

Visualizing Data Distributions and Model Results using Matplotlib and Seaborn

Table of Contents

- [Description](#)
 - [Problem Statement](#)
 - [Prerequisites](#)
 - [Software Required](#)
 - [Hardware Requirements](#)
 - [Setup Instructions](#)
 - [Step 1: Install Python and Required Libraries](#)
 - [Step 2: Install Matplotlib and Seaborn](#)
 - [Step 3: Verify Library Installation](#)
 - [Visualizing Data with Matplotlib](#)
 - [Plotting Simple Graphs](#)
 - [Customizing Plots](#)
 - [Visualizing Data Distributions](#)
 - [Visualizing Data with Seaborn](#)
 - [Plotting with Seaborn](#)
 - [Customizing Seaborn Plots](#)
 - [Seaborn Distribution Plots](#)
 - [Model Results Visualization](#)
 - [Visualizing Model Performance Metrics](#)
 - [Plotting Confusion Matrix](#)
 - [Plotting ROC Curve](#)
 - [Reference](#)
-

Description

This lab guide provides a detailed approach to visualizing data distributions and model results using **Matplotlib** and **Seaborn**, two popular Python libraries for creating graphs and visualizations. These tools are essential for gaining insights from data and evaluating machine learning model performance.

Problem Statement

Data visualization is a key component of data analysis and machine learning model evaluation. This guide helps users learn how to visualize data distributions and model results effectively using Matplotlib and Seaborn, which are crucial for understanding trends, patterns, and model performance.

Prerequisites

Completion of all previous lab guides (up to Lab Guide-07) is required before proceeding with Lab Guide-08.

Software Required

- **Python:** Python 3.11.9
- **Visual Studio Code (VSCode):** A lightweight code editor that provides powerful features for Python development, including extensions for linting, debugging, and version control.
- **Libraries:** [NumPy](#), [Pandas](#), [Matplotlib](#), [Seaborn](#), [Scikit-learn](#)

Hardware Requirements

- Minimum 4GB RAM.
- At least 1GB of free disk space.
- A GPU (optional, but recommended for faster training).

Setup Instructions

Step 1: Install Python and VSCode

Install Python:

You can download and install Python 3.11.9 from the official Python website:

- Visit the [official Python website](#).
- Locate a reliable version of Python 3, "**Download Python 3.11.9**".
- Choose the correct link for your device from the options provided: either Windows installer (64-bit) or Windows installer (32-bit) and proceed to download the executable file.



Install Visual Studio Code (VSCode):

Download and install VSCode from the official Visual Studio Code website: [Download Visual Studio Code](#)

Step 2: Install Matplotlib and Seaborn

1. Open **Terminal** in VSCode.
2. Install the necessary libraries using **pip**.

Using pip

```
pip install matplotlib
```

```
PS C:\Users\Administrator\Desktop\AIML> pip install pandas numpy matplotlib seaborn scikit-learn
Collecting pandas
  Using cached pandas-2.2.3-cp311-cp311-win_amd64.whl.metadata (19 kB)
Collecting numpy
  Using cached numpy-2.1.2-cp311-cp311-win_amd64.whl.metadata (59 kB)
Collecting matplotlib
  Using cached matplotlib-3.9.2-cp311-cp311-win_amd64.whl.metadata (11 kB)
Collecting seaborn
  Using cached seaborn-0.13.2-py3-none-any.whl.metadata (5.4 kB)
Collecting scikit-learn
  Using cached scikit_learn-1.5.2-cp311-cp311-win_amd64.whl.metadata (13 kB)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\program files\python311\lib\site-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\program files\python311\lib\site-packages (from pandas) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in c:\program files\python311\lib\site-packages (from pandas) (2024.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\program files\python311\lib\site-packages (from matplotlib) (1.3.0)
Requirement already satisfied: cycler>=0.10 in c:\program files\python311\lib\site-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in c:\program files\python311\lib\site-packages (from matplotlib) (4.54.1)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\program files\python311\lib\site-packages (from matplotlib) (1.4.7)
Requirement already satisfied: packaging>=20.0 in c:\program files\python311\lib\site-packages (from matplotlib) (24.1)
Requirement already satisfied: pillow>=8 in c:\program files\python311\lib\site-packages (from matplotlib) (11.0.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\program files\python311\lib\site-packages (from matplotlib) (3.2.0)
Requirement already satisfied: scipy>=1.6.0 in c:\program files\python311\lib\site-packages (from scikit-learn) (1.14.1)
Requirement already satisfied: joblib>=1.2.0 in c:\program files\python311\lib\site-packages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in c:\program files\python311\lib\site-packages (from scikit-learn) (3.5.0)
Requirement already satisfied: six>=1.5 in c:\program files\python311\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
Using cached pandas-2.2.3-cp311-cp311-win_amd64.whl (11.6 MB)
Using cached numpy-2.1.2-cp311-cp311-win_amd64.whl (12.9 MB)
Using cached matplotlib-3.9.2-cp311-cp311-win_amd64.whl (7.8 MB)
Using cached seaborn-0.13.2-py3-none-any.whl (294 kB)
Using cached scikit_learn-1.5.2-cp311-cp311-win_amd64.whl (11.0 MB)
Installing collected packages: numpy, pandas, scikit-learn, matplotlib, seaborn
Successfully installed matplotlib-3.9.2 numpy-2.1.2 pandas-2.2.3 scikit-learn-1.5.2 seaborn-0.13.2
```

