

# Activity\_Explore confidence intervals

January 7, 2024

## 1 Activity: Explore confidence intervals

### 1.1 Introduction

The Air Quality Index (AQI) is the Environmental Protection Agency's index for reporting air quality. A value close to 0 signals little to no public health concern, while higher values are associated with increased risk to public health. The United States is considering a new federal policy that would create a subsidy for renewable energy in states observing an average AQI of 10 or above.

You've just started your new role as a data analyst in the Strategy division of Ripple Renewable Energy (RRE). **RRE operates in the following U.S. states: California, Florida, Michigan, Ohio, Pennsylvania, Texas.** You've been tasked with constructing an analysis which identifies which of these states are most likely to be affected, should the new federal policy be enacted.

Your manager has requested that you do the following for your analysis: 1. Provide a summary of the mean AQI for the states in which RRE operates. 2. Construct a boxplot visualization for AQI of these states using `seaborn`. 3. Evaluate which state(s) may be most affected by this policy, based on the data and your boxplot visualization. 4. Construct a confidence interval for the RRE state with the highest mean AQI.

### 1.2 Step 1: Imports

#### 1.2.1 Import packages

Import `pandas` and `numpy`.

```
[1]: # Import relevant packages

import pandas as pd
import numpy as np
```

#### 1.2.2 Load the dataset

The dataset provided gives national Air Quality Index (AQI) measurements by state over time. `Pandas` is used to import the file `c4_epa_air_quality.csv` as a `DataFrame` named `aqi`. 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.

*Note: For the purposes of your analysis, you can assume this data is randomly sampled from a larger population.*

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

### YOUR CODE HERE ###
aqi = pd.read_csv('c4_epa_air_quality.csv')
```

## 1.3 Step 2: Data exploration

### 1.3.1 Explore your dataset

Before proceeding to your deliverables, spend some time exploring the `aqi` DataFrame.

```
[27]: # Explore your DataFrame `aqi`.
print("Use describe() to summarize AQI")
print(aqi.describe(include='all'))

print("For a more thorough examination of observations by state use ↵
↵values_counts()")
print(aqi['state_name'].value_counts())
```

Use describe() to summarize AQI

	Unnamed: 0	date_local	state_name	county_name	city_name \
count	260.000000	260	260	260	260
unique	NaN	1	52	149	190
top	NaN	2018-01-01	California	Los Angeles	Not in a city
freq	NaN	260	66	14	21
mean	129.500000	NaN	NaN	NaN	NaN
std	75.199734	NaN	NaN	NaN	NaN
min	0.000000	NaN	NaN	NaN	NaN
25%	64.750000	NaN	NaN	NaN	NaN
50%	129.500000	NaN	NaN	NaN	NaN
75%	194.250000	NaN	NaN	NaN	NaN
max	259.000000	NaN	NaN	NaN	NaN

	local_site_name	parameter_name	units_of_measure	arithmetic_mean \
count	257	260	260	260.000000
unique	253	1	1	NaN
top	Kapolei	Carbon monoxide	Parts per million	NaN
freq	2	260	260	NaN
mean	NaN	NaN	NaN	0.403169
std	NaN	NaN	NaN	0.317902
min	NaN	NaN	NaN	0.000000

25%	NaN	NaN	NaN	0.200000
50%	NaN	NaN	NaN	0.276315
75%	NaN	NaN	NaN	0.516009
max	NaN	NaN	NaN	1.921053

	aqi
count	260.000000
unique	NaN
top	NaN
freq	NaN
mean	6.757692
std	7.061707
min	0.000000
25%	2.000000
50%	5.000000
75%	9.000000
max	50.000000

For a more thorough examination of observations by state use values\_counts()

California	66
Arizona	14
Ohio	12
Florida	12
Texas	10
New York	10
Pennsylvania	10
Michigan	9
Colorado	9
Minnesota	7
New Jersey	6
Indiana	5
North Carolina	4
Massachusetts	4
Maryland	4
Oklahoma	4
Virginia	4
Nevada	4
Connecticut	4
Kentucky	3
Missouri	3
Wyoming	3
Iowa	3
Hawaii	3
Utah	3
Vermont	3
Illinois	3
New Hampshire	2
District Of Columbia	2
New Mexico	2

Montana	2
Oregon	2
Alaska	2
Georgia	2
Washington	2
Idaho	2
Nebraska	2
Rhode Island	2
Tennessee	2
Maine	2
South Carolina	1
Puerto Rico	1
Arkansas	1
Kansas	1
Mississippi	1
Alabama	1
Louisiana	1
Delaware	1
South Dakota	1
West Virginia	1
North Dakota	1
Wisconsin	1

Name: state\_name, dtype: int64

**Question:** What time range does this data cover?

It covers the range of the year 2018.

**Question:** What are the minimum and maximum AQI values observed in the dataset?

The minimum AQI value is 0 and the maximum AQI value is 50.

**Question:** Are all states equally represented in the dataset?

No all states are not equally represented

Hint 1

Refer to [the content about descriptive statistics](#).

Hint 2

Use `pandas` or `numpy` to explore the `aqi` DataFrame.

Hint 3

Use any of the following functions: - `pandas`: `describe()`,`value_counts()`,`shape()` - `numpy`: `unique()`,`mean()`

## 1.4 Step 3: Statistical tests

### 1.4.1 Summarize the mean AQI for RRE states

Start with your first deliverable. Summarize the mean AQI for the states in which RRE operates.

