

SCORE BOOSTER



SPPU
Hub

DATA ANALYTICS LAB

VIVA/ORAL SOLUTION

OPTIMISED



SOLUTIONS

Second Year (SE) Degree Course In
Electronics Engineering /
Electronics & Telecommunications
Engineering

2019
PATTERN

SPPU

ACCORDING TO NEW REVISED CREDIT SYSTEM



Savitribai Phule Pune University

**Department Of Electronics & Telecommunications
Engineering**

DATA ANALYTICS LAB
VIVA / ORAL SOLUTION

Topic Wise Question Bank

Understanding The Data

- What is data?
- What is Use of data ?
- Which are types of Data?
- What is difference between structured & unstructured data?

Understanding The Data Analytics :

- Define Data Analytics?
- What is importance of Data Analytics?
- Which are different types of Data Analytics?

Python In Data Analytics :

- What is jupyter notebook ? is python interpreter language ?
- What is python ?
- Features of python?
- Define list ?
- What are the different method of list ?
- explain the list indexing and slicing with an example?
- Define tuple?
- Define dictionary ?
- what are the different Properties of Dictionary?
- What is set?
- State the Properties of Sets.
- Which are different packages of python used in data science?

Numpy Module in Python

- What is use of NumPy? Explain with example?
- How to create 1D & 2D array using NumPy? Explain with example?
- What are the different method of creating a numpy array
- What is neseccary condition while creating a one dimension array ?
- What is neseccary condition while creating a two dimension array ?
- Write the syntax to rename a module.
- What is purpose of shape function in numpy array.
- What is reshape function ?

Matplotlib Module In Python

- What is matplotlib ?
- What is the purpose of object oriented interface of matplotlib ?
- What is line chart?
- What is bar chart?
- What is histogram?
- What is scatter plot ?
- How to create Bar Plot using Matplotlib? Explain with example?
- How to create Scatter Plot using Matplotlib? Explain with example?
- How to create Histogram using Matplotlib? Explain with example?
- How to create Boxplot using Matplotlib? Explain with example?
- How to create Pair plot Matplotlib? Explain with example?
- How to create Heat map using Matplotlib? Explain with example?
- What are the different library for data visualization in Python

Pandas library In Python

- What is pandas?

- What is use of Pandas? Explain with example?
- what is meant by data frame
- How to create data frame using Pandas? Explain with example?
- How to import & export csv file using Pandas? Explain with example?
- How to import & export excel file using Pandas? Explain with example?
- How can you read a CSV file in Python using pandas?
- What are some file formats supported by pandas for importing data?
- how to get a index of the data frame write syntax for it
- write syntax to get a column level of the data frame
- write syntax for to get total number of element from the data frame
- What is data Wrangling in Python ?
- which operator is used for indexing and selecting data in pandas

Data Normalization In Python

- What is data normalization?
- Which methods are used in python for data normalization?
- what is binning in python? Why it is used?
- How to find standard deviation & variance of given data in python?
- What is Hypothesis?
- What is Null Hypothesis?
- What is use of scipy module in python?

Exploratory Data Analysis

- What is EDA?
- what is probability
- Probability is the measure of how likely an event is to occur, represented by a number between 0 and 1.

- what are the basic rules of the probability
- what is probability distributions
- what is bootstrapping

Correlation In Python

- what is correlation?
- What is correlation coefficient?.
- What is Pearson's correlation coefficient equation?
- What is meant by covariance
- What is significance of each correlation
- What is groupby() operation in python ?
- What is splitting applying and combining ?
- How do we group by a list in Python
- What is ANOVA ?
- Why we use ANOVA ?
- How the ANOVA is work ?

Regression Analysis In Python

- What is use of Regression analysis?
- Which are types of Regression?
- What is linear regression and multiple linear regression
- Which are different parameters used for evaluation of regression analysis?
- What is polynomial regression
- Explain difference between linear & logistic regression?
- what are the difference between correlation and regression ?

Understanding The Data :

What is data?

Data is information that can be collected, stored, and analyzed. It can be in different forms like numbers, text, images, or videos. Data is used for making decisions and gaining insights. It can be structured or unstructured and is crucial in various industries.

What is Use of data ?

Data is used to make informed decisions, detect fraud and enhance security, predict future outcomes. drive research and innovation, monitor performance.

Which are types of Data?

There are different types of data

1. Numerical data:

Data represented by numbers, either discrete (whole numbers) or continuous (values that can take any value within a range).

2. Categorical data:

Data that represents different categories or groups, such as gender, colors, or car models.

3. Text data:

Unstructured data in the form of text or documents, such as articles, emails, or social media posts.

4. Multimedia data: Data that combines text, images, audio, and video, including multimedia files like photos, videos, or audio recordings.

What is difference between structured & unstructured data?

Structured Data	Unstructured Data
In this type of data, the data is stored in processed form or containing labels in which searching of data is easy.	In this type, the data is stored in unprocessed form or raw form in which searching is complex.
This form of data is generally used to store quantitative data such as height, weight, age.	This form of data is used to store qualitative data such as invoices, records, and media-related data.
To store such types of data, data warehouses are used.	To store unstructured data, data lakes are used.
In this form, the data is stored in predefined format support by underlying database architecture.	In this form, the data can be stored in different formats.
Several analytical tools are available for mining structured data.	There are no tools present currently for mining unstructured data.

In simple language, Structured data is organized and follows a specific format, like a table in a spreadsheet. Unstructured data is not organized and can be in different forms, like text, images, or videos, without a specific format.

Understanding The Data Analytics :

Define Data Analytics?

Data analytics converts raw data into useful information . It includes a range of tools, technologies, and processes used solve problems by using data.