Activate Windows
Go to Settings to activate Windows.

```
pip install seaborn
```

```
PS C:\Users\Administrator\Desktop\AIML> pip install matplotlib seaborn
Collecting matplotlib
  Using cached matplotlib-3.9.2-cp311-cp311-win_amd64.whl.metadata (11 kB)
Collecting seaborn
  Using cached seaborn-0.13.2-py3-none-any.whl.metadata (5.4 kB)
Requirement already satisfied: contourpy>=1.0.1 in c:\program files\python311\lib\site-packages (from matplotlib) (1.3.0)
Requirement already satisfied: cycler>=0.10 in c:\program files\python311\lib\site-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in c:\program files\python311\lib\site-packages (from matplotlib) (4.54.1)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\program files\python311\lib\site-packages (from matplotlib) (1.4.7)
Requirement already satisfied: numpy>=1.23 in c:\program files\python311\lib\site-packages (from matplotlib) (2.1.2)
Requirement already satisfied: packaging>=20.0 in c:\program files\python311\lib\site-packages (from matplotlib) (24.1)
Requirement already satisfied: pillow>=8 in c:\program files\python311\lib\site-packages (from matplotlib) (11.0.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\program files\python311\lib\site-packages (from matplotlib) (3.2.0)
Requirement already satisfied: python-dateutil>=2.7 in c:\program files\python311\lib\site-packages (from matplotlib) (2.9.0.post0)
Requirement already satisfied: pandas>=1.2 in c:\program files\python311\lib\site-packages (from seaborn) (2.2.3)
Requirement already satisfied: pytz>=2020.1 in c:\program files\python311\lib\site-packages (from pandas) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in c:\program files\python311\lib\site-packages (from pandas) (2024.2)
Requirement already satisfied: six>=1.5 in c:\program files\python311\lib\site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)
Using cached matplotlib-3.9.2-cp311-cp311-win_amd64.whl (7.8 MB)
Using cached seaborn-0.13.2-py3-none-any.whl (294 kB)
Installing collected packages: matplotlib, seaborn
Successfully installed matplotlib-3.9.2 seaborn-0.13.2
```

Step 3: Verify Library Installation

```
import matplotlib.pyplot as plt
import seaborn as sns
```

If there are no error messages, the libraries have been successfully installed.

Visualizing Data with Matplotlib

- Create a folder called **Matplotlib** in your **VScode**.
- Create a new file
 - Create a Python file named **data_visualization.py** inside your **Matplotlib** folder and add the following code.

Plotting Simple Graphs

Create a basic line plot using **Matplotlib**:

```
import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5]
y = [10, 20, 25, 30, 35]

plt.plot(x, y)
plt.title('Simple Line Plot')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.show()
```

Run the Python file

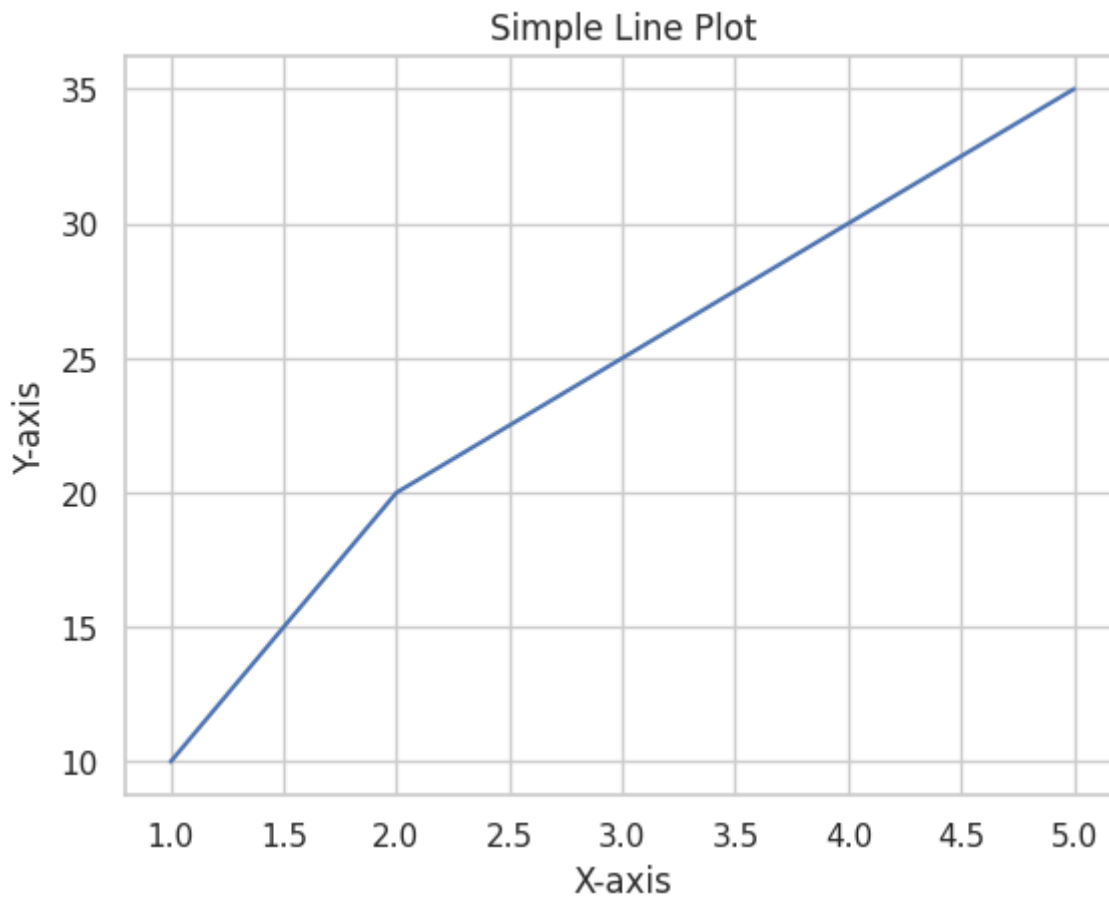
- Use the command below in your terminal to run the Python file:

```
cd Matplotlib
python data_visulization.py
```

```
PS C:\Users\Administrator\Desktop\AIML> cd Matplotlib
PS C:\Users\Administrator\Desktop\AIML\Matplotlib>
```

```
PS C:\Users\Administrator\Desktop\AIML\Matplotlib> python data_visulization.py
```

