Homework 1

- This is an introductory homework to familiarize you with handling datasets and visualizing them by using different libraries.
- Please go through the Pandas tutorial in Canvas module 1, prior to attempting this homework.
- There is a Notebook on Canvas called ISLRChapter2Prob8.ipynb which has solutions to Problem 8 from Chapter 2 in the book. Use that as a guide to learn about some Pandas functions.
- Contact the TA regarding any doubts (details available in the *office hours* tab in Canvas)
- All the data required for all the homeworks will be available in Canvas - Files/Homeworks/Data
- In this homework we will explore the **Iris dataset** .

Importing relevant libraries

- We start off by importing required libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import metrics
sns.set()
```

Loading data

- The data is in a csv file : 'Iris.csv'
- Import this file using "pd.read_csv( )" command
- When importing the file make sure the notebook is in the same location as the file or specify the path of the file like data\ iris.csv

```
iris_df = pd.read_csv('Iris.csv')
print(iris_df)
```

```
      Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm  \
0      1            5.1           3.5            1.4           0.2
1      2            4.9           3.0            1.4           0.2
```

✓ 1s completed at 11:57 PM

```
149  150            5.9           3.0           5.1           1.8

            Species
0       Iris-setosa
1       Iris-setosa
2       Iris-setosa
3       Iris-setosa
4       Iris-setosa
..              ...
145  Iris-virginica
146  Iris-virginica
147  Iris-virginica
148  Iris-virginica
149  Iris-virginica

[150 rows x 6 columns]
```

Gaining information from data

It is always a good idea to first understand what is your data set. What are the different features etc.

- Use "df.info( )" command to get basic information about the datafarme
- Use "df.describe( )" command to get statistical information about the dataframe
- df here refers to dataframe

```
print(iris_df.info())
print(iris_df.describe())
```

```
<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
None
```

- Use "df.duplicated( )" command to get the indices of duplicate rows
- Use "df['column_name'].value_counts( )" command to get the counts of different species in the dataset.

```
#ommiting the ID column
ID_ommited = iris_df.loc[:,['SepalLengthCm','SepalWidthCm','PetalLengthCm','PetalWi
print(ID_ommited[ID_ommited.duplicated(keep=False)])

iris_df['Species'].value_counts()
```

```
     SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm         Species
9              4.9           3.1            1.5           0.1     Iris-setosa
34             4.9           3.1            1.5           0.1     Iris-setosa
37             4.9           3.1            1.5           0.1     Iris-setosa
101            5.8           2.7            5.1           1.9  Iris-virginica
142            5.8           2.7            5.1           1.9  Iris-virginica
Iris-versicolor    50
Iris-setosa        50
Iris-virginica     50
Name: Species, dtype: int64
```

Data visualization

In this section you will be required to generate certain plots/graphs from the data and provide your comments/insights from the plots

a) Species count

- For this you are required to produce a histogram about the counts of different ***species*** in the datasets
- Use "sns.countplot( )"