What is importance of Data Analytics?

Data analytics is important because it provides valuable insights for decision-making, improves efficiency, enables personalization, creates a competitive advantage, and helps in risk management.

One example of the importance of data analysis is in the field of marketing. Suppose a company wants to launch a new product in the market. By conducting data analysis, they can gain valuable insights about their target audience, market trends, and competition.

Which are different types of Data Analytics?

There are different types of data analytics:

1. Descriptive Analytics: in which the data is analytics in specific time
2. Diagnostic Analytics: Investigates data to determine the reasons behind past outcomes or issues.
3. Predictive Analytics: Uses historical data and statistical modeling to make predictions about future outcomes.
4. Prescriptive Analytics: Recommends specific actions or decisions based on data analysis to achieve desired outcomes.

Python In Data Analytics

What is jupyter notebook ? is python interpreter language ?

Jupyter Notebook is an interactive web application used for creating and sharing documents containing code, visualizations, and explanatory text. Python is a programming language that can be executed interactively using an interpreter, and Jupyter Notebook provides an environment to write and execute Python code within the notebook.

What is python ?

Python is open source programming language which is High-level language, Easy to understand This Pseudo code nature of Python makes it easy to learn and understandable for beginners.

Features of python:

Python's key features include its easy-to-learn and readable syntax, extensive standard library, cross-platform compatibility, and support for object-oriented programming.

Define list ?

Python Lists are containers to store a set of values of any data type.

For e.g.

```
friends = ['Apple', 'Akash', 'Rohan', 7, False]
```

What are the different method of list ?

List Methods

Consider the following list:

```
l1 = [1, 8, 7, 2, 21, 15]
```

sort() – updates the list to [1,2,7,8,15,21]

reverse() – updates the list to [15,21,2,7,8,1]

append(8) – adds 8 at the end of the list

insert(3,8) – This will add 8 at 3 index

pop(2) – It will delete the element at index 2 and return its value

remove(21) – It will remove 21 from the list

explain the list indexing and slicing with an example?

– List indexing is accessing individual elements in a list using their position or index.

– List slicing is extracting a portion of elements from a list by specifying a start index, an end index, and an optional step value.

For.e.g.

```
L1 = [7, 9, 'raj']
```

```
L1[0] – 7 (this is an indexing)
```

```
L1[0:2] – [7,9] (This is known as List Slicing)
```

Define tuple?

A tuple is an ordered collection of values that cannot be changed after creation. It is often used to group related data together.

```
a = () #It is an example of empty tuple
```

```
a = (1,) #Tuple with only one element needs a comma
```

```
a = (1, 7, 2) #Tuple with more than one element
```

Define dictionary ?

dictionary in Python is a collection of key-value pairs used to store, retrieve, and manipulate data based on unique keys..

```
""" a = {"key": "value",
```

```
"ajay": "code",
```

```
"marks" : "100",
```

```
"list": [1,2,9]} """
```

```
a["key"] # Prints value
```

```
a["list"] # Prints [1,2,9] """
```

what are the different Properties of Dictionary?

- Dictionaries are mutable, allowing for modifications and additions of key-value pairs.
- They are unordered, meaning the elements are not stored in a specific order.

- Keys in a dictionary are unique.
- Dictionaries provide flexibility in storing and accessing data based on key-value pairs.

What is set?

a set in Python is an unordered collection of unique elements. Sets automatically remove duplicates and support various operations. They are mutable and useful for storing distinct elements efficiently.

```
S= Set()      # No repetition allowed!
```

```
S.add(1)
```

```
S.add(2)
```

```
# or Set = {1,2}
```

State the Properties of Sets.

- Sets are unordered # Elements order doesn't matter
- Sets are unindexed # Cannot access elements by index
- There is no way to change items in sets
- Sets cannot contain duplicate values

Which are different packages of python used in data science?

1. NumPy: Library for numerical computing with support for arrays and mathematical functions.
2. Pandas: Library for data manipulation and analysis, providing data structures like DataFrames.
3. Matplotlib: Plotting library for creating visualizations in Python.
4. Seaborn: Statistical data visualization library built on Matplotlib.
5. SciPy: Library for scientific computing with modules for optimization, interpolation, and more.

7. TensorFlow and PyTorch: Deep learning frameworks for building and training neural networks.

Numpy Module in Python

What is use of NumPy? Explain with example?

NumPy is a Python module that allows for efficient handling and manipulation of large arrays and matrices, providing a wide range of mathematical and array operations

How to create 1D & 2D array using NumPy? Explain with example?

To create 1D and 2D arrays using NumPy, you can utilize the 'numpy.array()' function.

examples of creating a 1D and a 2D array:

1. Creating a 1D array:

Import numpy as np

Create a 1D array

```
Arr_1d = np.array([1, 2, 3, 4, 5])
```

Print the array

```
Print(arr_1d)
```

Output:

```
[1 2 3 4 5]
```

In this example, we use the np.array() function to create a 1D array arr_1d with elements 1, 2, 3, 4, and 5.

2. Creating a 2D array:

Import numpy as np

Create a 2D array

```
Arr_2d = np.array([[1, 2, 3], [4, 5, 6]])
```

Print the array

```
Print(arr_2d)
```

Output:

```
[[1 2 3]
```

```
[4 5 6]]
```

In this example, we use the 'np.array()' function to create a 2D array 'arr_2d' with two rows and three columns. The inner lists represent the rows, and the elements within each row represent the columns.

.

What are the different method of creating a numpy array

NumPy arrays can be created using methods such as conversion from lists or tuples, using built-in functions like 'zeros()', 'ones()', 'full()', 'arange()', and generating random values, as well as reshaping existing arrays.

