

Dataset Exploration

```
In [2]: #Load the Iris dataset from sklearn.dataset.
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

```
In [3]: load_iris=pd.read_csv(r"D:\Iris.csv")
load_iris
```

```
Out[3]:
```

	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

	145	146	6.7	3.0	5.2	2.3	Iris-virginica
	146	147	6.3	2.5	5.0	1.9	Iris-virginica
	147	148	6.5	3.0	5.2	2.0	Iris-virginica
	148	149	6.2	3.4	5.4	2.3	Iris-virginica
	149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

```
In [4]: #Display the first five rows
load_iris.head()
```

```
Out[4]:
```

	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

```
In [5]: #Display the shape
load_iris.shape
```

```
Out[5]: (150, 6)
```

```
In [6]: #Statistics(mean)
load_iris.mean()
```

C:\Users\ANJALI\AppData\Local\Temp\ipykernel_7920\1933876758.py:1: FutureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError. Select only valid columns before calling the reduction.

```
load_iris.mean()
Out[6]: Id                75.500000
SepalLengthCm          5.843333
SepalWidthCm           3.054000
PetalLengthCm          3.758667
PetalWidthCm           1.198667
dtype: float64
```

```
In [7]: #standerd deviation(min)
load_iris.min()
```

```
Out[7]: Id                1
SepalLengthCm          4.3
SepalWidthCm           2.0
PetalLengthCm           1.0
PetalWidthCm           0.1
Species              Iris-setosa
dtype: object
```

```
In [8]: #standerd deviation(max)
load_iris.max()
```

```
Out[8]: Id 150
SepalLengthCm 7.9
SepalWidthCm 4.4
PetalLengthCm 6.9
PetalWidthCm 2.5
Species Iris-virginica
dtype: object
```

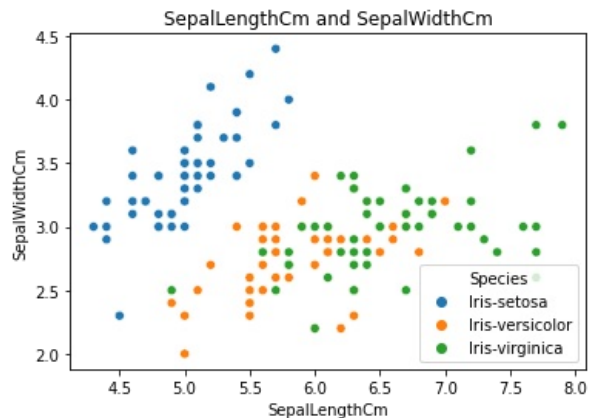
Data Splitting

```
In [9]: #Split the iris dataset training and testing sets using an 80-20 split
plt.figure(figsize=(80,20))
```