Output



Customizing Plots

Customize plot elements like color, style, and labels:

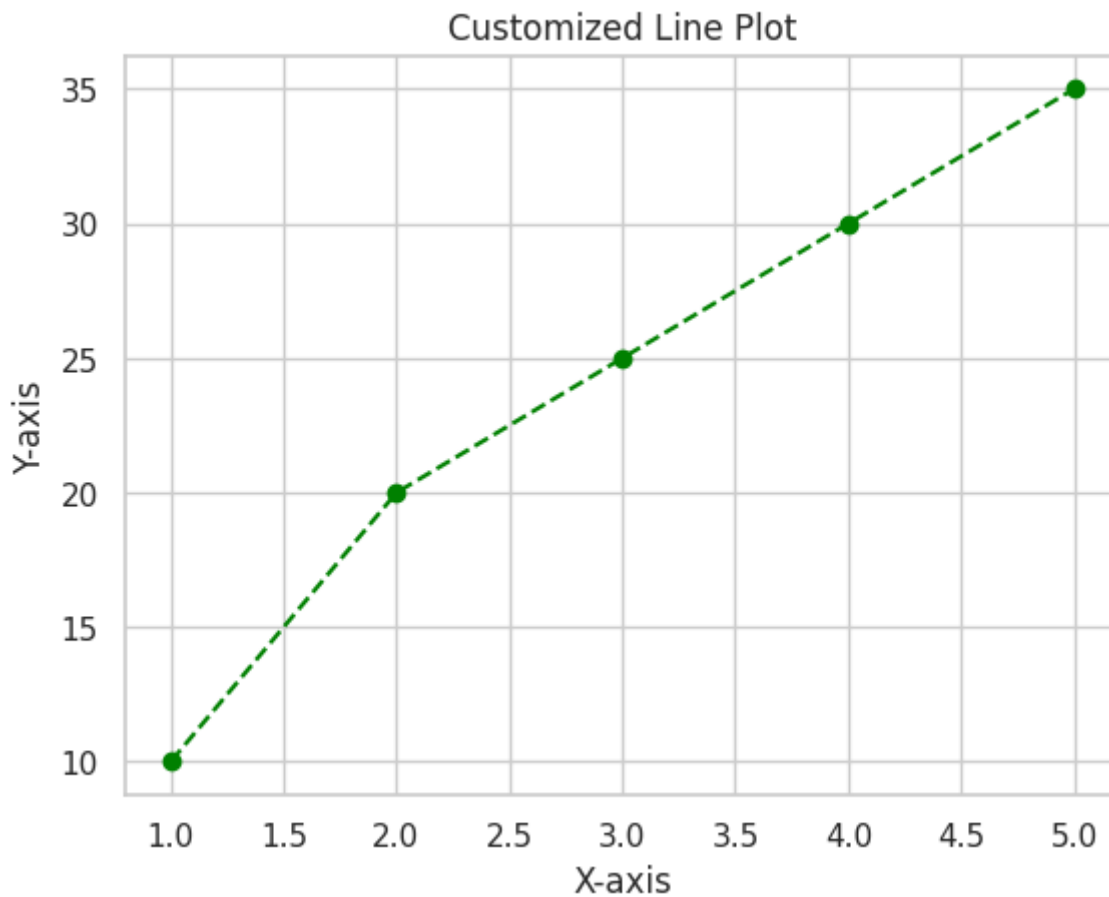
```
plt.plot(x, y, color='green', linestyle='--', marker='o')
plt.title('Customized Line Plot')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.grid(True)
plt.show()
```

Run the Python file

- Use the command below in your terminal to run the Python file:

```
python data_visulization.py
```