```
[28]: # Summarize the mean AQI for RRE states.

# Create a list of RRE states.
rre_states = ['California', 'Florida', 'Michigan', 'Ohio', 'Pennsylvania', 'Texas']

# Subset `aqi` to only consider these states.
aqi_rre = aqi[aqi['state_name'].isin(rre_states)]

# Find the mean aqi for each of the RRE states.
aqi_rre.groupby(['state_name']).agg({"aqi": "mean", "state_name": "count"})
```

```
[28]:
```

	aqi	state_name
state_name		
California	12.121212	66
Florida	5.500000	12
Michigan	8.111111	9
Ohio	3.333333	12
Pennsylvania	2.900000	10
Texas	2.700000	10

Hint 1

Subset your DataFrame to only include those states in which RRE operates.

Hint 2

Define a list consisting of the states in which RRE operates and use that list to subset your DataFrame.

Hint 3

Use `pandas isin()` to subset your DataFrame by the list of RRE states.

### 1.4.2 Construct a boxplot visualization for the AQI of these states

Seaborn is a simple visualization library, commonly imported as `sns`. Import `seaborn`. Then utilize a boxplot visualization from this library to compare the distributions of AQI scores by state.

```
[29]: # Import seaborn as sns.

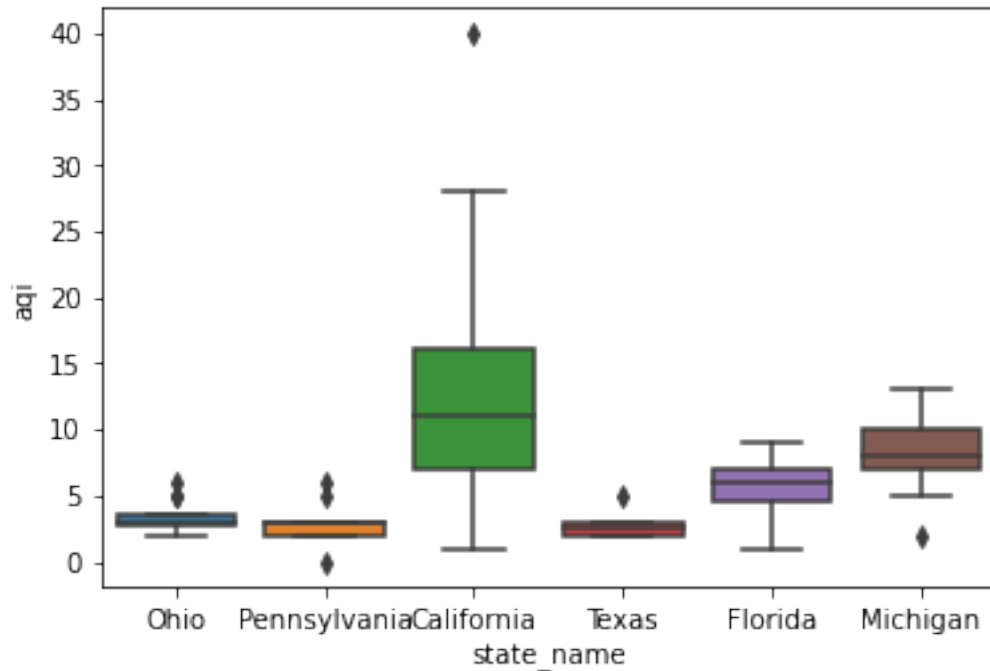
import seaborn as sns
```

### 1.4.3 Create an in-line visualization showing the distribution of aqi by state\_name

Now, create an in-line visualization showing the distribution of aqi by state\_name.