1. Using 'numpy.array()':

The 'numpy.array()' function allows you to create an array from a Python list or tuple.

Import numpy as np

Create an array from a list

```
Arr1 = np.array([1, 2, 3, 4, 5])
```

```
Print(arr1)
```


Output:

```
[1 2 3 4 5]
```

2. Using 'numpy.arange()':

The 'numpy.arange()' function creates an array with regularly spaced values within a specified range.

```
Import numpy as np
```

```
# Create an array with a range of values
```

```
Arr2 = np.arange(1, 10, 2) # Start at 1, end before 10, step by 2
```

```
Print(arr2)
```

Output:

```
[1 3 5 7 9]
```

3. Using 'numpy.zeros()':

The 'numpy.zeros()' function creates an array filled with zeros.

```
Import numpy as np
```

```
# Create an array filled with zeros
```

```
Arr3 = np.zeros((3, 4)) # Create a 3x4 array of zeros
```

```
Print(arr3)
```

Output:

```
[[0. 0. 0. 0.]
```

```
[0. 0. 0. 0.]
```

```
[0. 0. 0. 0.]]
```

4. Using 'numpy.ones()':

The 'numpy.ones()' function creates an array filled with ones.

Import numpy as np

```
# Create an array filled with ones
```

```
Arr4 = np.ones((2, 3)) # Create a 2x3 array of ones
```

```
Print(arr4)
```

Output:

```
[[1. 1. 1.]
```

```
[1. 1. 1.]]
```

5. Using 'numpy.eye()':

The 'numpy.eye()' function creates a 2D identity matrix, where diagonal elements are ones and others are zeros.

Import numpy as np

```
# Create a 2D identity matrix
```

```
Arr5 = np.eye(3) # Create a 3x3 identity matrix
```

```
Print(arr5)
```

'''

Output:

```
[[1. 0. 0.]
```

```
[0. 1. 0.]
```

```
[0. 0. 1.]]
```

Which command is used to install numpy module ?

To install the NumPy module, we can use the following command
`pip install numpy`

This command uses the pip package manager, which is commonly used for installing Python packages.

What is necessary condition while creating a one dimension array ?

when creating a one-dimensional array in NumPy, we need to specify the number of elements in the array using a single integer value. For example, `'np.zeros(5)'` creates a one-dimensional array with 5 elements.

What is necessary condition while creating a two dimension array ?

when creating a two-dimensional array in NumPy, we need to specify the shape as a tuple containing two integers representing the number of rows and columns in the array.

Write the syntax to rename a module.

To rename a module in Python, we can use the `as` keyword followed by the desired alias name. The syntax is:

```
Import module_name as alias_name
```

What is purpose of shape function in numpy array.

the `'shape'` function in NumPy returns the dimensions or shape of a NumPy array as a tuple, providing information about its size and structure.

What is reshape function ?

Reshape function is used to convert a one-dimensional array into a two-dimensional array . also the `'reshape'` function in NumPy is used to change the shape of an array without modifying its data. It returns a new view of the array with the desired shape.

Matplotlib Module In Python

What is matplotlib ?

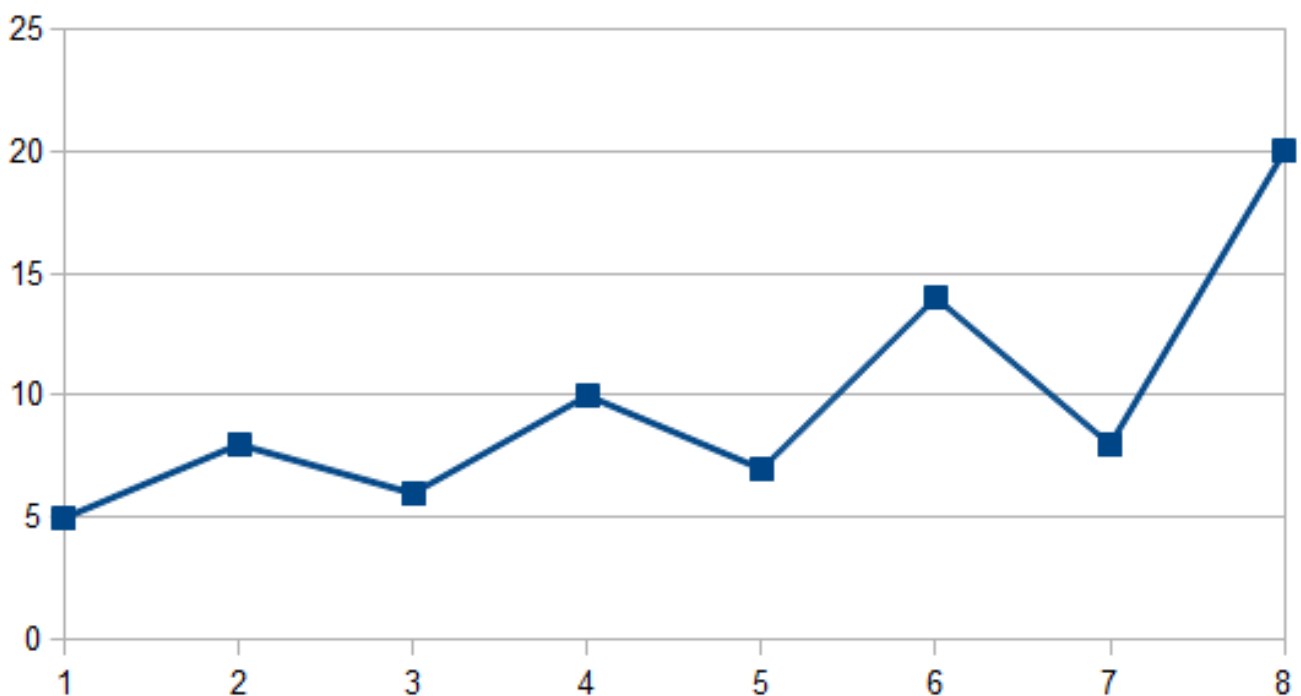
Matplotlib is a Python library for creating plots and visualizations. It provides a range of tools and functions for generating different types of 2D and 3D plots with customization options.