Output



Visualizing Data Distributions

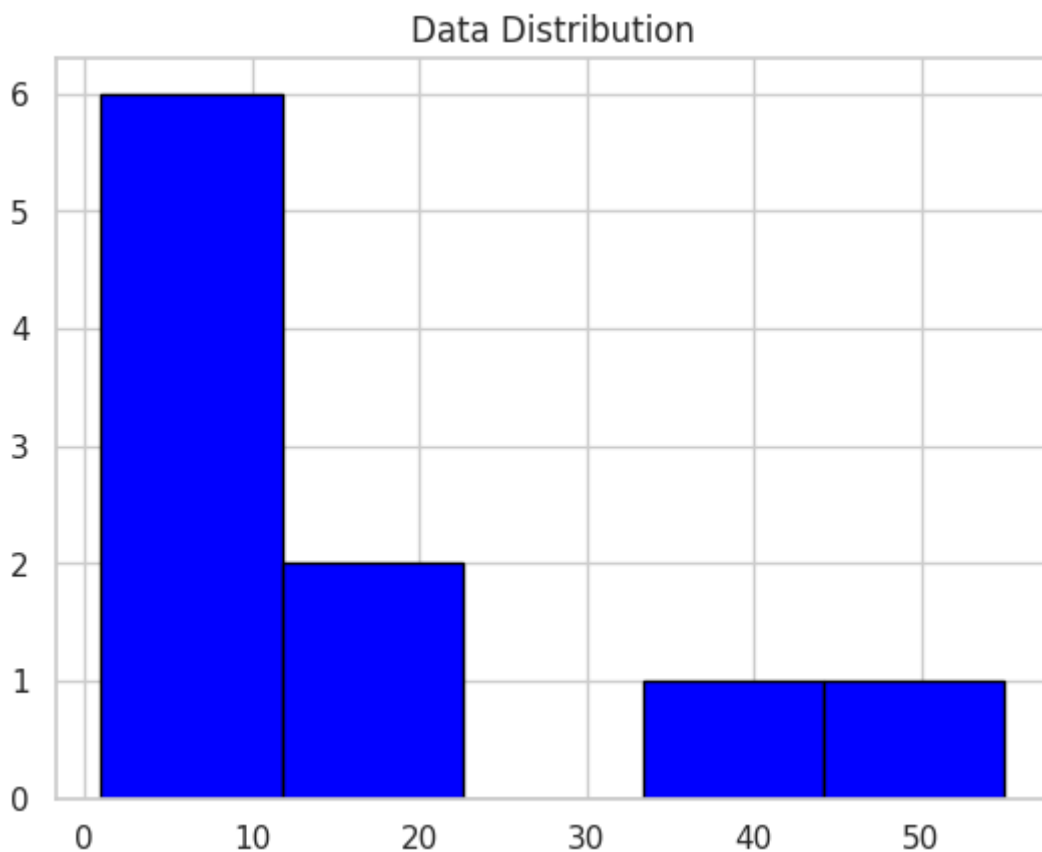
Use histograms to visualize the distribution of data:

```
data = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55]
plt.hist(data, bins=5, color='blue', edgecolor='black')
plt.title('Data Distribution')
plt.show()
```

Run the Python file

- Use the command below in your terminal to run the Python file:

```
python data_visulization.py
```



Output

Visualizing Data with Seaborn

- Create a folder called **Seaborn** in your **VSCode**.
- Create a new file
 - Create a Python file named **data_visualization.py** inside your **Seaborn** folder and add the following code.

Plotting with Seaborn

Create a simple scatter plot using Seaborn:

```
import seaborn as sns
import matplotlib.pyplot as plt

# Load sample dataset
data = sns.load_dataset('iris')

sns.scatterplot(x='sepal_length', y='sepal_width', data=data)
plt.title('Sepal Length vs Sepal Width')
plt.show()
```

