The task involves loading and exploring the Iris dataset using Python's pandas library, followed by generating visualizations using Matplotlib and Seaborn. Key steps include displaying dataset information, descriptive statistics, and creating various plots to understand the distributions and relationships between features across different species.

WEEK 1

INTODUCTION TO ML

ML-WEEK 1.1

Noor Ul Ain

OBJECTIVE

Get acquainted with Python tools for data science and explore a basic dataset to understand key statistical concepts.

ACTIVITIES

1. Set up the environment: o Ensure Python and Jupyter Notebooks are installed. o Install necessary libraries: pandas, matplotlib, seaborn. To Load the dataset using pandas.

2. Data Exploration: to Display the first few rows of the dataset to understand its structure. o Use pandas functions like describe (), info (), and value_counts() to gather key statistics and information about the dataset.

3. Visualizations: to Create histograms for each numerical feature to understand distributions. o Use scatter plots to explore potential relationships between features. o Employ seaborn's pairplot to visualize the dataset with hue based on the species.

DATASET

Iris Dataset from Kaggle. This dataset includes 150 samples of iris flowers from three different species along with four features: sepal length, sepal width, petal length, and petal width.

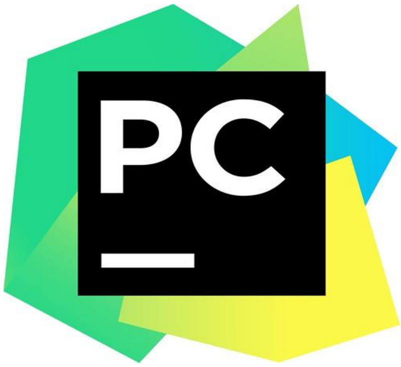
LINK TO DATASET

Iris Dataset on Kaggle: https://www.kaggle.com/datasets/uciml/iris

SOLUTION

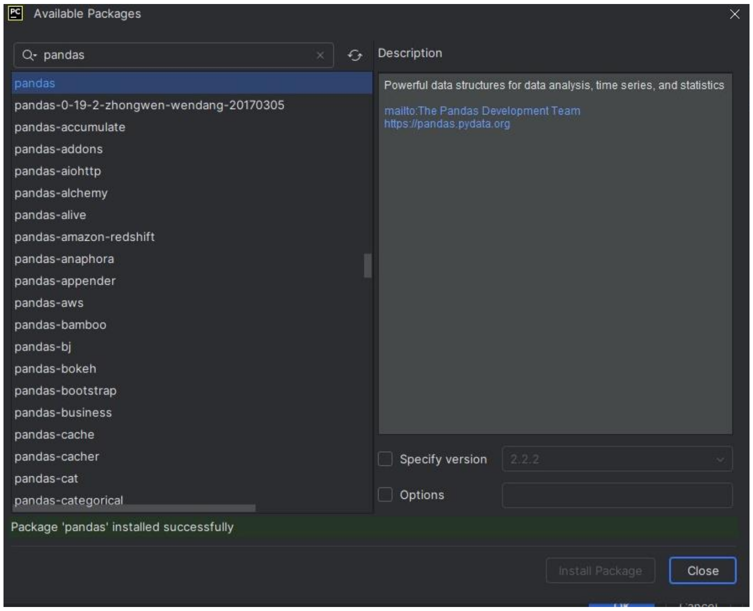
SOFTWARE USED

PyCharm



INSTALLING NECESSARY LIBRARIES

‘pandas’



pandas

pandas
pandas-0-19-2-zhongwen-wendang-20170305
pandas-accumulate
pandas-addons
pandas-aiohttp
pandas-alchemy
pandas-alive
pandas-amazon-redshift
pandas-anaphora
pandas-appender
pandas-aws
pandas-bamboo
pandas-bj
pandas-bokeh
pandas-bootstrap
pandas-business
pandas-cache
pandas-cacher
pandas-cat
pandas-categorical

