Activity_Explore descriptive statistics

January 7, 2024

1 Activity: Explore descriptive statistics

1.1 Introduction

Data professionals often use descriptive statistics to understand the data they are working with and provide collaborators with a summary of the relative location of values in the data, as well an information about its spread.

For this activity, you are a member of an analytics team for the United States Environmental Protection Agency (EPA). You are assigned to analyze data on air quality with respect to carbon monoxide, a major air pollutant. The data includes information from more than 200 sites, identified by state, county, city, and local site names. You will use Python functions to gather statistics about air quality, then share insights with stakeholders.

1.2 Step 1: Imports

Import the relevant Python libraries `pandas` and `numpy`.

[1]:
```
# Import relevant Python libraries.

import numpy as np
import pandas as pd
```

The dataset provided is in the form of a .csv file named `c4_epa_air_quality.csv`. It contains a susbet of data from the U.S. EPA. As shown in this cell, the dataset has been automatically loaded in for you. You do not need to download the .csv file, or provide more code, in order to access the dataset and proceed with this lab. Please continue with this activity by completing the following instructions.

[2]:
```
# RUN THIS CELL TO IMPORT YOUR DATA.

### YOUR CODE HERE
epa_data = pd.read_csv("c4_epa_air_quality.csv", index_col = 0)
```

Hint 1

Refer to the video about loading data in Python.

Hint 2

There is a function in the `pandas` library that allows you to read in data from a .csv file and load it into a DataFrame.

Hint 3

Use the `read_csv` function from the pandas `library`. The `index_col` parameter can be set to `0` to read in the first column as an index (and to avoid `"Unnamed: 0"` appearing as a column in the resulting DataFrame).

1.3 Step 2: Data exploration

To understand how the dataset is structured, display the first 10 rows of the data.

[3]:
```
# Display first 10 rows of the data.

epa_data.head(10)
```

[3]:

	date_local	state_name	county_name	city_name	local_site_name	parameter_name	units_of_measure	arithmetic_mean	aqi
0	2018-01-01	Arizona	Maricopa	Buckeye	BUCKEYE	Carbon monoxide	Parts per million	0.473684	7
1	2018-01-01	Ohio	Belmont	Shadyside	Shadyside	Carbon monoxide	Parts per million	0.263158	5
2	2018-01-01	Wyoming	Teton	Not in a city	Yellowstone National Park - Old Faithful Snow …	Carbon monoxide	Parts per million	0.111111	2
3	2018-01-01	Pennsylvania	Philadelphia	Philadelphia	North East Waste (NEW)	Carbon monoxide	Parts per million	0.300000	3
4	2018-01-01	Iowa	Polk	Des Moines	CARPENTER	Carbon monoxide	Parts per million	0.215789	3
5	2018-01-01	Hawaii	Honolulu	Not in a city	Kapolei	Carbon monoxide	Parts per million	0.994737	14
6	2018-01-01	Hawaii	Honolulu	Not in a city	Kapolei	Carbon monoxide	Parts per million	0.842105	12
7	2018-01-01	Pennsylvania	Erie	Erie	NaN	Carbon monoxide	Parts per million	0.242105	3
8	2018-01-01	Hawaii	Honolulu	Honolulu	Honolulu	Carbon monoxide	Parts per million	0.652632	9
9	2018-01-01	Colorado	Larimer	Fort Collins	Fort Collins - CSU - S. Mason	Carbon monoxide	Parts per million	0.178947	2

```
6  Parts per million         0.200000    2
7  Parts per million         0.200000    2
8  Parts per million         0.400000    5
9  Parts per million         0.300000    6
```

Hint 1

Refer to the video about exploratory data analysis in Python.

Hint 2

There is a function in the `pandas` library that allows you to get a specific number of rows from the top of a DataFrame.

Hint 3

Use the `head()` function from the `pandas` library.

Question: What does the `aqi` column represent?

Air Quality Index

Now, get a table that contains some descriptive statistics about the data.

[5]:
```
# Get descriptive stats.

epa_data.describe()
```

[5]:

	arithmetic_mean	aqi
count	260.000000	260.000000
mean	0.403169	6.757692
std	0.317902	7.061707
min	0.000000	0.000000
25%	0.200000	2.000000
50%	0.276315	5.000000
75%	0.516009	9.000000
max	1.921053	50.000000

Hint 1

Refer to the video about descriptive statistics in Python.

Hint 2

There is a function in the `pandas` library that allows you to generate a table of basic descriptive statistics about the numeric columns in a DataFrame.

Hint 3

Use the `describe()` function from the `pandas` library.

Question: Based on the table of descriptive statistics, what do you notice about the count value for the `aqi` column?

There are 260 measures represented in the dataset.

Question: What do you notice about the 25th percentile for the `aqi` column?

This is an important measure for understanding where the aqi values lie.

The 25th percemtile shows that 25% of the values are below 2.

Question: What do you notice about the 75th percentile for the `aqi` column?

This is another important measure for understanding where the aqi values lie.

The 75th percentile shows that 75% of the data is below 9.

1.4 Step 3: Statistical tests

Next, get some descriptive statistics about the states in the data.

[6]:
```
# Get descriptive stats about the states in the data.

epa_data["state_name"].describe()
```

[6]:
```
count            260
unique            52
top       California
freq              66
Name: state_name, dtype: object
```

Hint 1

Refer to the video about descriptive statistics in Python.