What is the purpose of object oriented interface of matplotlib ?

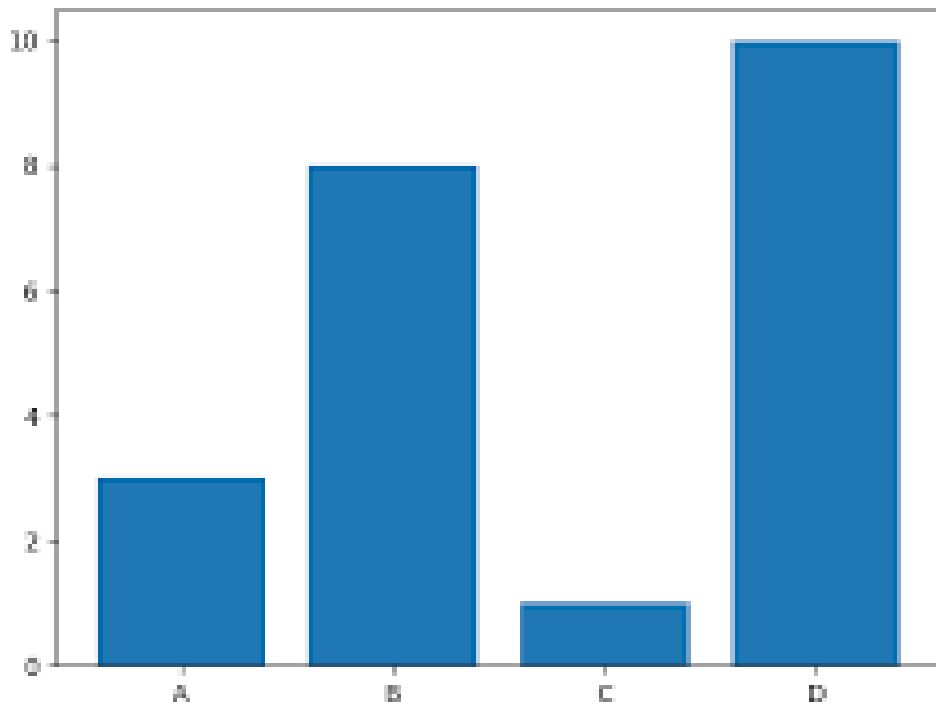
In short, the object-oriented interface of Matplotlib allows for greater control and customization of plots by directly manipulating Figure and Axes objects.

what is line chart?

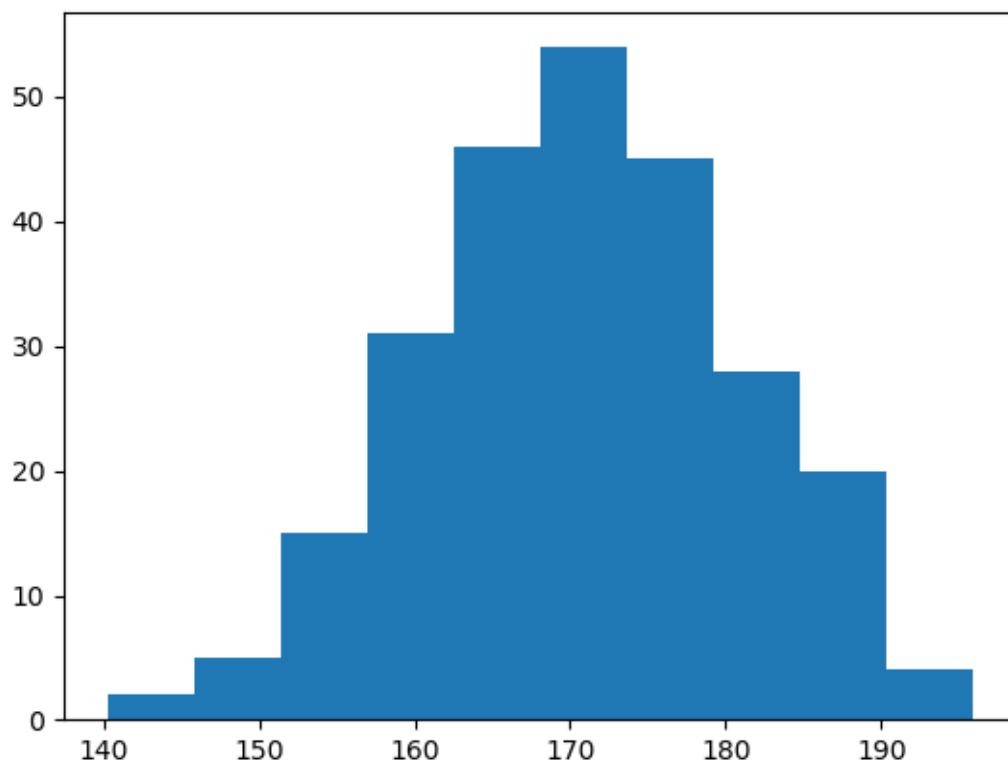
- **Line Chart:** provide relationships between two variables over a continuous interval, commonly used for time series data



- **Bar Chart:** Compare and display categorical data using rectangular bars.