Powerful data structures for data analysis, time series, and statistics

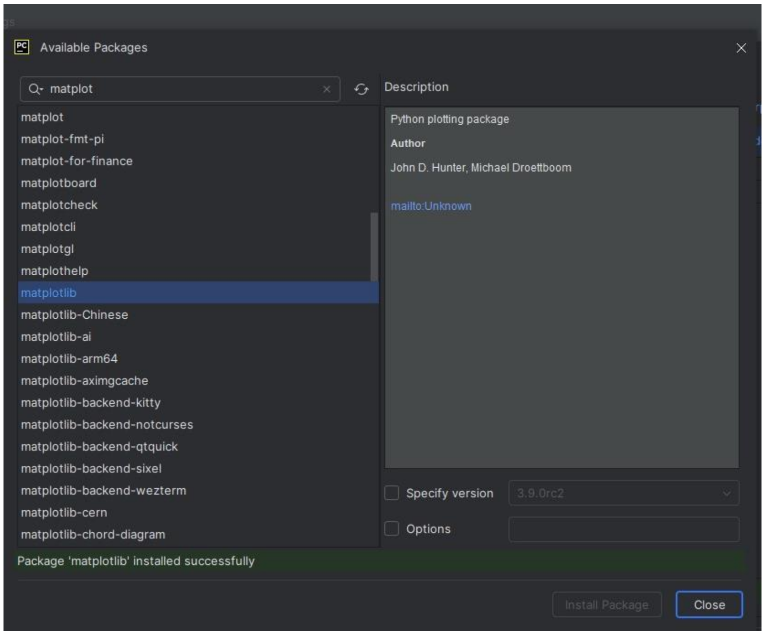
mailto:The Pandas Development Team
https://pandas.pydata.org

□ Specify version 2.2.2

□ Options

Package 'pandas' installed successfully

‘matplotlib’



Python plotting package

Author

John D. Hunter, Michael Droettboom

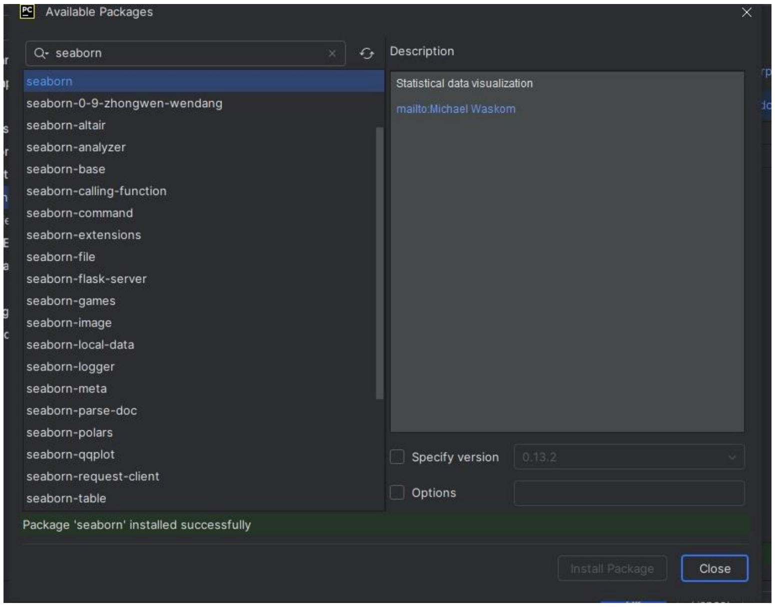
mailto:Unknown

□ Specify version 3.9.0rc2

□ Options

Package 'matplotlib' installed successfully

‘seaborn’