Hint 2

There is a function in the `pandas` library that allows you to generate basic descriptive statistics about a DataFrame or a column you are interested in.

Hint 3

Use the `describe()` function from the `pandas` library. Note that this function can be used: - "on a DataFrame (to find descriptive statistics about the numeric columns)" - "directly on a column containing categorical data (to find pertinent descriptive statistics)"

Question: What do you notice while reviewing the descriptive statistics about the states in the data?

Note: Sometimes you have to individually calculate statistics. To review to that approach, use the `numpy` library to calculate each of the main statistics in the preceding table for the `aqi` column.

There are also 260 state value names. California is the most repeated state. Appearing 66 times throughout the data.

1.5 Step 4. Results and evaluation

Now, compute the mean value from the `aqi` column.

[7]:
```
# Compute the mean value from the aqi column.
np.mean(epa_data["aqi"])
```

[7]:
```
6.757692307692308
```

Hint 1

Refer to the video about descriptive statistics in Python.

Hint 2

There is a function in the `numpy` library that allows you to get the mean value from an array or a Series of values.

Hint 3

Use the `mean()` function from the `numpy` library.

Question: What do you notice about the mean value from the `aqi` column?

This is an important measure, as it tells you what the average air quality is based on the data.

Rounding the decimal it shows the air quality is 6.76.

Next, compute the median value from the aqi column.

[8]:
```
# Compute the median value from the aqi column.

np.median(epa_data["aqi"])
```

[8]:
```
5.0
```

Hint 1

Refer to the video about descriptive statistics in Python.

Hint 2

There is a function in the `numpy` library that allows you to get the median value from an array or a series of values.

Hint 3

Use the `median()` function from the `numpy` library.

Question: What do you notice about the median value from the `aqi` column?

This is an important measure for understanding the central location of the data.

The median is 5, which means half of the values are below 5.

Next, identify the minimum value from the `aqi` column.

[9]:
```
# Identify the minimum value from the aqi column.
np.min(epa_data["aqi"])
```

[9]:
```
0
```

Hint 1

Refer to the video about descriptive statistics in Python.

Hint 2

There is a function in the `numpy` library that allows you to get the minimum value from an array or a Series of values.

Hint 3

Use the `min()` function from the `numpy` library.

Question: What do you notice about the minimum value from the `aqi` column?

This is an important measure, as it tell you the best air quality observed in the data.

The minumum aqi value is 0.

Now, identify the maximum value from the `aqi` column.

[10]:
```
# Identify the maximum value from the aqi column.

np.max(epa_data["aqi"])
```

[10]:
```
50
```

Hint 1

Refer to the video about descriptive statistics in Python.

Hint 2

There is a function in the `numpy` library that allows you to get the maximum value from an array or a Series of values.

Hint 3

Use the `max()` function from the `numpy` library.

Question: What do you notice about the maximum value from the `aqi` column?

This is an important measure, as it tells you which value in the data corresponds to the worst air quality observed in the data.

This means the highest aqi value is 50.

Now, compute the standard deviation for the `aqi` column.

By default, the `numpy` library uses 0 as the Delta Degrees of Freedom, while `pandas` library uses 1. To get the same value for standard deviation using either library, specify the `ddof` parameter to 1 when calculating standard deviation.

[11]:
```
# Compute the standard deviation for the aqi column.

np.std(epa_data["aqi"], ddof=1)
```

[11]: 7.0617066788207215

Hint 1

Refer to the video section about descriptive statistics in Python.

Hint 2

There is a function in the `numpy` library that allows you to get the standard deviation from an array or a series of values.

Hint 3

Use the `std()` function from the `numpy` library. Make sure to specify the `ddof` parameter as 1. To read more about this function, refer to its documentation in the references section of this lab.

Question: What do you notice about the standard deviation for the `aqi` column?

This is an important measure of how spread out the aqi values are.

The nearest decimal being 7.05 this is the spread of our data.

1.6 Considerations

What are some key takeaways that you learned during this lab?

Using python for statistics makes it a lot easier to get important information about our data. Coding the stats commands are also a lot more simple than I personally was expecting.

How would you present your findings from this lab to others? Consider the following relevant points noted by AirNow.gov as you respond: - "AQI values at or below 100 are generally thought of as satisfactory. When AQI values are above 100, air quality is considered to be unhealthy—at first for certain sensitive groups of people, then for everyone as AQI values increase." - "An AQI of 100 for carbon monoxide corresponds to a level of 9.4 parts per million."

Our findings show that all unique locations meet very healthy aqi, none surpassing 50, and 75% are under 9.

What summary would you provide to stakeholders? Use the same information provided previously from AirNow.gov as you respond.

After viewing the statistics of the data provided we have found that all locations in question meet healthy air requirments and none fall under unhealthy. Which 75% are healthy but the following 25% should have funding to find ways to make their AQI healthier.

References

Air Quality Index - A Guide to Air Quality and Your Health. (2014,February)

Numpy.Std — NumPy v1.23 Manual

US EPA, OAR. (2014, 8 July).*Air Data: Air Quality Data Collected at Outdoor Monitors Across the US*.

Congratulations! You've completed this lab. However, you may not notice a green check mark next to this item on Coursera's platform. Please continue your progress regardless of the check mark. Just click on the "save" icon at the top of this notebook to ensure your work has been logged.