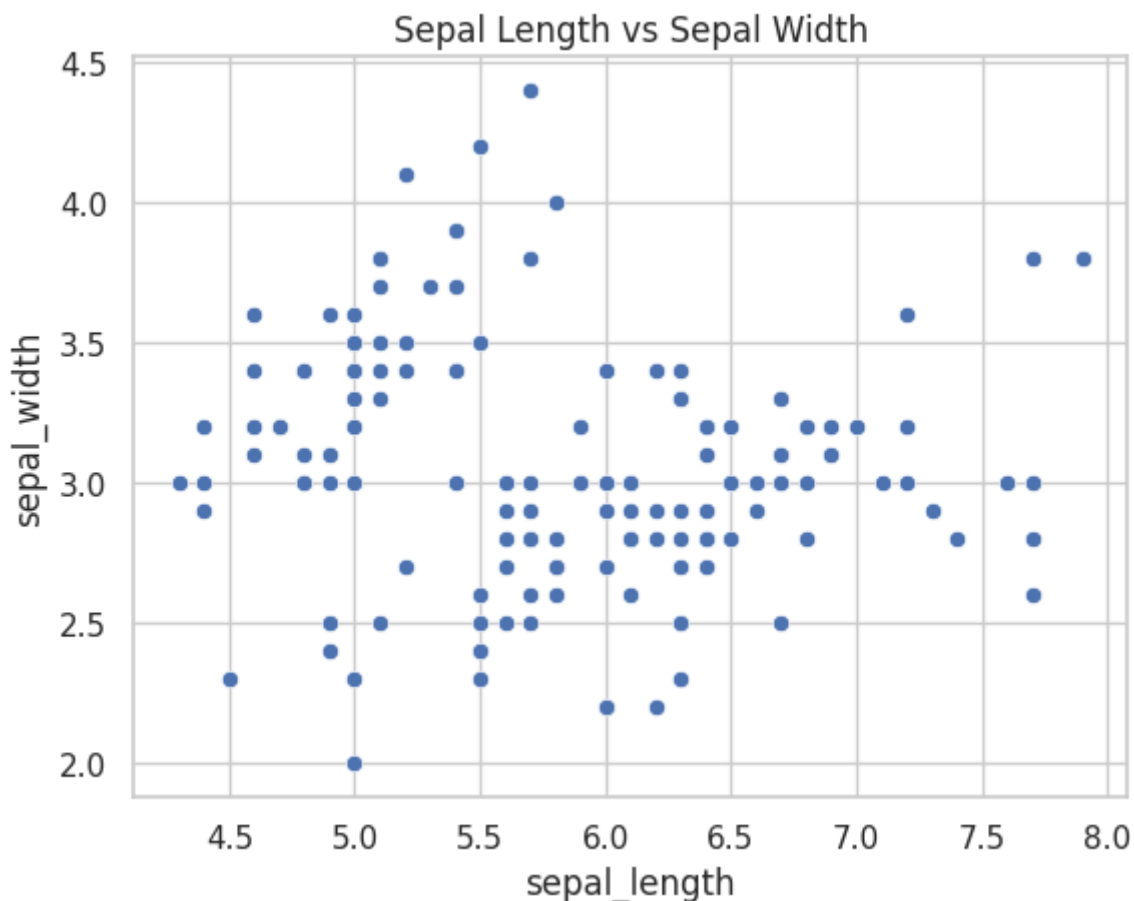
Run the Python file

- Use the command below in your terminal to run the Python file:

```
cd Seaborn
python data_visulization.py
```

```
PS C:\Users\Administrator\Desktop\AIML> cd Seaborn
PS C:\Users\Administrator\Desktop\AIML\Seaborn> python data_visulization.py
```

Output



Customizing Seaborn Plots

Seaborn provides high-level commands to easily customize plots:

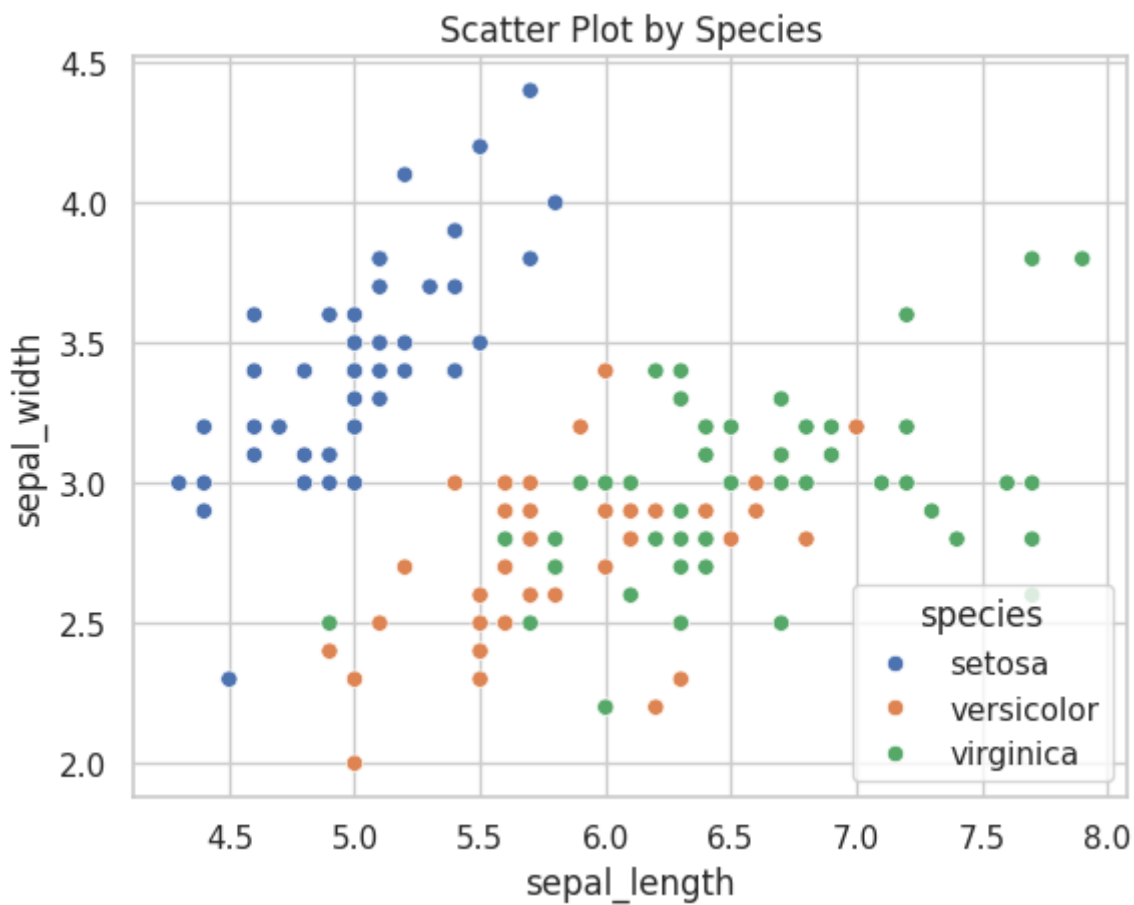
```
sns.set(style='whitegrid')
sns.scatterplot(x='sepal_length', y='sepal_width', hue='species', data=data)
plt.title('Scatter Plot by Species')
plt.show()
```