Statistical data visualization

mailto:Michael Waskom

□ Specify version 0.13.2

□ Options

Package 'seaborn' installed successfully

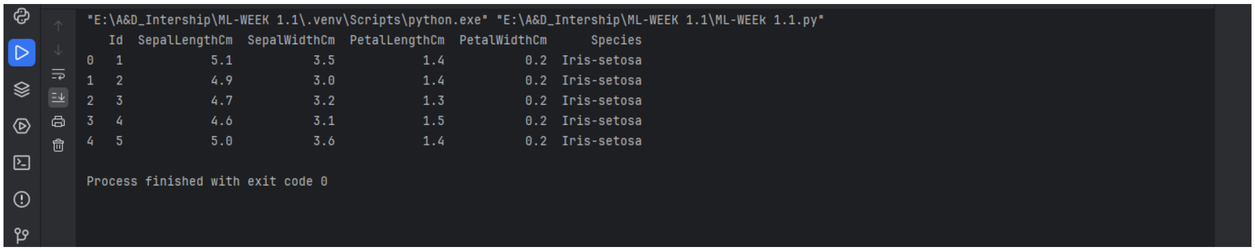
CODE

```
#'pandas' lib used for data manipulation & analysis
import pandas as pd

# Use a raw string by prefixing the path with r
df = pd.read_csv(r'E:\A&D_Intership\ML-WEEK 1.1\Iris.csv')

print(df.head())

#To get the basic info of the dataset
df.info()

#to display descriptive statistics
print(df.describe())

#To display the count of each sepal
print(df['Species'].value_counts())

#                                <-----Visualization------>
#'matplotlib' is used for creating visualizations
import matplotlib.pyplot as plt

# Create histograms
df.hist(figsize=(10, 8))
plt.show()

#Create Scatter plots
plt.figure(figsize=(10,8))
plt.scatter(df['SepalLengthCm'], df['SepalWidthCm'], c='r', label='Sepal_Length vs Sepal_width')
plt.scatter(df['PetalLengthCm'], df['PetalWidthCm'], c='b', label='Petal_Length vs Petal_width')
plt.xlabel('Length')
plt.ylabel('Width')
plt.legend
plt.show()

#Visualize the Dataset with Hue Based on the Species
#'seaborn' library is used for attractive & informative statistical graphs
import seaborn as sns

# Create a pairplot
sns.pairplot(df, hue='Species')
plt.show()
```

OUTPUT

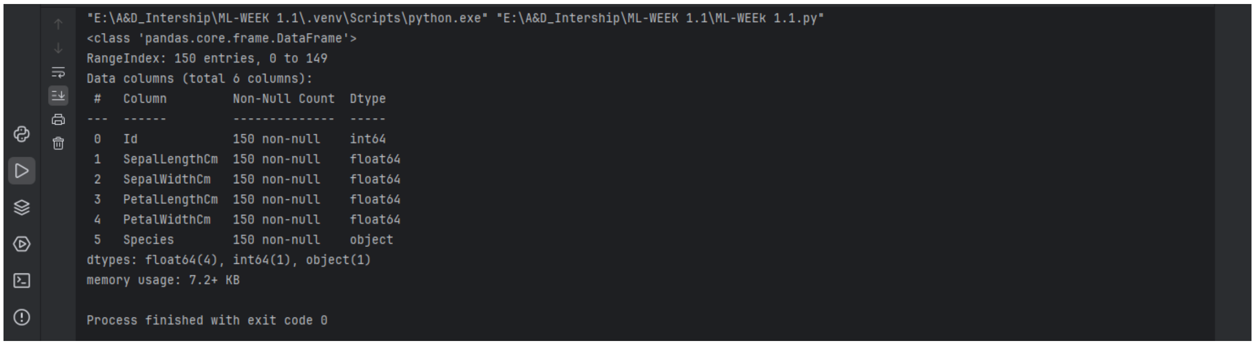
DATA LOADING & DATA EXPLORATION

Reads the Iris dataset from a CSV file into a “pandas DataFrame”.



