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 In [40]: # 11) Finding the "Wind Speed" unique values in the data. df["Wind Speed\_km/h"].unique() 43, 48, 37, 28, 17, 11, 0, 83, 70, 57, 46, 41, 52, 50, 63, 54, 2], dtype=int64) In [42]: df["Wind Speed\_km/h"].nunique() Out[42]: 34 In [52]: # 11) Find the number of times when the weather is exactly "Clear". # a) First way df[df.Weather == "Clear"] Out[52]: Date/Time Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa Weather **67** 1/3/2012 19:00 -16.9 -24.8 Clear 25.0 101.74 **114** 1/5/2012 18:00 -7.1 -14.4 25.0 100.71 Clear 61 **115** 1/5/2012 19:00 -9.2 -15.4 25.0 Clear 100.80 **116** 1/5/2012 20:00 -9.8 -15.7 62 25.0 100.83 Clear 63 **117** 1/5/2012 21:00 -9.0 -14.8 13 25.0 100.83 Clear **8646** 12/26/2012 6:00 -14.8 89 102.47 Clear -13.4 25.0 **8698** 12/28/2012 10:00 -6.1 -8.6 82 101.27 Clear **8713** 12/29/2012 1:00 -11.9 -13.6 87 11 101.31 Clear 25.0 **8714** 12/29/2012 2:00 -11.8 -13.1 90 25.0 101.33 Clear **8756** 12/30/2012 20:00 -16.5 80 -13.8 24 25.0 101.52 Clear 1326 rows × 8 columns In [54]: # b) Second Way df.groupby("Weather").get\_group("Clear") Out[54]: Date/Time Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa Weather **67** 1/3/2012 19:00 -16.9 -24.8 50 25.0 101.74 Clear **114** 1/5/2012 18:00 -7.1 -14.4 56 11 25.0 100.71 Clear 61 **115** 1/5/2012 19:00 -9.2 -15.4 25.0 100.80 Clear **116** 1/5/2012 20:00 -15.7 62 25.0 100.83 Clear **117** 1/5/2012 21:00 -9.0 -14.8 63 13 25.0 100.83 Clear -14.8 89 **8646** 12/26/2012 6:00 -13.4 102.47 Clear 25.0 **8698** 12/28/2012 10:00 -6.1 -8.6 82 19 24.1 101.27 Clear -13.6 87 **8713** 12/29/2012 1:00 -11.9 25.0 101.31 Clear 11 **8714** 12/29/2012 2:00 -11.8 -13.1 13 25.0 101.33 Clear -16.5 80 24 **8756** 12/30/2012 20:00 -13.8 25.0 101.52 Clear 1326 rows × 8 columns In [64]: # 11) Find the number of times when the wind speed was exactly "4km/h". df.groupby("Wind Speed\_km/h").get\_group(4) Out[64]: Date/Time Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa Weather 1/1/2012 0:00 -1.8 -3.9 86 8.0 101.24 Fog -3.7 101.24 1/1/2012 1:00 Fog -8.8 -11.7 79 1/5/2012 0:00 9.7 100.32 Snow -7.0 -9.5 1/5/2012 5:00 100.19 Snow Cloudy -11.1 79 1/7/2012 2:00 -8.1 19.3 100.15 12/31/2012 8:00 -8.6 87 -10.3 3.2 101.14 Snow Showers **8769** 12/31/2012 9:00 -9.6 101.09 Snow **8770** 12/31/2012 10:00 -7.4 -8.9 89 6.4 101.05 Snow,Fog **8772** 12/31/2012 12:00 -7.5 12.9 100.78 Snow -6.6 86 **8773** 12/31/2012 13:00 -4.6 12.9 100.63 Snow 474 rows × 8 columns In [68]: df.isnull().sum() Out[68]: Date/Time  $Temp\_C$ Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa Weather dtype: int64 In [82]: # 12) Renaming a column "Weather" to "Weather condition" df.rename(columns = {"Weather":"Weather Condition"}, inplace = True) df.head(2) Out[82]: Date/Time Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa Weather Condition **0** 1/1/2012 0:00 -1.8 101.24 Fog 1 1/1/2012 1:00 -1.8 -3.7 8.0 101.24 Fog In [84]: # 13) What is the mean visibilty? df.Visibility\_km.mean() Out[84]: 27.664446721311478 In [86]: # 14) What is the standard deviation of pressure in this data? df.Press\_kPa.std() Out[86]: 0.8440047459486474 In [94]: # 15) What is the variance of relative humidity in this data? df["Rel Hum\_%"].var() Out[94]: 286.2485501984998 In [98]: # 16) Find all the instances when snow was recorded? df["Weather Condition"].value\_counts() Out[98]: Weather