```
Out[9]: <Figure size 5760x1440 with 0 Axes>
```

```
<Figure size 5760x1440 with 0 Axes>
```

```
In [11]: sns.scatterplot(data=load_iris, x="SepalLengthCm", y="SepalWidthCm", hue="Species")
plt.title("SepalLengthCm and SepalWidthCm")
plt.show()
```



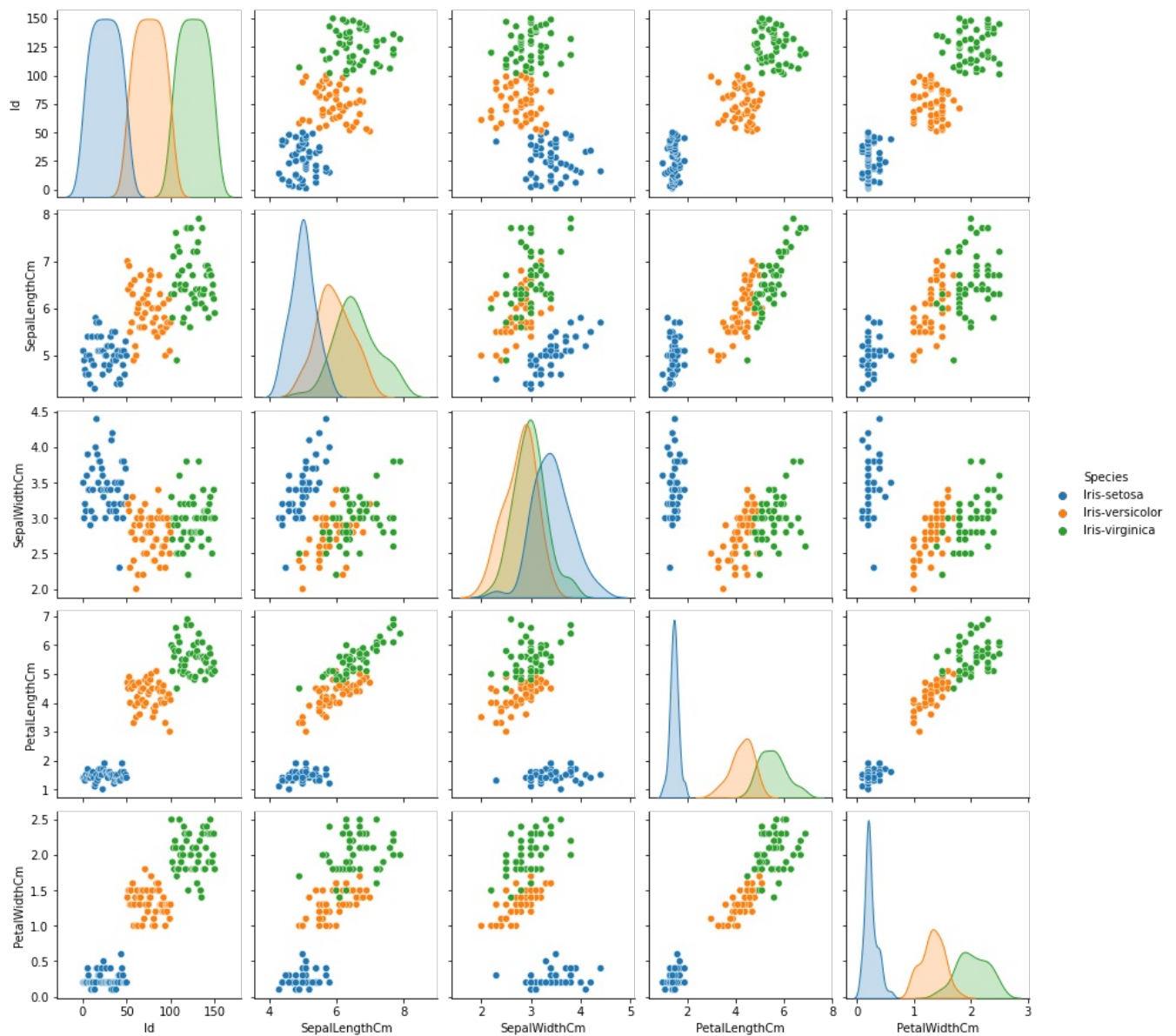
```
In [6]: plt.figure(figsize=(80,20))
sns.scatterplot(data=load_iris, x="PetalLengthCm", y="PetalWidthCm", hue="Species")
plt.title("PetalLengthCm and PetalWidthCm", fontsize=30)
plt.xlabel("PetalLengthCm", fontsize=30)
plt.ylabel("PetalWidthCm", fontsize=30)
plt.show()
```

```
-----
NameError                                Traceback (most recent call last)
Input In [6], in <cell line: 1>()
----> 1 plt.figure(figsize=(80,20))
      2 sns.scatterplot(data=load_iris, x="PetalLengthCm", y="PetalWidthCm", hue="Species")
      3 plt.title("PetalLengthCm and PetalWidthCm", fontsize=30)

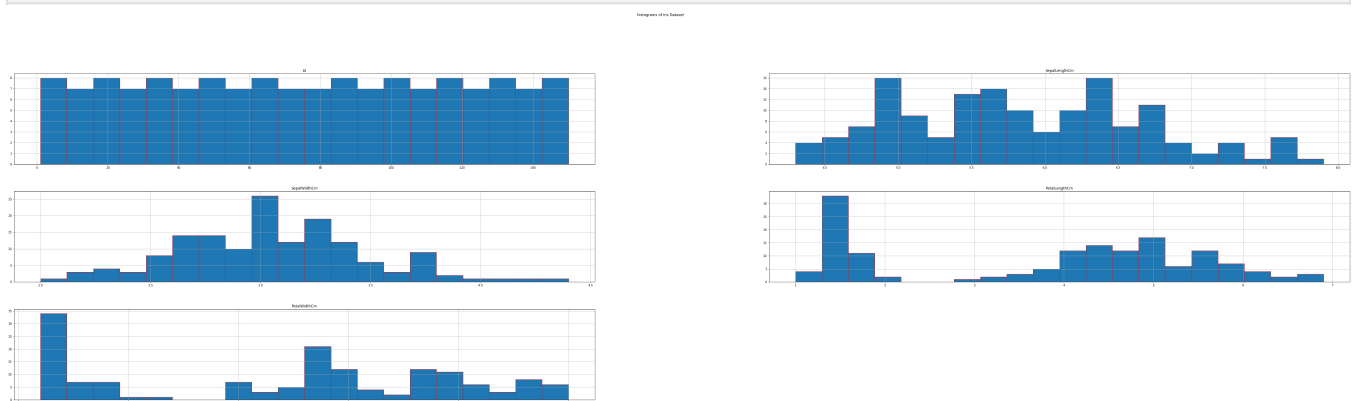
NameError: name 'plt' is not defined
```

