Assignment 1

Score: 55 / 56

What is the purpose of using Python for data analysis?

The variety of libraries available in Python for data analysis is what makes it the best. For example, pandas, NumPy, matplotlib and seaborn provide functional tools for data cleaning, transformations, manipulations and plotting.

Feedback

Great answer! You have clearly explained how Python libraries like pandas, NumPy, matplotlib, and seaborn are used for data analysis.

How can you create a DataFrame in pandas?

Dataframes can be created in Pandas in various ways such as from dictionaries, lists or NumPy arrays.

Feedback

Great answer! Keep it up!

Explain the difference between numpy arrays and lists in Python.

Numpy array vs list python:

Numpy arrays are homogeneous and have fixed sizes, while lists have varying elements that can be any type of data.

While numpy arrays perform vectorized operations and mathematical functions, enabling them to be more efficient in numerical computations as compared to lists,

Also, numpy arrays use less memory space than lists, resulting in better performance on large datasets.

Feedback

Good explanation of the differences between numpy arrays and lists. Consider providing an example or elaborating on how numpy arrays are created and accessed for a more comprehensive answer.

What are some common data visualization techniques used in matplotlib?

Matplotlib has several commonly used visualisation techniques, including:

line plots

scatter plots

bar plots

histograms

box plots

heatmaps

pie charts

Feedback

Great Answer! Well done!

How can you handle missing data in a pandas DataFrame?

Missing values in a Pandas data frame could be resolved through the following methods:

remove rows or columns with missing values using dropna()

filling missing values with specified values using fillna()

filling missing values using interpolate()

Feedback

Great Answer! Well explained methods for handling missing data in a Pandas DataFrame.

What is the purpose of using the groupby function in pandas?

The groupby() function in Pandas is used to split the data into groups based on some criteria, apply a function to each group independently, and then combine the results into a DataFrame.

Feedback

Great answer! You have provided a clear and accurate explanation of the purpose of the groupby() function in Pandas.

Explain the concept of broadcasting in numpy.

In NumPy, the ability to operate arrays of diverse shapes together without explicitly reshaping or broadcasting the arrays is referred to as broadcasting. It automatically aligns the dimensions of arrays of varied shapes, enabling element-wise actions between them.

Feedback

Great answer! You've provided a clear and accurate explanation of broadcasting in NumPy.

How can you concatenate multiple DataFrames in pandas?

Using the concat() method in Pandas, several DataFrames can be concatenated by specifying the axis along which to concatenate (axis = 0 for rows and axis = 1 for columns is typical).

Feedback

Great answer! Concise and accurate explanation of concatenating DataFrames in Pandas using the concat() method.

What are the advantages of using seaborn over matplotlib for data visualization?

For data visualisation, there are a few benefits to utilising Seaborn rather than Matplotlib.

Higher-level features offered by Seaborn enable the creation of more visually appealing and educational visualisations with less code.

Regression plots, box plots, violin plots, and other statistical plots are all supported natively by Seaborn.

Seaborn handles missing data and categorical variables more elegantly and works nicely with Pandas DataFrames.

Feedback

Great answer! You provided a detailed explanation of the benefits of using Seaborn over Matplotlib for data visualization. Well done!

Write a code snippet to create a scatter plot using matplotlib.

```
import matplotlib.pyplot as plt

# Sample data
x = [1, 2, 3, 4, 5]
y = [2, 3, 5, 7, 11]

# Create scatter plot
plt.scatter(x, y)
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.title('Scatter Plot')
plt.grid(True)
plt.show()
```

Feedback

Great Answer! Well done!