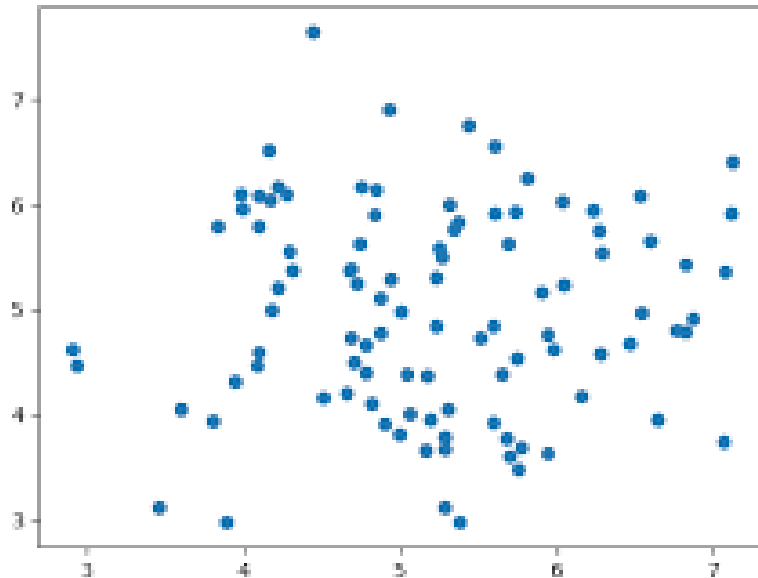


- **Histogram:** Show the distribution and frequency of continuous or discrete data using bins.



What is scatter plot ?

a scatter plot is a type of plot used to display the relationship between two numerical variables. It represents data points as individual markers on a two-dimensional coordinate system.



How to create Line chart using Matplotlib? Explain with example?

Import matplotlib.pyplot as plt

```
X = [1, 2, 3, 4, 5]
```

```
Y = [2, 4, 6, 8, 10]
```

```
Plt.plot(x, y)
```

```
Plt.xlabel('X-axis')
```

```
Plt.ylabel('Y-axis')
```

```
Plt.title('Line Chart')
```

```
Plt.show()
```

In this example, we import the 'pyplot' module from Matplotlib and define the data for the x-axis ('x') and y-axis ('y').

We then use the 'plt.plot()' function to create the line chart by passing the 'x' and 'y' data. The chart is displayed using 'plt.show()'.

How to create Bar Plot using Matplotlib? Explain with example?

To create a bar plot using Matplotlib, you can utilize the 'matplotlib.pyplot.bar()' function. Here's an example:

```
Import matplotlib.pyplot as plt
```

```
# Data for x-axis and y-axis
```

```
X = ['A', 'B', 'C', 'D', 'E']
```

```
Y = [10, 15, 7, 12, 9]
```

```
plt.bar(x, y)          # Create a bar plot
```

```
# Add labels and title
```

```
plt.xlabel('Categories')
```

```
plt.ylabel('Values')
```

```
plt.title('Bar Plot')
```

```
# Display the plot      plt.show()
```

In this example, we import the 'pyplot' module from the Matplotlib library as 'plt'. We then define the data for the x-axis ('x') and the y-axis ('y').

Next, we use the 'plt.bar()' function to create the bar plot by passing the 'x' and 'y' data. This function plots vertical bars representing the values of each category.

We can add labels to the x-axis and y-axis using the 'plt.xlabel()' and 'plt.ylabel()' functions, respectively. The 'plt.title()' function is used to set the title of the plot.

Finally, we use 'plt.show()' to display the plot.

How to create Scatter Plot using Matplotlib? Explain with example?

To create a scatter plot using Matplotlib, you can use the 'matplotlib.pyplot.scatter()' function. Here's an example:

```
import matplotlib.pyplot as plt

# Data for x-axis and y-axis
x = [1, 2, 3, 4, 5]
y = [2, 4, 6, 8, 10]

plt.scatter(x, y)      # Create a scatter plot

# Add labels and title
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.title('Scatter Plot')

plt.show()             # Display the plot
```

In this example, we import the 'pyplot' module from the Matplotlib library as 'plt'. We then define the data for the x-axis ('x') and the y-axis ('y').

Next, we use the 'plt.scatter()' function to create the scatter plot by passing the 'x' and 'y' data. This function plots individual data points as separate markers.

We can add labels to the x-axis and y-axis using the 'plt.xlabel()' and 'plt.ylabel()' functions, respectively. The 'plt.title()' function is used to set the title of the plot.

Finally, we use 'plt.show()' to display the plot.

How to create Histogram using Matplotlib? Explain with example?

To create a histogram using Matplotlib, you can use the 'matplotlib.pyplot.hist()' function. Here's an example:

```
Import matplotlib.pyplot as plt
```

```
Data = [2, 2, 3, 4, 4, 4, 5, 5, 6, 7, 8, 8, 9, 9, 9]          # Data for the histogram
```

```
Plt.hist(data)          # Create a histogram
```

```
# Add labels and title
```

```
Plt.xlabel('Values')
```

```
Plt.ylabel('Frequency')
```

```
Plt.title('Histogram')
```

```
# Display the histogram
```

```
Plt.show()
```



In this example, we import the 'pyplot' module from the Matplotlib library as 'plt'. We then define the data for the histogram in the 'data' list.

Next, we use the 'plt.hist()' function to create the histogram by passing the 'data'. This function automatically calculates the frequency of the values and creates the histogram bins.