Run the Python file

- Use the command below in your terminal to run the Python file:

```
python data_visulization.py
```


Output



Seaborn Distribution Plots

Create distribution plots for visualizing univariate distributions:

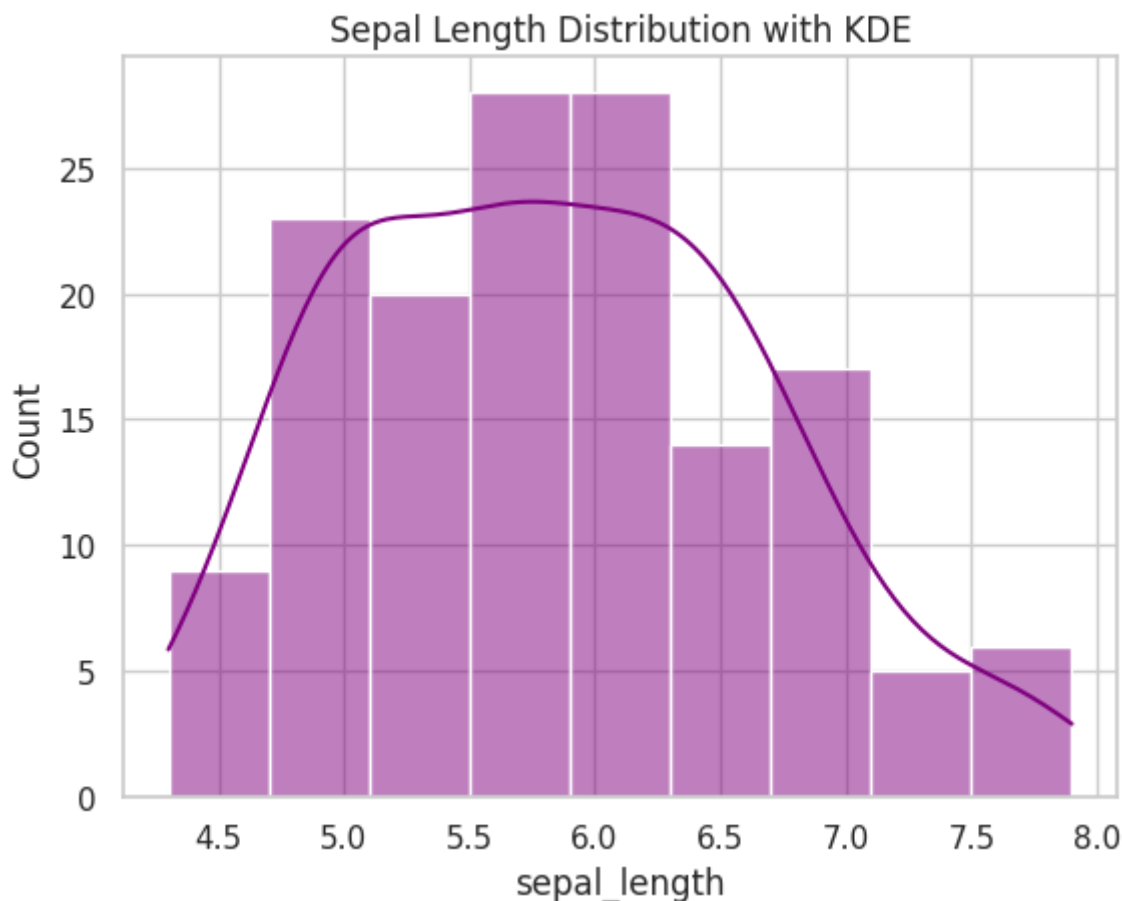
```
sns.histplot(data['sepal_length'], kde=True, color='purple')
plt.title('Sepal Length Distribution with KDE')
plt.show()
```

Run the Python file

- Use the command below in your terminal to run the Python file:

```
python data_visulization.py
```

Output



Model Results Visualization

- Create a new file
 - Create a Python file named `data_visualization.py` inside your `Matplotlib` folder and add the following code.