```
plt.figure(figsize=( 8 , 4 ))
```

Species

Type your insights here

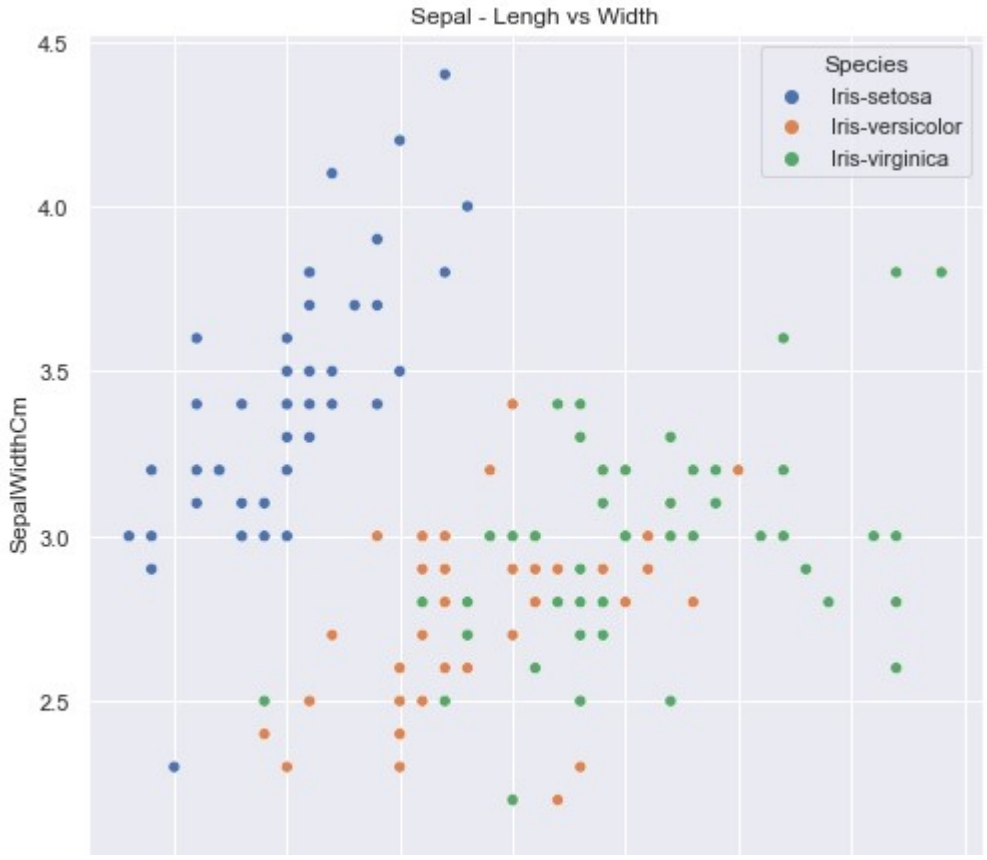
- __

b) Uni-variate analysis : Comparison between various species based on sepal length and width

- For this you are required to produce a ***scatter plot between sepal length and sepal width for different species*** in the dataset
- Use "sns.scatterplot( )"
- Set the hue parameter to be the species column of the dataframe

```
plt.figure(figsize=( 8 , 8 ))
sns.scatterplot(
    x = iris_df['SepalLengthCm'],
    y = iris_df['SepalWidthCm'],
    hue = iris_df['Species']
)
plt.title('Sepal - Lengh vs Width')
```