We can add labels to the x-axis and y-axis using the 'plt.xlabel()' and 'plt.ylabel()' functions, respectively. The 'plt.title()' function is used to set the title of the histogram.

Finally, we use 'plt.show()' to display the histogram.

How to create Boxplot using Matplotlib? Explain with example?

To create a boxplot using Matplotlib, you can use the 'matplotlib.pyplot.boxplot()' function. Here's an example:

```
Import matplotlib.pyplot as plt
```

```
# Data for the boxplot
```

```
Data = [25, 30, 34, 38, 42, 45, 49, 50, 52, 56, 58, 60, 64]
```

```
Plt.boxplot(data)      # Create a boxplot
```

```
# Add labels and title
```

```
Plt.xlabel('Data')
```

```
Plt.ylabel('Values')
```

```
Plt.title('Boxplot')
```

```
# Display the boxplot
```

```
Plt.show()
```

In this example, we import the 'pyplot' module from the Matplotlib library as 'plt'. We then define the data for the boxplot in the 'data' list.

Next, we use the 'plt.boxplot()' function to create the boxplot by passing the 'data'. This function displays the distribution of the data using quartiles, a box, and whiskers.

We can add labels to the x-axis and y-axis using the 'plt.xlabel()' and 'plt.ylabel()' functions, respectively. The 'plt.title()' function is used to set the title of the boxplot.

Finally, we use 'plt.show()' to display the boxplot.

How to create Pair plot Matplotlib? Explain with example?

```
import seaborn as sns  
  
# Load the example iris dataset  
iris = sns.load_dataset('iris')  
  
# Create a pair plot  
sns.pairplot(iris)  
  
# Display the pair plot  
plt.show()
```

How to create Heat map using Matplotlib? Explain with example?

```
Import matplotlib.pyplot as plt  
Import numpy as np  
  
# Generate a random 2D array  
Data = np.random.rand(5, 5)  
  
Plt.imshow(data, cmap='hot')  
  
# Add colorbar  
Plt.colorbar()  
  
# Display the heatmap  
Plt.show()
```

What are the different library for data visualization in Python

1. **Matplotlib:** A versatile plotting library for creating static, interactive, and publication-quality visualizations.
2. **Seaborn:** A high-level library built on top of Matplotlib that simplifies the creation of visually appealing statistical graphics.
3. **Plotly:** A library for creating interactive and dynamic visualizations, including charts, graphs, and maps.
4. **Bokeh:** A Python library for interactive visualization that focuses on creating web-based plots with interactive features.
5. **Altair:** A declarative statistical visualization library that allows we to build visualizations using a concise and intuitive syntax.

Pandas library In Python

What is pandas?

Pandas is a Python library used for data manipulation and analysis. It provides different data structures and functions to work with structured data, including tabular and time series data.

What is use of Pandas? Explain with example?

Pandas provides different data structures and functions to work with structured data, including tabular and time series data.

Example :

```
import pandas as pd
```

```
# Create a DataFrame
```

```
data = {'Name': ['John', 'Jane', 'Alice'],  
        'Age': [25, 30, 35],  
        'City': ['New York', 'London', 'Paris']}
```

```
df = pd.DataFrame(data)
```



```
# Print the DataFrame
```

```
print(df)
```

Output :

```
   Name  Age  City
0  John   25 New York
1  Jane   30  London
2  Alice  35   Paris
```

In this example, we create a DataFrame with three columns: Name, Age, and City. We then print the DataFrame, which displays the data in a tabular format. Pandas provides a convenient way to create, manipulate, and analyze structured data like this.

what is meant by data frame

DataFrame is a two-dimensional data structure in Python provided by the Pandas library.

It represents data in a tabular format, similar to a spreadsheet, and allows for efficient manipulation and analysis of data.

How to create data frame using Pandas? Explain with example?

Example :

```
import pandas as pd
```

```
# Create a DataFrame
```

```
data = {'Name': ['John', 'Jane', 'Alice'],
        'Age': [25, 30, 35],
        'City': ['New York', 'London', 'Paris']}
```

```
df = pd.DataFrame(data)
```

```
# Print the DataFrame
```

```
print(df)
```

Output :

```
   Name  Age  City
0  John   25 New York
1  Jane   30  London
2  Alice  35   Paris
```

In this example, we create a DataFrame with three columns: Name, Age, and City. We then print the DataFrame, which displays the data in a tabular format. Pandas provides a convenient way to create, manipulate, and analyze structured data like this.

How to import & export csv file using Pandas? Explain with example?

1. Importing CSV File:

To import a CSV file into a Pandas DataFrame, use `pd.read_csv('filename.csv')`.

example:

```
import pandas as pd

# Import CSV file into a DataFrame

df = pd.read_csv('data.csv')
```

2. Exporting DataFrame to CSV:

To export a Pandas DataFrame to a CSV file, use `df.to_csv('filename.csv', index=False)`.

Example:

```
import pandas as pd
```

```
# Create a DataFrame
```

```
data = {'Name': ['John', 'Jane', 'Alice'],  
        'Age': [25, 30, 35],  
        'City': ['New York', 'London', 'Paris']}
```

```
df = pd.DataFrame(data)
```

```
# Export DataFrame to CSV file
```

```
df.to_csv('output.csv', index=False)
```

In these examples, replace `filename.csv` with the name of the CSV file you want to import or export.

How to import & export excel file using Pandas? Explain with example?

1. Importing Excel File:

To import an Excel file into a Pandas DataFrame, use `pd.read_excel('filename.xlsx', sheet_name='sheetname')`.