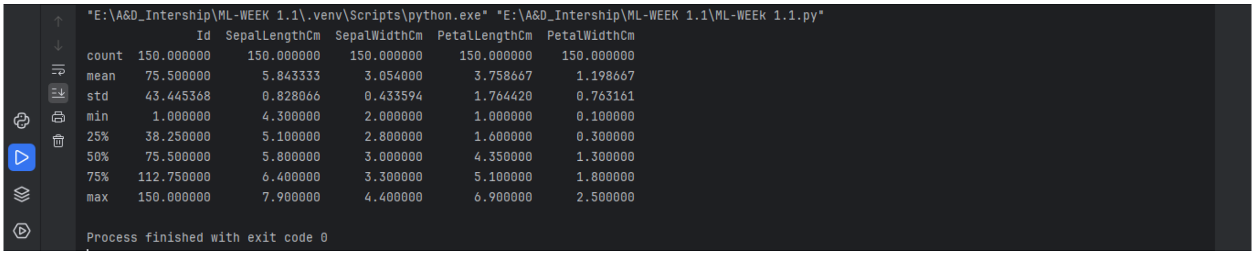
```
"E:\A&D_Intership\ML-WEEK 1.1\.venv\Scripts\python.exe" "E:\A&D_Intership\ML-WEEK 1.1\ML-WEEk 1.1.py"
   Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm      Species
0   1            5.1           3.5            1.4           0.2  Iris-setosa
1   2            4.9           3.0            1.4           0.2  Iris-setosa
2   3            4.7           3.2            1.3           0.2  Iris-setosa
3   4            4.6           3.1            1.5           0.2  Iris-setosa
4   5            5.0           3.6            1.4           0.2  Iris-setosa

Process finished with exit code 0
```

#To get the basic info of the dataset



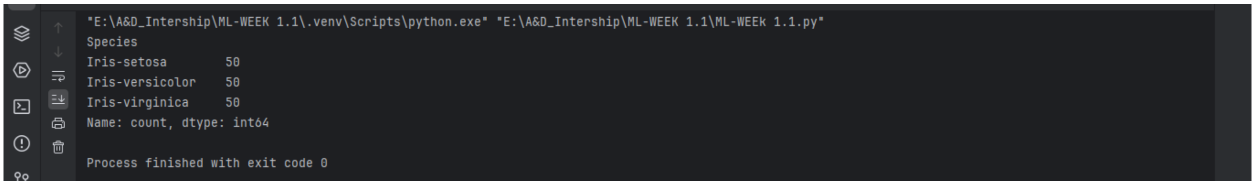
```
"E:\A&D_Intership\ML-WEEK 1.1\.venv\Scripts\python.exe" "E:\A&D_Intership\ML-WEEK 1.1\ML-WEEk 1.1.py"
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Id             150 non-null    int64
 1   SepalLengthCm  150 non-null    float64
 2   SepalWidthCm   150 non-null    float64
 3   PetalLengthCm  150 non-null    float64
 4   PetalWidthCm   150 non-null    float64
 5   Species        150 non-null    object
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB

Process finished with exit code 0
```

#To display descriptive statistics



```
"E:\A&D_Intership\ML-WEEK 1.1\.venv\Scripts\python.exe" "E:\A&D_Intership\ML-WEEK 1.1\ML-WEEk 1.1.py"
               Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm
count  150.000000     150.000000    150.000000     150.000000    150.000000
mean    75.500000       5.843333      3.054000       3.758667      1.198667
std     43.445368       0.828066      0.433594       1.764420      0.763161
min      1.000000       4.300000      2.000000       1.000000      0.100000
25%     38.250000       5.100000      2.800000       1.600000      0.300000
50%     75.500000       5.800000      3.000000       4.350000      1.300000
75%    112.750000       6.400000      3.300000       5.100000      1.800000
max    150.000000       7.900000      4.400000       6.900000      2.500000

Process finished with exit code 0
```