```
Text(0.5, 1.0, 'Sepal - Lengh vs Width')
```



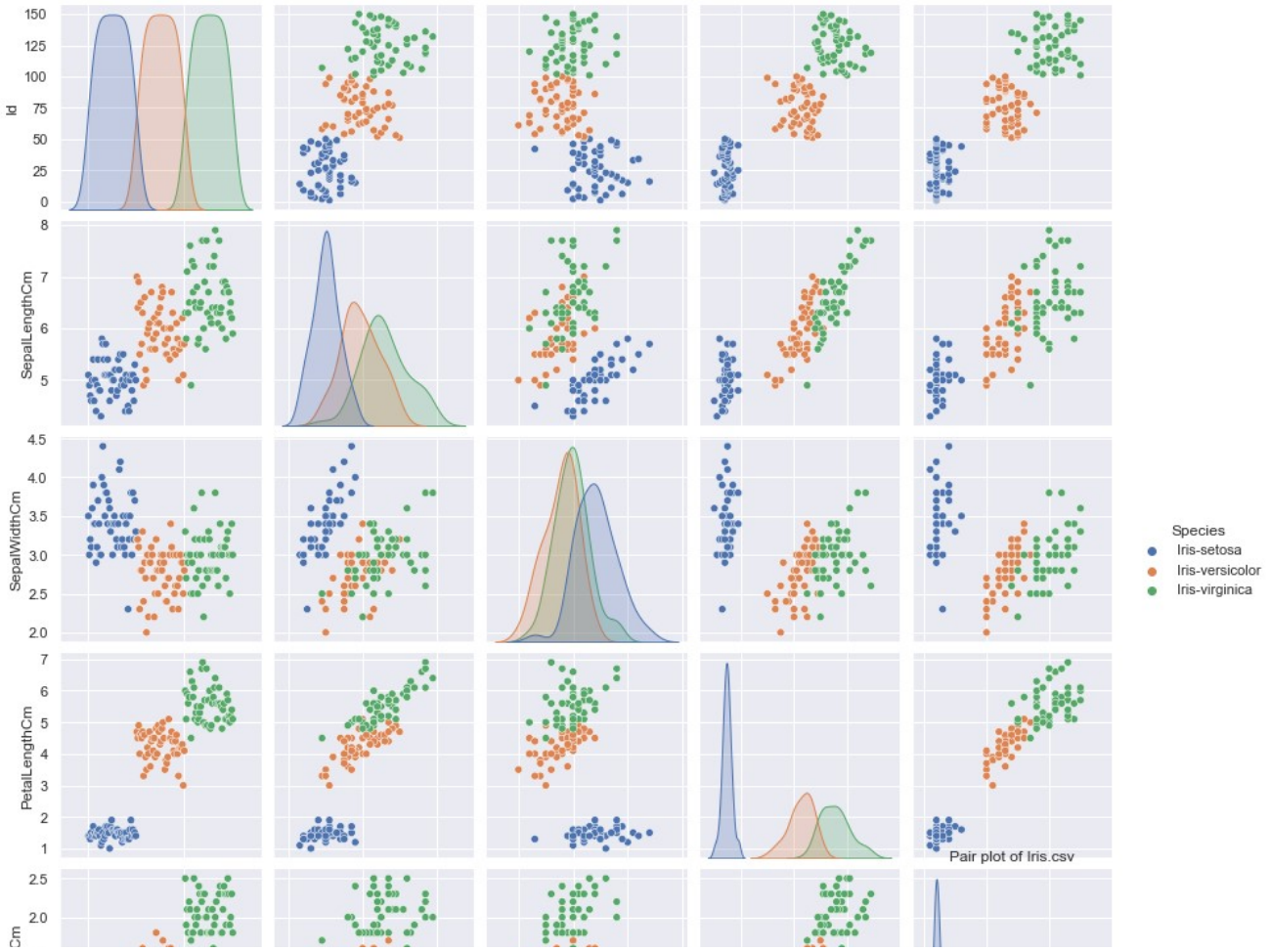
- I saw that the iris-setosa had was much shorter on average than iris-versicolor and iris-virginca, but had a larger width. They all seemed to have larger sepa widths when the length of the sepal increased. Iris-versicolor and Iris-virginica clustered together pretty much, showing that they have similar sepal sizes, virginica had slightly larger, but overall they were close.

c) Bi-variate analysis : Plot pairwise relationships

- For this you are required to produce a ***pair plot between all the features for different species*** in the dataset
- Use "sns.pairplot( )"
- Set the hue parameter to be the species column of the dataframe