Example:

```
import pandas as pd
```

```
# Import Excel file into a DataFrame
```

```
df = pd.read_excel('data.xlsx', sheet_name='Sheet1')
```

2. Exporting DataFrame to Excel:

To export a Pandas DataFrame to an Excel file, use `df.to_excel('filename.xlsx', sheet_name='sheetname', index=False)`.

Example:

```
import pandas as pd
```

```
# Create a DataFrame
```

```
data = {'Name': ['John', 'Jane', 'Alice'],
```

```
'Age': [25, 30, 35],  
'City': ['New York', 'London', 'Paris']}]  
  
df = pd.DataFrame(data)  
  
# Export DataFrame to Excel file  
  
df.to_excel('output.xlsx', sheet_name='Sheet1', index=False)
```

In these examples, replace ``filename.xlsx`` with the name of the Excel file you want to import or export, and ``sheetname`` with the specific sheet name within the Excel file.

How can you read a CSV file in Python using pandas?

example of reading a CSV file using Pandas:

```
import pandas as pd  
  
# Read CSV file into a DataFrame  
df = pd.read_csv('data.csv')  
  
# Display the DataFrame  
print(df)
```

In this example, we import the Pandas library using `import pandas as pd`. Then, we use the `read_csv()` function to read the CSV file named `data.csv` into a Pandas DataFrame named `df`. Finally, we print the DataFrame using `print(df)`.

What are some file formats supported by pandas for importing data?

Pandas supports importing data from the following file formats:

1. CSV (Comma-Separated Values)
2. Excel (.xls, .xlsx)
3. JSON (JavaScript Object Notation)

4. SQL Databases

5. HTML tables

6. Parquet

7. Feather

Pandas provides specific functions like ``read_csv()``, ``read_excel()``, ``read_json()``, ``read_sql()``, ``read_html()``, ``read_parquet()``, and ``read_feather()`` to import data from these file formats.

how to get a index of the data frame write syntax for it

To get the index of a Pandas DataFrame, we can use the 'index' attribute: `'df.index'`.

write syntax to get a column level of the data frame

to get a column level of a Pandas DataFrame, we can use the syntax `'df['column_name']'`, where "column_name" is the name of the desired column.

write syntax for to get total number of element from the data frame

to get the total number of elements in a Pandas DataFrame, we can use the 'size' attribute: `'df.size'`.

What is data Wrangling in Python ?

data wrangling in Python refers to the process of cleaning, transforming, and preparing raw data for analysis. It involves tasks like handling missing values, removing duplicates, reshaping data, and merging datasets.

which operator is used for indexing and selecting data in pandas

Python slicing operator '[']' and attribute/ dot operator '.' are used for indexing. It Provides Quick and easy access to pandas data structures

Data Normalization In Python

What is data normalization?

Data normalization is the process of standardizing numerical data to a similar scale. It eliminates bias caused by variables with different magnitudes and improves model accuracy. Methods include min-max scaling, z-score standardization, and decimal scaling.

Which methods are used in python for data normalization?

data normalization can be performed using the following methods:

1. Min-Max Scaling:

```
from sklearn.preprocessing import MinMaxScaler  
scaler = MinMaxScaler()    # Create the scaler  
scaled_data = scaler.fit_transform(data)    # Fit and transform the data
```

this scales the data to a specified range, often between 0 and 1.

2. Z-Score Standardization:

```
from sklearn.preprocessing import StandardScaler  
scaler = StandardScaler()    # Create the scaler  
scaled_data = scaler.fit_transform(data)    # Fit and transform the data
```

This standardizes the data to have a mean of 0 and a standard deviation of 1.

what is binning in python? Why it is used?

Binning in Python is the process of categorizing continuous data into discrete bins or intervals. It is used to simplify the data, create new features, handle outliers, and enable visual representation. In short, binning helps organize continuous data into meaningful categories.

How to find standard deviation & variance of given data in python?

example of how to find the standard deviation and variance of given data in Python:

```
import numpy as np  
data = np.array([3, 4, 5, 6, 7, 8, 9])  
std_dev = np.std(data)  
variance = np.var(data)
```

In this example, the `np.array()` function is used to create a NumPy array containing the data. The `np.std()` function calculates the standard deviation of the data, and the `np.var()` function calculates the variance. The results are stored in the variables `std_dev` and `variance`, respectively.

What is Hypothesis?

A hypothesis is a proposed explanation or statement about a population or phenomenon. It consists of the null hypothesis (no difference/relationship) and the alternative hypothesis (specific difference/relationship). Hypotheses are tested using data and evidence.

What is Null Hypothesis?

the null hypothesis is a statement that assumes no significant difference, effect, or relationship between variables in a population. It

serves as the default position to be tested against an alternative hypothesis.

What is use of scipy module in python?

the `scipy` module in Python is used for scientific and technical computing tasks such as numerical integration, optimization, interpolation, linear algebra, signal and image processing, and statistics.

Exploratory Data Analysis

What is EDA?

EDA (Exploratory Data Analysis) is an approach to analyze and summarize a dataset by using statistical and visualization techniques. Its main objectives are to understand the data and identify patterns, relationships, and outliers.

What are some common techniques used in EDA?

common techniques used in Exploratory Data Analysis

1. Data visualization: Creating visual representations like histograms, scatter plots, and box plots to explore the distribution, relationships, and patterns in the data.
2. Correlation analysis: Examining the strength and direction of relationships between variables using correlation matrices or heatmaps.
3. Missing data analysis: Identifying missing values and understanding their patterns to determine appropriate strategies for handling them.