```
[30]: sns.boxplot(x=aqi_rre["state_name"],y=aqi_rre["aqi"])
```

```
[30]: <matplotlib.axes._subplots.AxesSubplot at 0x7ff9d7df8510>
```



Hint 1

Use the boxplot visual for this purpose.

Hint 2

Reference [Seaborn's boxplot visualization documentation](#).

Hint 3

Assign `state_name` to the x argument and `aqi` to the y.

**Question:** Based on the data and your visualizations, which state(s) do you suspect will be most affected by this policy?

California and Michigan

Hint 1

Consider the mean AQI for the RRE states, as well as the distribution in the boxplots relative to the policy limit (10).

#### 1.4.4 Construct a confidence interval for the RRE state with the highest mean AQI

Recall the 4-step process in constructing a confidence interval:

1. Identify a sample statistic.
2. Choose a confidence level.
3. Find the margin of error.
4. Calculate the interval.

#### 1.4.5 Construct your sample statistic

To construct your sample statistic, find the mean AQI for your state.

```
[31]: # Find the mean aqi for your state.  
aqi_ca = aqi[aqi['state_name']=='California']  
  
sample_mean = aqi_ca['aqi'].mean()  
sample_mean
```

```
[31]: 12.121212121212121
```

Hint 1

Reference what you've previously learned to recall what a [sample statistic](#) is.

Hint 2

Calculate the mean for your highest AQI state to arrive at your sample statistic.

Hint 3

Call the `mean()` function within `pandas` on your `DataFrame`.

#### 1.4.6 Choose your confidence level

Choose your confidence level for your analysis. The most typical confidence level chosen is 95%; however, you can choose 90% or 99% if you want decrease or increase (respectively) your level of confidence about your result.

```
[32]: # Input your confidence level here:  
  
confidence_level = 0.95  
confidence_level
```

```
[32]: 0.95
```

### 1.4.7 Find your margin of error (ME)

Recall **margin of error** =  $z * \text{standard error}$ , where  $z$  is the appropriate  $z$ -value for the given confidence level. To calculate your margin of error:

- Find your  $z$ -value.
- Find the approximate  $z$  for common confidence levels.
- Calculate your **standard error** estimate.

Confidence Level	Z Score
90%	1.65
95%	1.96
99%	2.58

```
[33]: # Calculate your margin of error.

#### YOUR CODE HERE ####

# Begin by identifying the z associated with your chosen confidence level.
z_value = 1.96

# Next, calculate your standard error.
standard_error = aqi_ca['aqi'].std() / np.sqrt(aqi_ca.shape[0])
print("standard error:")
print(standard_error)

# Lastly, use the preceding result to calculate your margin of error.

margin_of_error = standard_error * z_value
print("margin of error:")
print(margin_of_error)
```

```
standard error:
0.8987209641127412
margin of error:
1.7614930896609726
```

### 1.4.8 Calculate your interval

Calculate both a lower and upper limit surrounding your sample mean to create your interval.

```
[34]: # Calculate your confidence interval (upper and lower limits).

upper_ci_limit = sample_mean + margin_of_error
lower_ci_limit = sample_mean - margin_of_error
(lower_ci_limit, upper_ci_limit)
```



[34]: (10.359719031551148, 13.882705210873095)

Hint 1

Refer to [the content about constructing a confidence interval](#).

Hint 2

Identify the sample mean from your prior work. Then use the margin of error to construct your upper and lower limits.

Hint 3

Subtract the margin of error from the sample mean to construct your lower limit, and add the margin of error to your sample mean to construct your upper limit.

#### 1.4.9 Alternative: Construct the interval using `scipy.stats.norm.interval()`

`scipy` presents a simpler solution to developing a confidence interval. To use this, first import the `stats` module from `scipy`.

```
[35]: # Import stats from scipy.  
  
from scipy import stats
```

### 1.5 Step 4: Results and evaluation

#### 1.5.1 Recalculate your confidence interval

Provide your chosen `confidence_level`, `sample_mean`, and `standard_error` to `stats.norm.interval()` and recalculate your confidence interval.

```
[36]: stats.norm.interval(alpha=confidence_level, loc=sample_mean,   
    ↪ scale=standard_error)
```

[36]: (10.359751399400034, 13.882672843024208)

## 2 Considerations

**What are some key takeaways that you learned from this lab?**

California and Michigan have the highest AQI levels, so they would be most affected by the policy change. **What findings would you share with others?**

Have the audience understand that this data meets the required 95% CI, the sample data is showing a yeild of [10.36 , 13.88] **What would you convey to external stakeholders?**

California is considerably higher than other states including Michigan which was second and also above 10 AQI. Both states will be impacted by this policy change.

## References

[seaborn.boxplot](#) — [seaborn 0.12.1 documentation](#). (n.d.).

**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.