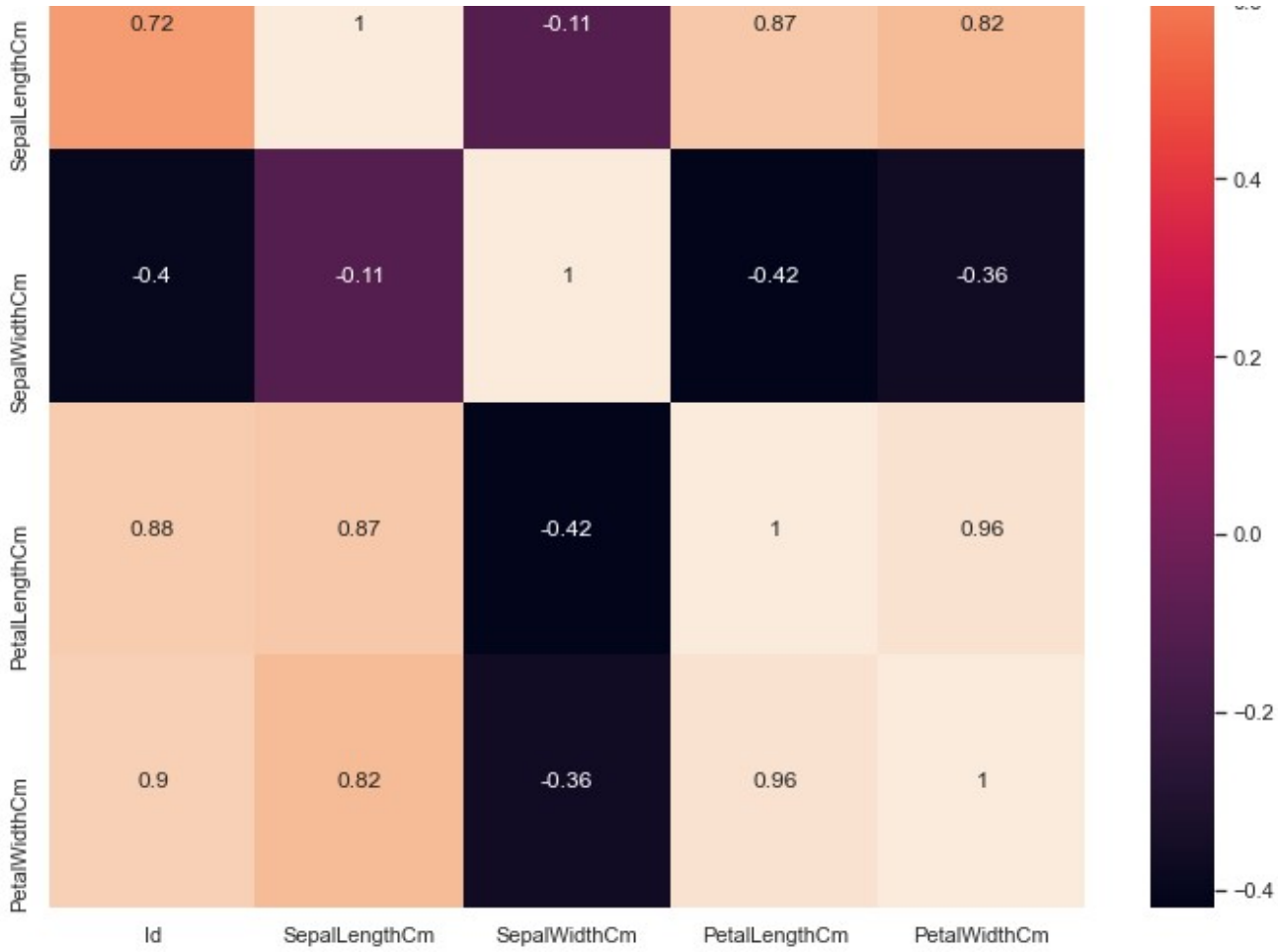
```
plt.figure(figsize=( 14 , 14 ))
sns.pairplot(iris_df, hue = "Species")
plt.title('Pair plot of Iris.csv')
```

```
Text(0.5, 1.0, 'Pair plot of Iris.csv')
<Figure size 1008x1008 with 0 Axes>
```



Type your insights here

- looking at the data I saw common groupings that presented of the 3 species of flowers.


0.72
1
-0.11
0.87
0.82
-0.4
-0.11
1
-0.42
-0.36
0.88
0.87
-0.42
1
0.96
0.9
0.82
-0.36
0.96
1
SepalLengthCm
SepalWidthCm
PetalLengthCm
PetalWidthCm
Id
SepalLengthCm
SepalWidthCm
PetalLengthCm
PetalWidthCm
0.4
0.2
0.0
−0.2
−0.4

```
!export PATH=/Library/TeX/texbin:$PATH
```

```
'export' is not recognized as an internal or external command,
operable program or batch file.
```