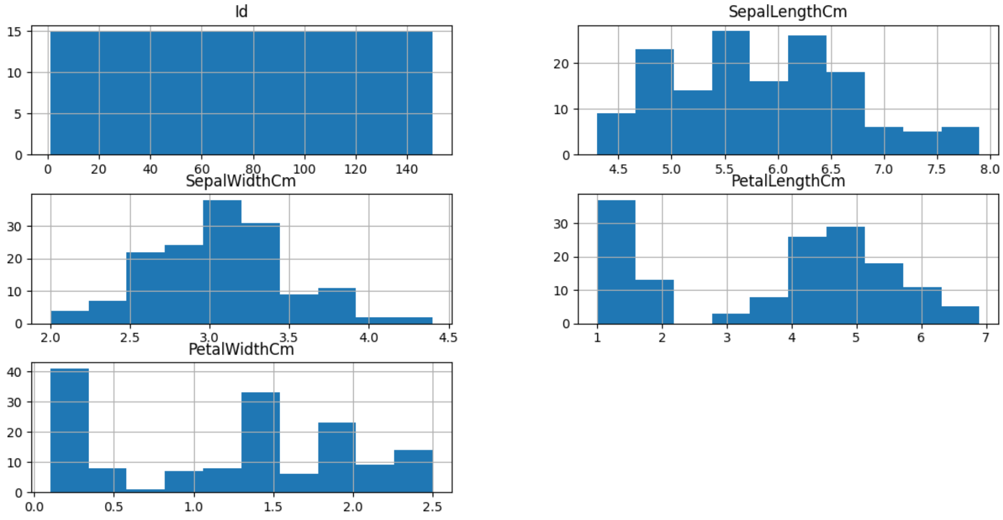
#To display the count of each sepal



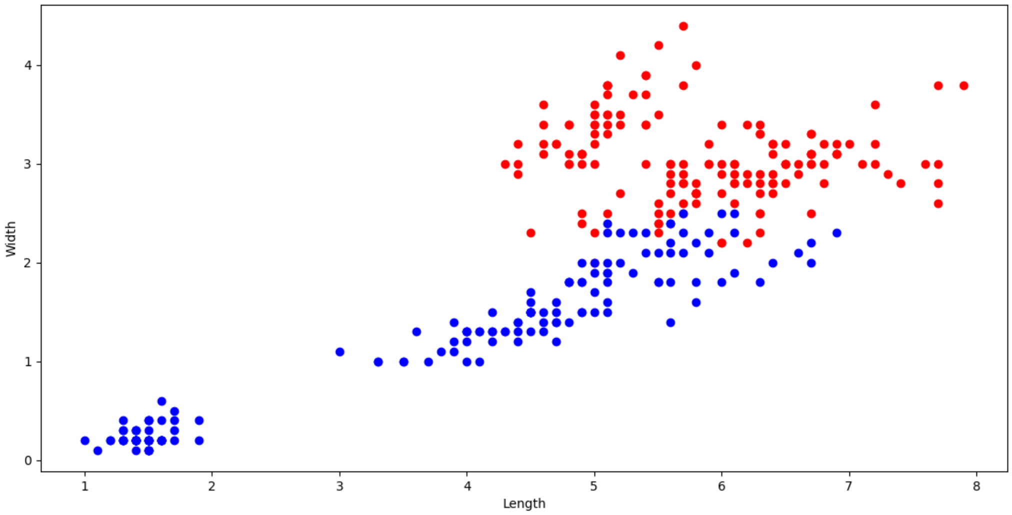
```
"E:\A&D_Intership\ML-WEEK 1.1\.venv\Scripts\python.exe" "E:\A&D_Intership\ML-WEEK 1.1\ML-WEEk 1.1.py"
Species
Iris-setosa        50
Iris-versicolor    50
Iris-virginica     50
Name: count, dtype: int64

Process finished with exit code 0
```