These techniques help in understanding data characteristics, and formulating further analysis or hypotheses.

what is probability

Probability is the measure of how likely an event is to occur, represented by a number between 0 and 1.

what are the basic rules of the probability

1. Probability is a number between 0 and 1 that represents the likelihood of an event occurring.
2. The complement rule states that the probability of an event not happening is equal to 1 minus the probability of the event happening.

what is probability distributions

Probability distributions represent the likelihood of different outcomes and can be discrete or continuous.

what is bootstrapping

Bootstrapping is a statistical technique that involves resampling the data to estimate uncertainty by creating multiple datasets through random sampling with replacement.

Correlation In Python

what is correlation?

Correlation is a statistical measure of the relationship between two variables, expressed as a correlation coefficient. It indicates whether the variables move together (positive correlation), move in opposite directions (negative correlation), or have no linear relationship (zero correlation).

What is correlation coefficient?

A correlation coefficient is a number between -1 and +1 that measures the strength and direction of the relationship between two variables.

What is Pearson's correlation coefficient equation?

The equation for Pearson's correlation coefficient r is:

$$\rho_{X,Y} = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y}$$

Where, cov is the covariance, σ_x is the standard deviation of X and σ_y is the standard deviation of Y.

Cov(X,Y) is computed as

$$\text{Cov}(X, Y) = \frac{\sum (X - \bar{X})(Y - \bar{Y})}{n-1}$$

What is meant by covariance

Covariance measures how two variables vary together. A positive covariance means they move in the same direction, while a negative covariance means they move in opposite directions.

What is significance of each correlation

The significance of a correlation indicates whether the observed correlation is likely to be real or due to chance. It is determined through statistical tests using a predetermined threshold (p-value). If the p-value is below the threshold, the correlation is considered statistically significant, indicating a genuine relationship between the variables.

What is groupby() operation in python ?

'groupby()' is a function in Python's pandas library that enables grouping data based on specific criteria. It allows for performing aggregate computations or analysis on each group.

The groupby() methods allows we to group we r data and create a function on this groups

What is splitting applying and combining ?

'groupby()' in pandas is used to group data based on specified criteria and apply functions to these groups for analysis and summary purposes.

How do we group by a list in Python

we cannot directly use the 'groupby()' function from pandas on a list. However, we can use Python's 'itertools.groupby()' function to achieve similar functionality and group elements of a list based on a key function.

What is ANOVA ?

ANOVA (Analysis of Variance) is a statistical test used to compare the means of three or more groups to determine if there are significant differences between them.

Why we use ANOVA ?

We use ANOVA (Analysis of Variance) to compare the means of three or more groups efficiently and determine if there are significant differences among them.

How the ANOVA is work ?

ANOVA works by comparing the variation between groups to the variation within groups to determine if the group means are significantly different.

Regression Analysis In Python

What is use of Regression analysis?

In other words, regression analysis is used to predict future values, understand how variables are related, select important variables, test

hypotheses, and make informed decisions. It helps in modeling and analyzing the relationship between a dependent variable and one or more independent variables.

Which are types of Regression?

1. Linear Regression: Models the relationship between the dependent variable and independent variables using a straight line equation.
2. Logistic Regression: Models the probability of a binary or categorical outcome using a logistic function.
3. Polynomial Regression: Models non-linear relationships by fitting a polynomial function to the data.
4. Ridge Regression: Performs linear regression with a penalty term to address multicollinearity among independent variables.

What is linear regression and multiple linear regression ?

Linear regression is a technique that establishes a linear relationship between a dependent variable and one or more independent variables.

Multiple linear regression extends this to include multiple independent variables. The goal is to estimate the coefficients of the linear equation that best fits the data.

Which are different parameters used for evaluation of regression analysis?

Parameter used for the evaluation of regression analysis:

1. Mean Squared Error (MSE): Average squared difference between predicted and actual values.
2. Root Mean Squared Error (RMSE): Square root of MSE, represents average error in the same units as the dependent variable.
3. Mean Absolute Error (MAE): Average absolute difference between predicted and actual values.

4. R-squared (R^2) coefficient: Proportion of variance in the dependent variable explained by independent variables.
5. Adjusted R-squared: R-squared adjusted for the number of independent variables and sample size.
6. F-statistic: Overall significance of the regression model, comparing explained and unexplained variance.

What is polynomial regression

polynomial regression is a type of regression analysis that models the relationship between variables using a polynomial function instead of a straight line. It allows for curve fitting by including polynomial terms of higher degree.

Explain difference between linear & logistic regression?

Basis	Linear Regression	Logistic Regression
Core Concept	The data is modelled using a straight line	The probability of some obtained event is represented as a linear function of a combination of predictor variables.
Used with	Continuous Variable	Categorical Variable
Output/Prediction	Value of the variable	Probability of occurrence of event
Accuracy and Goodness of fit	measured by loss, R squared, Adjusted R squared etc.	Accuracy, Precision, Recall, F1 score, ROC curve, Confusion Matrix, etc

In simple language , the main difference between linear regression and logistic regression is the type of dependent variable they can handle. Linear regression is used for continuous variables, while logistic regression is used for categorical variables.

what are the difference between correlation and regression ?

Basis For Comparison	Correlation	Regression
Meaning	Correlation is a statistical measure that determines the association between two variables.	Regression describes how to numerically relate an independent variable to the dependent variable.
Usage	To represent a linear relationship between two variables.	To fit the best line and to estimate one variable based on another.
Dependent and Independent Variables	No Difference.	Both variables are different.
Objective	To find a numerical value expressing the relationship between the variables.	To estimate values of random variables based on the values of fixed variables.

In simple language, correlation measures the relationship between variables, while regression predicts or estimates one variable based on other variables. Correlation can be used for any type of variables, while regression typically involves at least one continuous dependent variable and one or more independent variables.

★★★

Best Of Luck