```
In [15]: sns.pairplot(load_iris, hue="Species")
plt.suptitle("pairplot of Iris Dataset", y=1.02)
plt.show()
```

pairplot of Iris Dataset



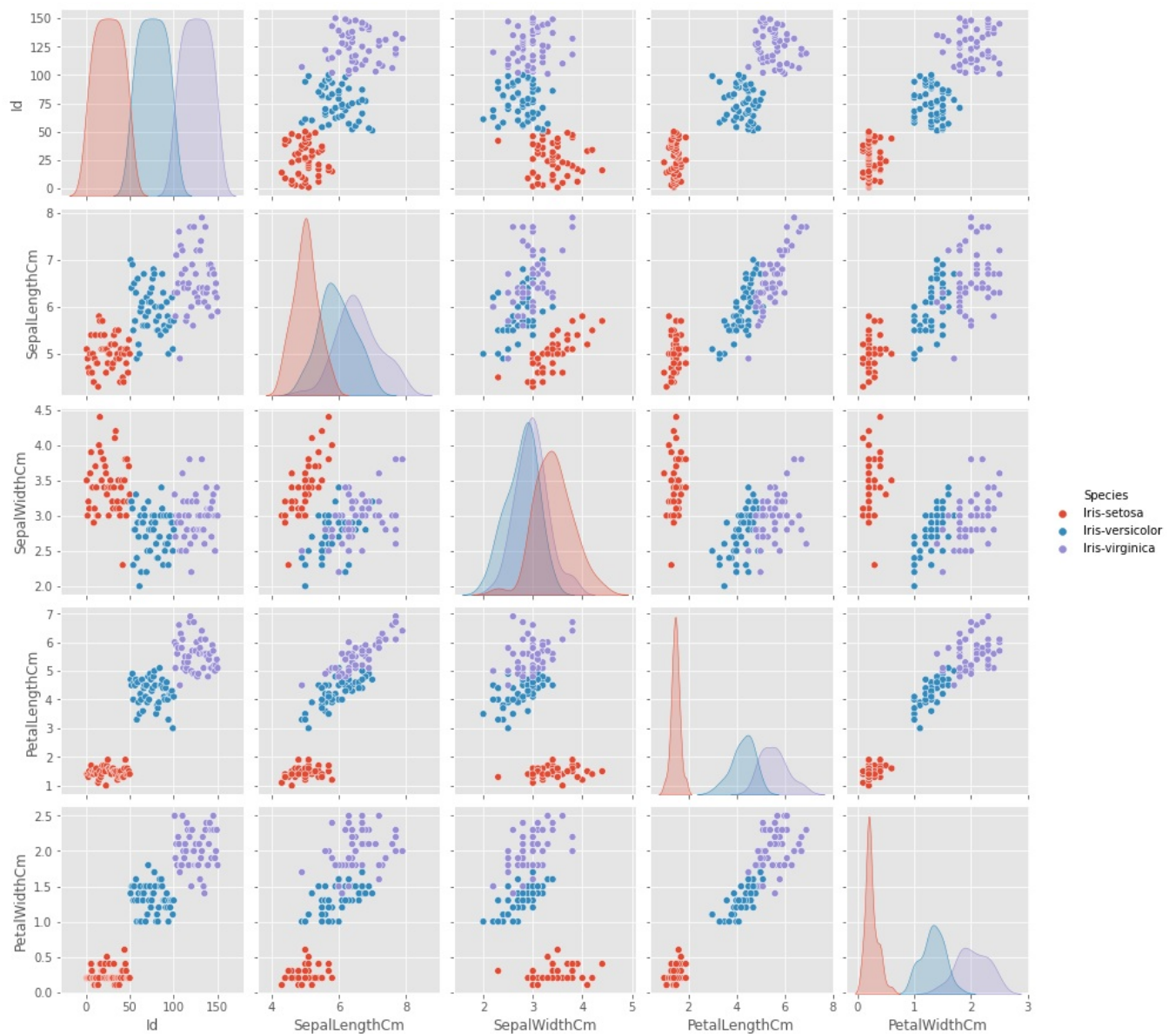
```
In [17]: load_iris.hist(figsize=(80,20), bins=20, edgecolor="red")
plt.suptitle("histograms of Iris Dataset", y=1.02)
plt.show()
```