VISUALIZATION

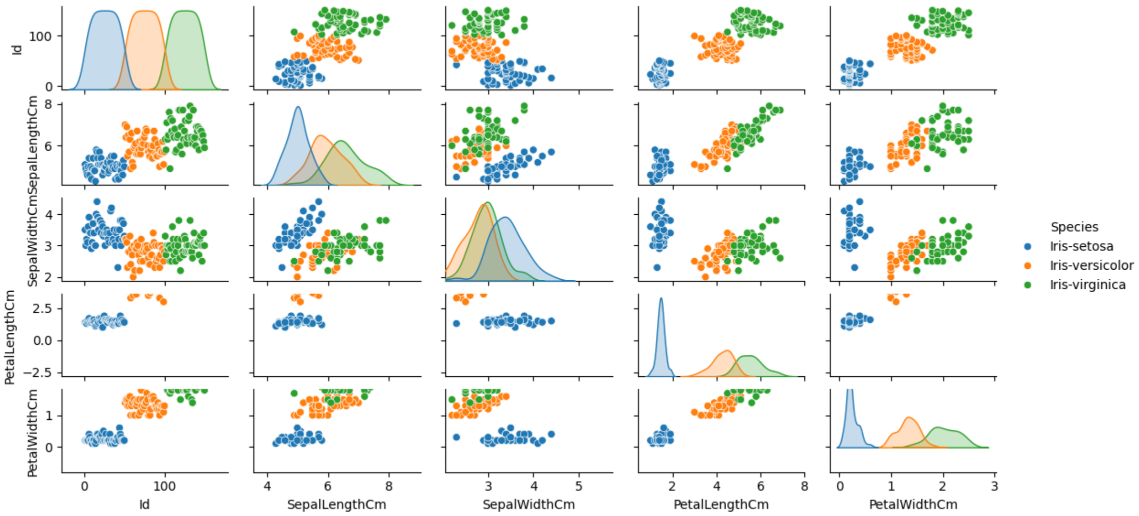
Histograms



Scatter Plot



seaborn’s pairplot to visualize the dataset with hue based on the species



EXPLANATION

`pandas` library, which is used for data manipulation and analysis in Python.

# Use a raw string by prefixing the path with r

- **df = pd.read_csv(r'E:\A&D_Intership\ML-WEEK 1.1\Iris.csv')**

This line reads the CSV file located at the specified path into a pandas DataFrame called `df`. The `r` before the string indicates a raw string, which treats backslashes (`\`) as literal characters, making it easier to specify file paths.

- **print(df.head())**

This line prints the first five rows of the DataFrame, allowing you to see a quick preview of the data.

# To get the basic info of the dataset

- **df.info()**

This line prints a summary of the DataFrame, including the number of entries, column names, data types, and non-null counts for each column. It helps you understand the structure and content of the dataset.

# To display descriptive statistics

- **print(df.describe())**

This line prints descriptive statistics for each numerical column in the DataFrame, including measures such as mean, standard deviation, minimum, and maximum values. It provides a quick overview of the data distribution.

# To display the count of each species

- **print(df['Species'].value_counts())**

This line prints the count of each unique value in the 'Species' column, allowing you to see the distribution of different species in the dataset.

# <----- Visualization ------>

- **import matplotlib.pyplot as plt**

This line imports the `matplotlib.pyplot` library, which is used for creating visualizations in Python.

- **# Create histograms**

  df.hist(figsize=(10, 8))

  plt.show()

These lines create histograms for all numerical columns in the DataFrame, providing a visual representation of the distribution of each feature. The `figsize` parameter sets the size of the figure, and `plt.show()` displays the histograms.

# Create scatter plots

- **plt.figure(figsize=(10, 8))**

  plt.scatter(df['SepalLengthCm'], df['SepalWidthCm'], c='r', label='Sepal Length vs Sepal Width')

  plt.scatter(df['PetalLengthCm'], df['PetalWidthCm'], c='b', label='Petal Length vs Petal Width')

  plt.xlabel('Length')

  plt.ylabel('Width')

```
plt.legend()
plt.show()
```

These lines create a scatter plot to visualize the relationships between different pairs of features. Two scatter plots are created: one for sepal length vs. sepal width (in red) and one for petal length vs. petal width (in blue). The `figsize` parameter sets the size of the figure, `plt.xlabel()` and `plt.ylabel()` set the labels for the x and y axes, `plt.legend()` adds a legend, and `plt.show()` displays the scatter plot.

```
# Visualize the dataset with hue based on the species
```

- **import seaborn as sns**

This line imports the `seaborn` library, which is used for creating attractive and informative statistical graphics.

```
# Create a pairplot
```

- **sns.pairplot(df, hue='Species')**

  plt.show()

These lines create a pairplot using Seaborn, which plots pairwise relationships in the dataset with different colors (hues) based on the species. This helps visualize the relationships between features and how they differ across species. `plt.show()` displays the pairplot.

SUMMARY

The code performs the following tasks:

- **Data Loading:** Reads the Iris dataset from a CSV file into a ‘pandas DataFrame’.
- **Data Exploration:** Prints the first few rows, basic information, descriptive statistics, and the count of each species.
- **Visualization:** Creates histograms for numerical features, scatter plots for feature relationships, and a pairplot to visualize the data with hues based on species.