Covariance and Correlation: Questions

Dr. Roi Yozevitch

Basic Questions

Question 1: Define covariance and correlation. Explain the difference between them.

Question 2: Given the following data points for two variables X and Y:

$$X = [1, 2, 3, 4, 5], \quad Y = [2, 4, 6, 8, 10]$$

Compute the covariance and the Pearson correlation coefficient.

Intermediate Questions

Question 3: Load the `tips` dataset from the `seaborn` library in Python. Compute the covariance matrix for the numerical variables in the dataset.

Question 4: Using the `tips` dataset, compute the correlation matrix. Identify and interpret the highest correlation in the matrix.

Question 5: Write a Python function to compute the Pearson correlation coefficient between two arrays without using any built-in functions. Test your function on the arrays $X = [1, 2, 3, 4, 5]$ and $Y = [5, 4, 3, 2, 1]$.

```
import seaborn as sns

# Load the tips dataset
tips = sns.load_dataset('tips')
```

Advanced Questions

Question 6: Download the `iris` dataset from the `sklearn` library in Python. Compute the covariance and correlation matrices. Discuss any patterns you observe.

```
from sklearn.datasets import load_iris
import pandas as pd

# Load the iris dataset
data = load_iris()
iris = pd.DataFrame(data.data, columns=data.feature_names)
```

Question 7: Write a Python program that reads a CSV file containing two columns of numerical data. The program should compute and print the covariance and correlation between the two columns.

Question 8: Explain Simpson's Paradox. Provide an example where computing the correlation on a combined dataset gives a different result than computing the correlations on separated groups.

Question 9: Using the `tips` dataset, demonstrate Simpson's Paradox by computing the correlation between `total_bill` and `tip` for smokers and non-smokers separately, and then for the combined dataset. Interpret your results.

Question 10: Create a synthetic dataset where Simpson's Paradox is evident. Write a Python script to demonstrate the paradox by computing the correlations for subgroups and the combined group.

```
import numpy as np

# Example of creating a synthetic dataset
data = {
    'group': np.repeat(['A', 'B'], 50),
    'x': np.concatenate([np.random.normal(10, 2, 50), np.random.normal(20, 5, 50
    'y': np.concatenate([np.random.normal(15, 2, 50), np.random.normal(25, 5, 50
}
df = pd.DataFrame(data)
```