```
In [18]: load_iris["Species"].value_counts()
```

```
Out[18]: Iris-setosa      50
Iris-versicolor    50
Iris-virginica     50
Name: Species, dtype: int64
```

```
In [20]: plt.style.use('ggplot')
sns.pairplot(load_iris,hue="Species")
plt.show()
```



In []:

In [33]:

```
X= load_iris.drop(labels="SepalLengthCm", axis= 1)
y= load_iris["SepalLengthCm"]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size= 0.33, random_state= 101)
X_train.shape
```

Out[33]: (100, 4)

In [34]: X_test.shape

Out[34]: (50, 4)

LinearRegression() Model

In [35]: lr = LinearRegression()

In [36]: lr.fit(X_train, y_train)

Out[36]: LinearRegression()

In [37]: lr.predict(X_test)

Out[37]: array([5.48508371, 5.09089138, 4.89512851, 7.04192788, 6.53871592,
6.02247942, 5.63209747, 5.46190717, 5.87710525, 4.71240875,
6.29813106, 5.56233328, 4.89557897, 7.34079215, 6.21946738,
6.0740031 , 5.97388322, 5.97615321, 4.73028901, 6.7484953 ,
5.4821475 , 5.21292714, 6.00114952, 6.22692563, 6.05204256,
5.54564204, 5.08496483, 5.85337869, 4.84318173, 4.06243592,
6.66748 , 5.56140562, 6.6910352 , 5.69150289, 6.49569541,
6.178897 , 6.37102084, 5.94373446, 5.84419136, 6.80202715,
5.10593422, 4.78659556, 4.97323396, 6.41043817, 6.16949291,
4.53680494, 6.81262011, 5.99228598, 4.89106638, 4.91714345])

```
In [38]: pred = lr.predict(X_test)
```

```
In [43]: load_iris.loc[6]
```

```
Out[43]: Id                7.0  
SepalLengthCm          4.6  
SepalWidthCm           3.4  
PetalLengthCm           1.4  
PetalWidthCm            0.3  
Name: 6, dtype: float64
```

```
In [44]: d = {'sepal length (cm)' : [4.6],  
             'sepal width (cm)'  : [3.4],  
             'petal length (cm)' : [1.4],  
             'petal width (cm)'  : [0.3],  
             'species'          : 0}
```

```
In [45]: test_df = pd.DataFrame(data= d)
```

```
In [46]: test_df
```

```
Out[46]:
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	species
0	4.6	3.4	1.4	0.3	0

```
In [47]: X_test = test_df.drop('sepal length (cm)', axis= 1)  
y_test = test_df['sepal length (cm)']  
lr.predict(X_test)
```

C:\ProgramData\Anaconda3\lib\site-packages\sklearn\base.py:493: FutureWarning: The feature names should match those that were passed during fit. Starting version 1.2, an error will be raised.

Feature names unseen at fit time:

- petal length (cm)
- petal width (cm)
- sepal width (cm)
- species

Feature names seen at fit time, yet now missing:

- Id
- PetalLengthCm
- PetalWidthCm
- SepalWidthCm

warnings.warn(message, FutureWarning)

```
Out[47]: array([2.92758729])
```

```
In [48]: pred = lr.predict(X_test)
```

C:\ProgramData\Anaconda3\lib\site-packages\sklearn\base.py:493: FutureWarning: The feature names should match those that were passed during fit. Starting version 1.2, an error will be raised.

Feature names unseen at fit time:

- petal length (cm)
- petal width (cm)
- sepal width (cm)
- species

Feature names seen at fit time, yet now missing:

- Id
- PetalLengthCm
- PetalWidthCm
- SepalWidthCm

warnings.warn(message, FutureWarning)

```
In [49]: print('Predicted Sepal Length (cm):', pred[0])  
print('Actual Sepal Length (cm):', 4.6)
```

Predicted Sepal Length (cm): 2.927587288410093
Actual Sepal Length (cm): 4.6

```
In [ ]:
```