Visualizing Model Performance Metrics

Visualize model evaluation metrics like accuracy or loss over time:

```
import matplotlib.pyplot as plt
epochs = range(1, 11)
accuracy = [0.65, 0.72, 0.75, 0.78, 0.80, 0.82, 0.83, 0.85, 0.86, 0.87]

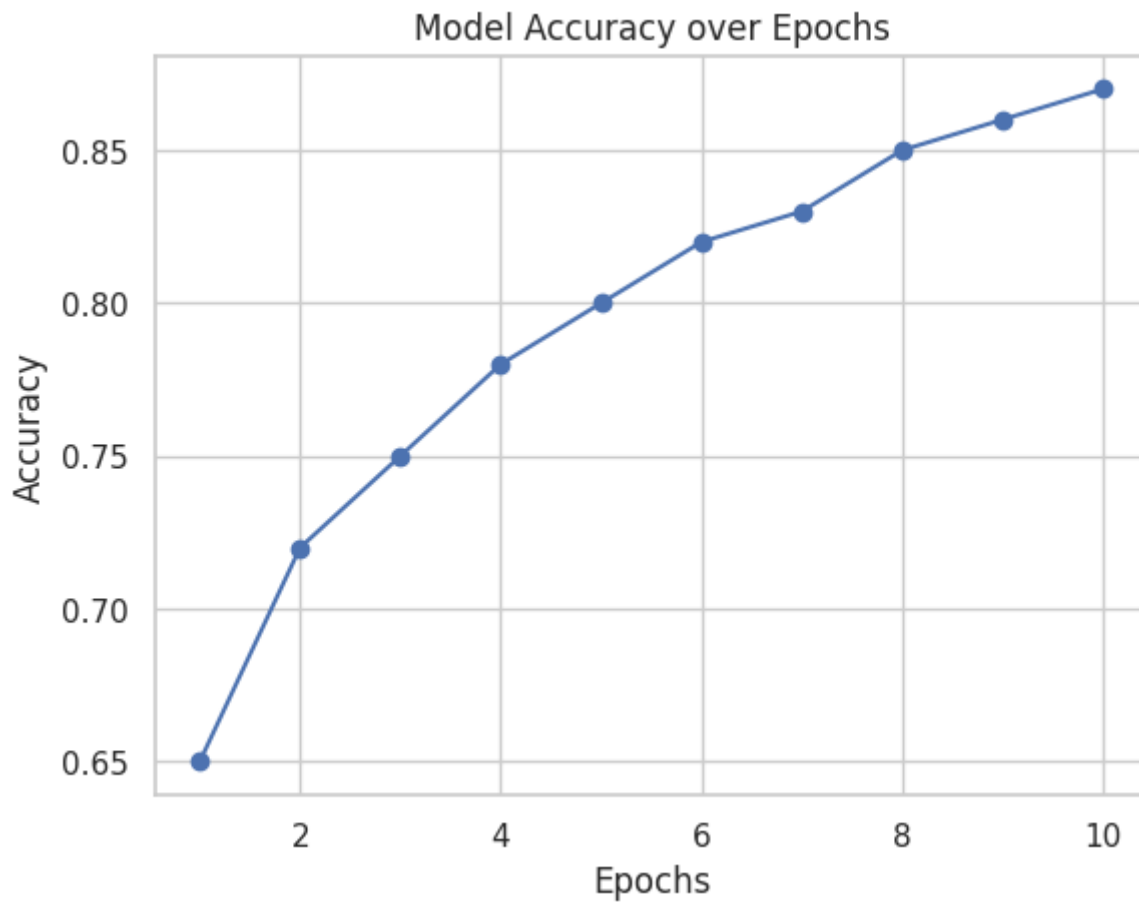
plt.plot(epochs, accuracy, marker='o', linestyle='--', color='b')
plt.title('Model Accuracy over Epochs')
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.show()
```

Run the Python file

- Use the command below in your terminal to run the Python file:

```
python data_visulization.py
```

Output



Plotting Confusion Matrix

Confusion matrix visualization helps evaluate classification performance:

```
from sklearn.metrics import confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt
# Example confusion matrix data
cm = confusion_matrix([1, 1, 0, 0], [1, 0, 1, 0])

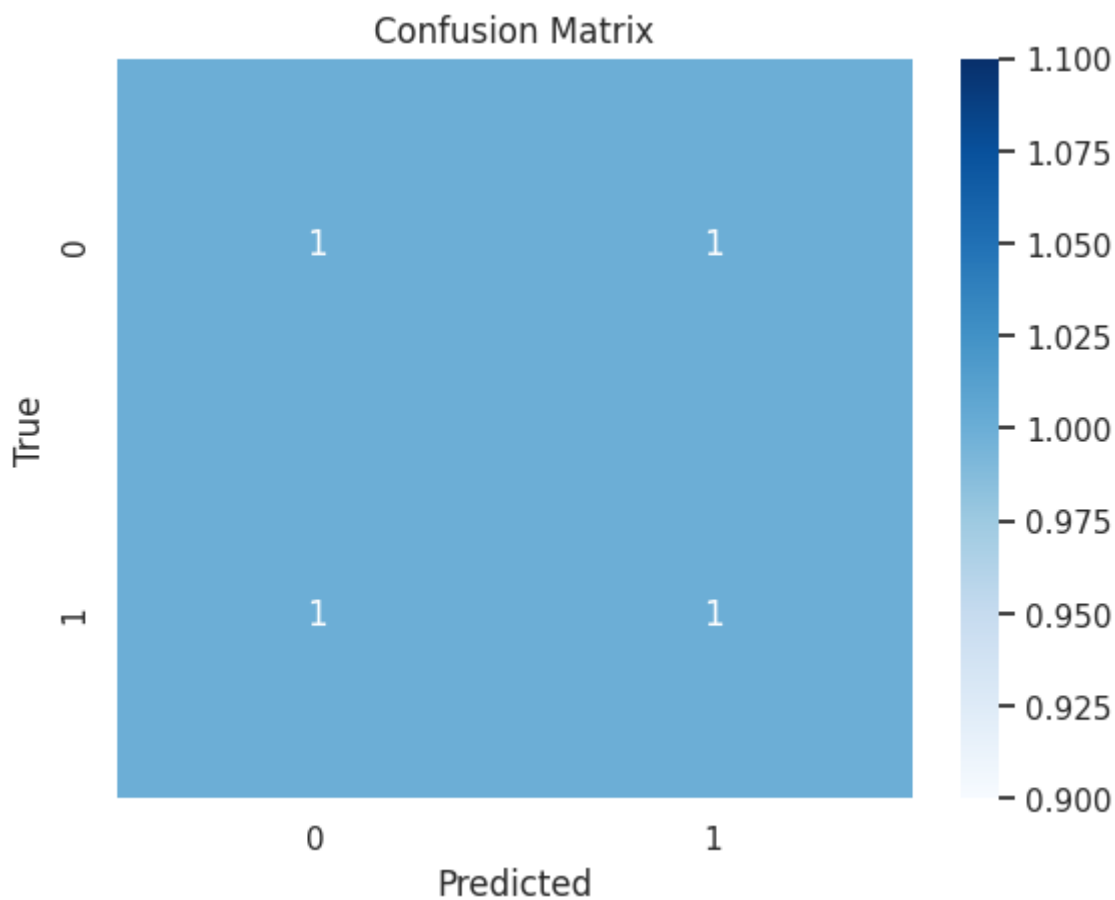
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues')
plt.title('Confusion Matrix')
plt.xlabel('Predicted')
plt.ylabel('True')
plt.show()
```