Condition 2106 Mainly Clear Mostly Cloudy 2069 1728 Cloudy 1326 Clear 390 Snow Rain 306 Rain Showers 188 150 Fog Rain,Fog 116 Drizzle, Fog 80 Snow Showers 60 Drizzle 41 37 Snow, Fog Snow, Blowing Snow 19 18 Rain, Snow 16 Thunderstorms, Rain Showers 16 Haze Drizzle, Snow, Fog 15 14 Freezing Rain 11 Freezing Drizzle, Snow Freezing Drizzle Snow, Ice Pellets Freezing Drizzle, Fog Snow, Haze Freezing Fog Snow Showers, Fog Moderate Snow Rain, Snow, Ice Pellets Freezing Rain, Fog Freezing Drizzle, Haze Rain, Haze Thunderstorms, Rain Thunderstorms, Rain Showers, Fog Freezing Rain, Haze Drizzle, Snow Rain Showers, Snow Showers Thunderstorms Moderate Snow, Blowing Snow Rain Showers, Fog Thunderstorms, Moderate Rain Showers, Fog Snow Pellets Rain, Snow, Fog Moderate Rain, Fog Freezing Rain, Ice Pellets, Fog Drizzle, Ice Pellets, Fog Thunderstorms, Rain, Fog Rain, Ice Pellets Rain, Snow Grains Thunderstorms, Heavy Rain Showers Freezing Rain, Snow Grains Name: count, dtype: int64 In [104... # second way is Filtering df[df["Weather Condition"] == "Snow"] Out [104... Date/Time Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa Weather Condition 1/3/2012 7:00 -14.0 -19.5 63 25.0 100.95 Snow 1/4/2012 12:00 -13.7 -21.7 51 101.25 Snow 1/4/2012 14:00 53 Snow -11.3 -19.0 19.3 100.97 1/4/2012 15:00 -10.2 -16.3 61 11 100.89 Snow 1/4/2012 16:00 -15.5 61 13 19.3 -9.4 100.79 Snow **8779** 12/31/2012 19:00 0.1 -2.7 81 30 100.13 9.7 Snow 0.2 -2.4 83 100.03 **8780** 12/31/2012 20:00 Snow **8781** 12/31/2012 21:00 -0.5 -1.5 93 28 4.8 99.95 Snow **8782** 12/31/2012 22:00 -0.2 -1.8 99.91 Snow **8783** 12/31/2012 23:00 -2.1 86 30 0.0 11.3 99.89 Snow 390 rows × 8 columns In [108... # Third way df[df["Weather Condition"].str.contains("Snow")] Date/Time Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa Weather Condition -9.5 57 **41** 1/2/2012 17:00 -2.1 22 25.0 99.66 **Snow Showers** -5.6 **44** 1/2/2012 20:00 -13.4 24 25.0 100.07 Snow Showers **45** 1/2/2012 21:00 -5.8 -12.8 58 26 25.0 100.15 **Snow Showers** 1/2/2012 23:00 -7.4 -14.1 59 17 19.3 100.27 Snow Showers 57 28 1/3/2012 0:00 -9.0 -16.0 25.0 100.35 Snow Showers ... **8779** 12/31/2012 19:00 0.1 -2.7 81 30 9.7 100.13 Snow **8780** 12/31/2012 20:00 0.2 -2.4 83 24 100.03 9.7 Snow 93 **8781** 12/31/2012 21:00 -0.5 -1.5 28 4.8 99.95 Snow -1.8 **8782** 12/31/2012 22:00 -0.2 28 99.91 9.7 Snow 86 **8783** 12/31/2012 23:00 0.0 -2.1 30 11.3 99.89 Snow 583 rows × 8 columns In [112... # 17) Find all the instances where Wind speed is above 24 and visisbility is 25 df[(df["Wind Speed\_km/h"] > 24) & (df["Visibility\_km"] == 25)] Date/Time Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa Weather Condition **23** 1/1/2012 23:00 5.3 2.0 79 30 25.0 99.31 Cloudy 1/2/2012 0:00 Rain Showers 5.2 1.5 77 35 25.0 99.26 1/2/2012 1:00 4.6 0.0 72 39 25 25.0 99.26 Cloudy 1/2/2012 2:00 3.9 -0.9 71 32 25.0 99.26 Mostly Cloudy 27 1/2/2012 3:00 3.7 -1.5 69 33 25.0 99.30 Mostly Cloudy -12.0 76 **8705** 12/28/2012 17:00 25.0 101.34 Mainly Clear **8753** 12/30/2012 17:00 -15.8 25.0 101.26 Mainly Clear 77 **8755** 12/30/2012 19:00 -13.4 -16.5 26 101.47 Mainly Clear 25.0 **8759** 12/30/2012 23:00 -15.1 25.0 101.52 Mostly Cloudy **8760** 12/31/2012 0:00 -11.1 -14.4 77 26 25.0 101.51 Cloudy 308 rows × 8 columns In [144... # 17) What is the mean value of each column against each weather condition? df.groupby("Weather Condition").mean(numeric\_only=True).head(10) Out [144... Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa **Weather Condition** 6.825716 0.089367 64.497738 10.557315 30.153243 101.587443 26.625752 100.911441 **Cloudy** 7.970544 2.375810 69.592593 16.127315 17.931707 100.435366 **Drizzle** 7.353659 88.243902 5.504878 16.097561 Drizzle,Fog 8.067500 7.033750 93.275000 11.862500 5.257500 100.786625 Drizzle,Ice Pellets,Fog 0.400000 92.000000 -0.700000 20.000000 4.000000 100.790000 Drizzle,Snow 1.050000 0.150000 93.500000 14.000000 10.500000 100.890000 Drizzle, Snow, Fog 0.693333 95.866667 15.533333 0.120000 5.513333 99.281333 Fog 4.303333 92.286667 7.946667 6.248000 101.184067 3.159333 Freezing Drizzle -5.657143 -8.000000 83.571429 16.571429 9.200000 100.202857 Freezing Drizzle,Fog -2.533333 -4.183333 88.500000 17.000000 5.266667 100.441667 In [148... # 18) what is the minimum and maximum value of each column against each weather condition df.groupby("Weather Condition").max(numeric\_only=True).head(5) Out[148... Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa **Weather Condition** 32.8 20.4 33 48.3 103.63 Clear 22.6 103.65 Cloudy 18.8 17.7 96 30 Drizzle 25.0 101.56 Drizzle,Fog 19.1 28 9.7 102.07 20 0.4 -0.7 92 Drizzle,Ice Pellets,Fog 100.79 4.0 # 19) what is the minimum and maximum value of each column against each weather condition df.groupby("Weather Condition").min(numeric\_only=True).head(5) Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa **Weather Condition** -23.3 -28.5 99.52 11.3 -21.4 -26.8 11.3 98.39 Cloudy 1.1 -0.2 6.4 97.84 Drizzle 0.0 98.65 Drizzle,Fog -1.6 1.0 20 Drizzle,Ice Pellets,Fog -0.7 92 4.0 100.79 In [152... # 20) Show all records where weather condition is Fog df[df["Weather Condition"] == "Fog"] Date/Time Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa Weather Condition 1/1/2012 0:00 -1.8 -3.9 101.24 Fog 1 1/1/2012 1:00 -1.8 101.24 -3.7 Fog 1/1/2012 4:00 -1.5 -3.3 88 101.23 Fog 1/1/2012 5:00 -1.4 -3.3 87 101.27 Fog 1/1/2012 6:00 -1.5 -3.1 89 101.29 Fog 12/29/2012 4:00 -16.0 -17.2 90 101.25 Fog 91 **8717** 12/29/2012 5:00 -14.8 -15.9 101.25 Fog 12/29/2012 6:00 -13.8 -15.3 101.25 Fog **8719** 12/29/2012 7:00 -14.8 101.22 -16.4 Fog 90 **8722** 12/29/2012 10:00 -12.0 -13.3 101.15 Fog 150 rows × 8 columns In [160... # 21) Find all instances where weather is clear or visibility is above 40 df[(df["Weather Condition"] == "Clear") | (df["Visibility\_km"] > 40)].head(5) Date/Time Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa Weather Condition **67** 1/3/2012 19:00 -16.9 Clear -24.8 24 25.0 101.74 **106** 1/5/2012 10:00 -10.0 73 17 48.3 100.45 Mainly Clear **107** 1/5/2012 11:00 -10.2 70 22 48.3 100.41 Mainly Clear **108** 1/5/2012 12:00 -9.6 20 48.3 100.38 Mainly Clear -4.4 -9.7 26 **109** 1/5/2012 13:00 48.3 100.40 Mainly Clear In [166... # 22) Find all instances where Weather is clear and Relative humidity is greater than 50 # Visibility is above 40 df[(df["Weather Condition"] == "Clear") & (df["Rel Hum\_%"] > 50) | (df["Visibility\_km"] > 40)] .head(5) Date/Time Temp\_C Dew Point Temp\_C Rel Hum\_% Wind Speed\_km/h Visibility\_km Press\_kPa Weather Condition 106 1/5/2012 10:00 73 17 -10.0 48.3 100.45 Mainly Clear 107 1/5/2012 11:00 -10.2 22 48.3 100.41 Mainly Clear **108** 1/5/2012 12:00 -4.7 -9.6 48.3 100.38 Mainly Clear **109** 1/5/2012 13:00 26 48.3 100.40 Mainly Clear **110** 1/5/2012 14:00 22 48.3 100.46 -10.7 65 Mainly Clear

