LetsGroMore VIP - Data Science

Iris Flower Classification

In [1]:

```
import pandas as pd
import numpy as np
import os
import seaborn as sns
import matplotlib.pyplot
from sklearn.metrics import confusion_matrix
```

In [2]:

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

In [3]:

```
df.head()
```

Out[3]:

	5.1	3.5	1.4	0.2	Iris-setosa
0	4.9	3.0	1.4	0.2	Iris-setosa
1	4.7	3.2	1.3	0.2	Iris-setosa
2	4.6	3.1	1.5	0.2	Iris-setosa
3	5.0	3.6	1.4	0.2	Iris-setosa
4	5.4	3.9	1.7	0.4	Iris-setosa

In [4]:

```
df.tail()
```

Out[4]:

	5.1	3.5	1.4	0.2	Iris-setosa
144	6.7	3.0	5.2	2.3	Iris-virginica
145	6.3	2.5	5.0	1.9	Iris-virginica
146	6.5	3.0	5.2	2.0	Iris-virginica
147	6.2	3.4	5.4	2.3	Iris-virginica
148	5.9	3.0	5.1	1.8	Iris-virginica

In [5]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 149 entries, 0 to 148
Data columns (total 5 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   5.1          149 non-null    float64
 1   3.5          149 non-null    float64
 2   1.4          149 non-null    float64
 3   0.2          149 non-null    float64
 4   Iris-setosa  149 non-null    object 
dtypes: float64(4), object(1)
memory usage: 5.9+ KB
```

In [26]:

```
df.shape
```

Out[26]:

```
(149, 5)
```

In [8]:

```
df.dtypes
```

Out[8]:

```
5.1            float64
3.5            float64
1.4            float64
0.2            float64
Iris-setosa     object
dtype: object
```

In [10]:

```
df.isnull().sum()
```

Out[10]:

```
5.1            0
3.5            0
1.4            0
0.2            0
Iris-setosa    0
dtype: int64
```

In [11]:

```
df.describe()
```

Out[11]:

	5.1	3.5	1.4	0.2
count	149.000000	149.000000	149.000000	149.000000
mean	5.848322	3.051007	3.774497	1.205369
std	0.828594	0.433499	1.759651	0.761292
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.400000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

In [13]:

```
df['1.4'].value_counts()
```

Out[13]:

```
1.5    14
1.4    11
5.1     8
4.5     8
1.3     7
1.6     7
5.6     6
4.0     5
4.9     5
4.7     5
4.8     4
1.7     4
4.4     4
4.2     4
5.0     4
4.1     3
5.5     3
4.6     3
6.1     3
5.7     3
3.9     3
5.8     3
1.2     2
1.9     2
6.7     2
3.5     2
5.9     2
6.0     2
5.4     2
5.3     2
3.3     2
4.3     2
5.2     2
6.3     1
1.1     1
6.4     1
3.6     1
3.7     1
3.0     1
3.8     1
6.6     1
6.9     1
1.0     1
Name: 1.4, dtype: int64
```

In [14]:

```
df.corr()
```

Out[14]:

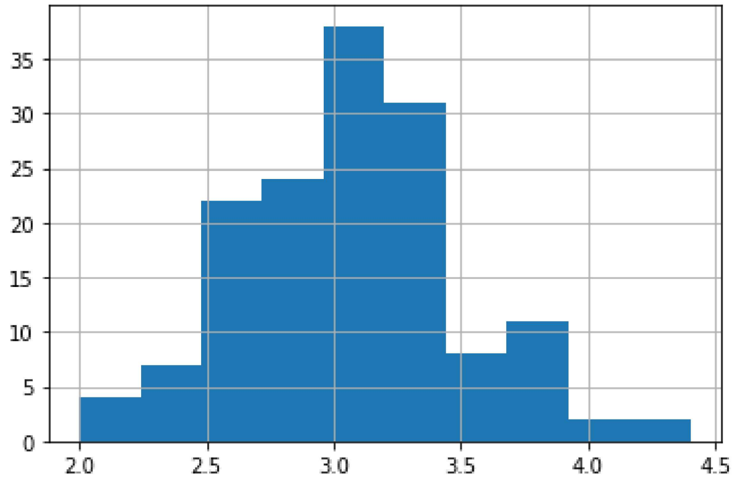
	5.1	3.5	1.4	0.2
5.1	1.000000	-0.103784	0.871283	0.816971
3.5	-0.103784	1.000000	-0.415218	-0.350733
1.4	0.871283	-0.415218	1.000000	0.962314
0.2	0.816971	-0.350733	0.962314	1.000000

In [15]:

```
df['3.5'].hist()
```

Out[15]:

```
<AxesSubplot:>
```

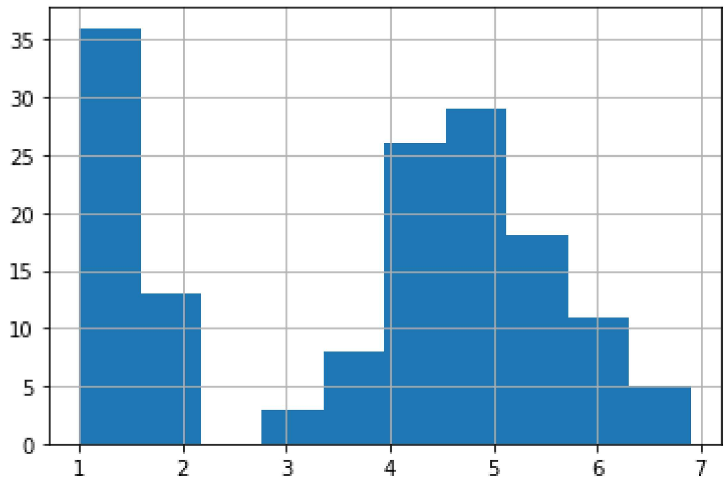


In [16]:

```
df['1.4'].hist()
```

Out[16]:

```
<AxesSubplot:>
```

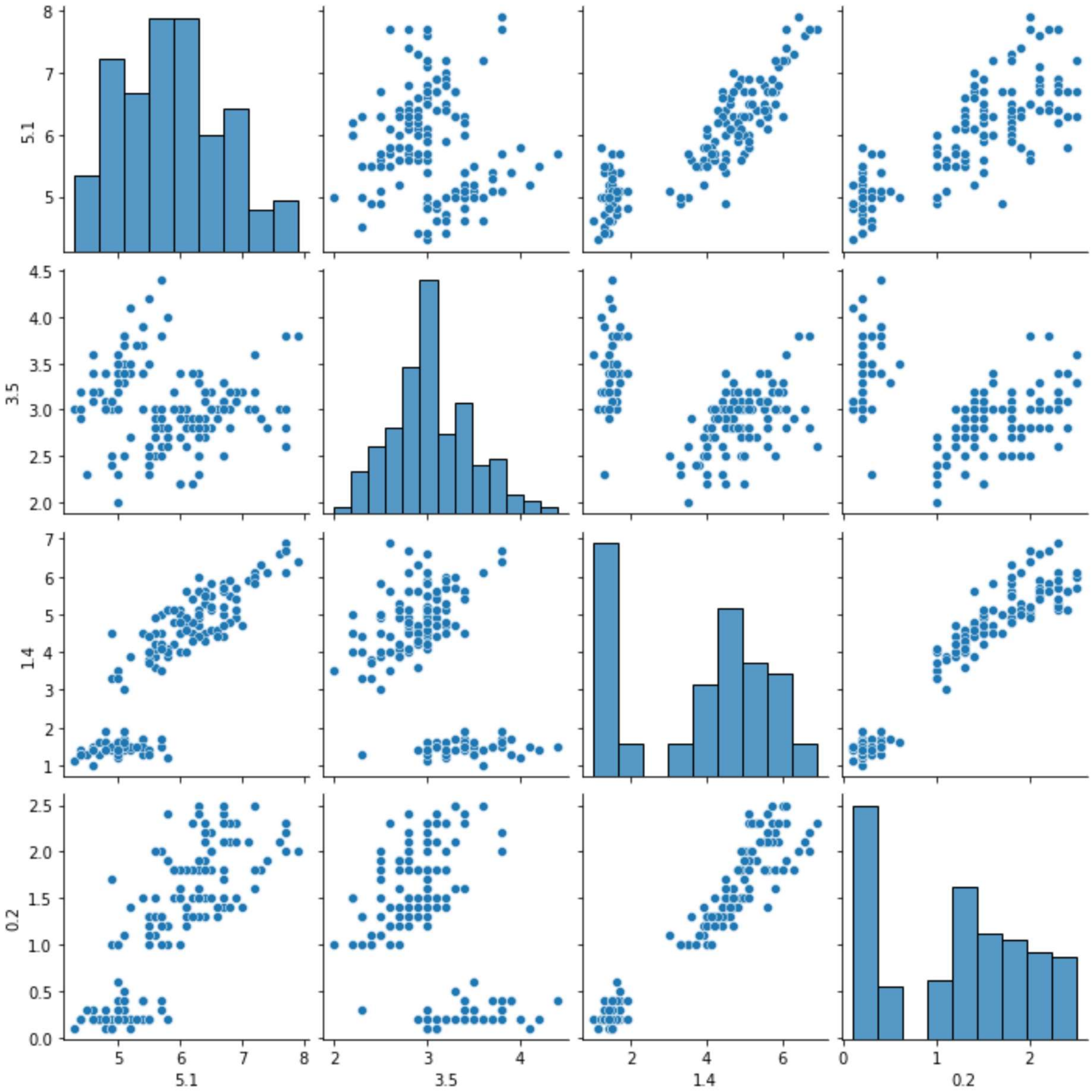


In [18]:

```
sns.pairplot(df)
```

Out[18]:

```
<seaborn.axisgrid.PairGrid at 0x2ab78339a90>
```

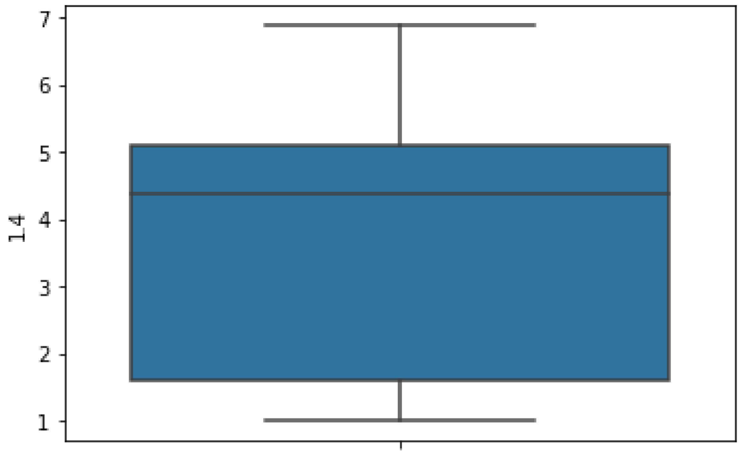


In [24]:

```
sns.boxplot(y = df['1.4'])
```

Out[24]:

```
<AxesSubplot:ylabel='1.4'>
```



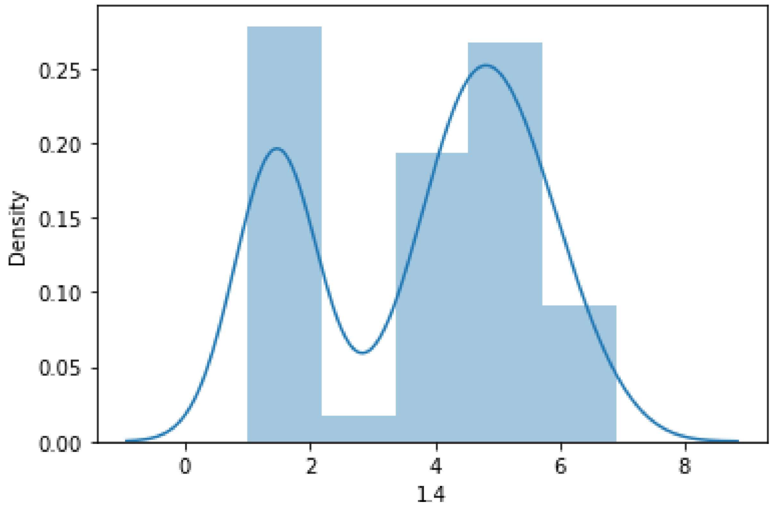
In [25]:

```
sns.distplot(df['1.4'])
```

```
D:\python_anaconda\lib\site-packages\seaborn\distributions.py:2551: FutureWa
rning: `distplot` is a deprecated function and will be removed in a future v
ersion. Please adapt your code to use either `displot` (a figure-level funct
ion with similar flexibility) or `histplot` (an axes-level function for hist
ograms).
  warnings.warn(msg, FutureWarning)
```

Out[25]:

```
<AxesSubplot:xlabel='1.4', ylabel='Density'>
```



In [ ]:

```

```