Run the Python file

- Use the command below in your terminal to run the Python file:

```
python data_visulization.py
```

Output



Plotting ROC Curve

Visualize the Receiver Operating Characteristic (ROC) curve:

```
from sklearn.metrics import roc_curve
from sklearn.metrics import confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt

# Example model predictions
fpr, tpr, thresholds = roc_curve([1, 1, 0, 0], [0.9, 0.8, 0.4, 0.2])

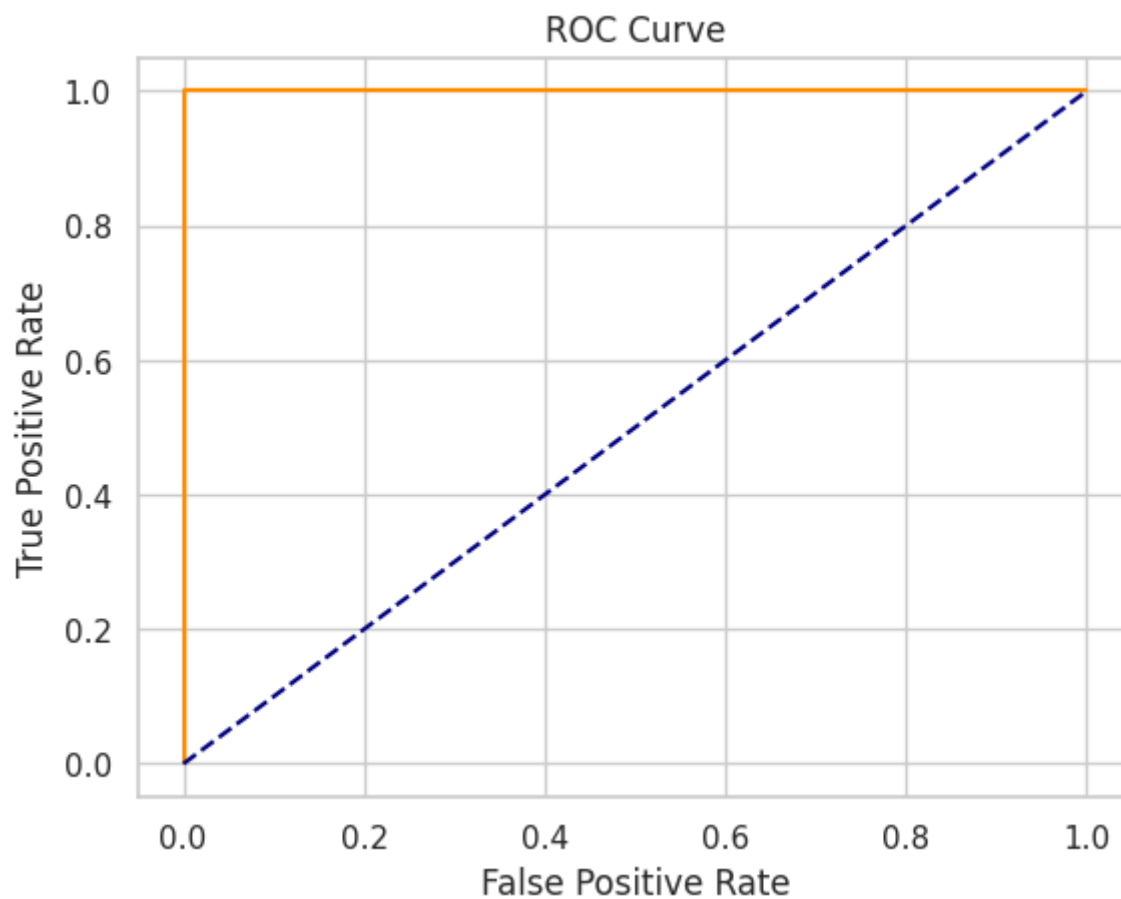
plt.plot(fpr, tpr, color='darkorange')
plt.plot([0, 1], [0, 1], color='navy', linestyle='--')
plt.title('ROC Curve')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.show()
```

Run the Python file

- Use the command below in your terminal to run the Python file:

```
python data_visulization.py
```

Output



Reference

- [Matplotlib Documentation](#)
- [Seaborn Documentation](#)
- [Scikit-learn Metrics](#)