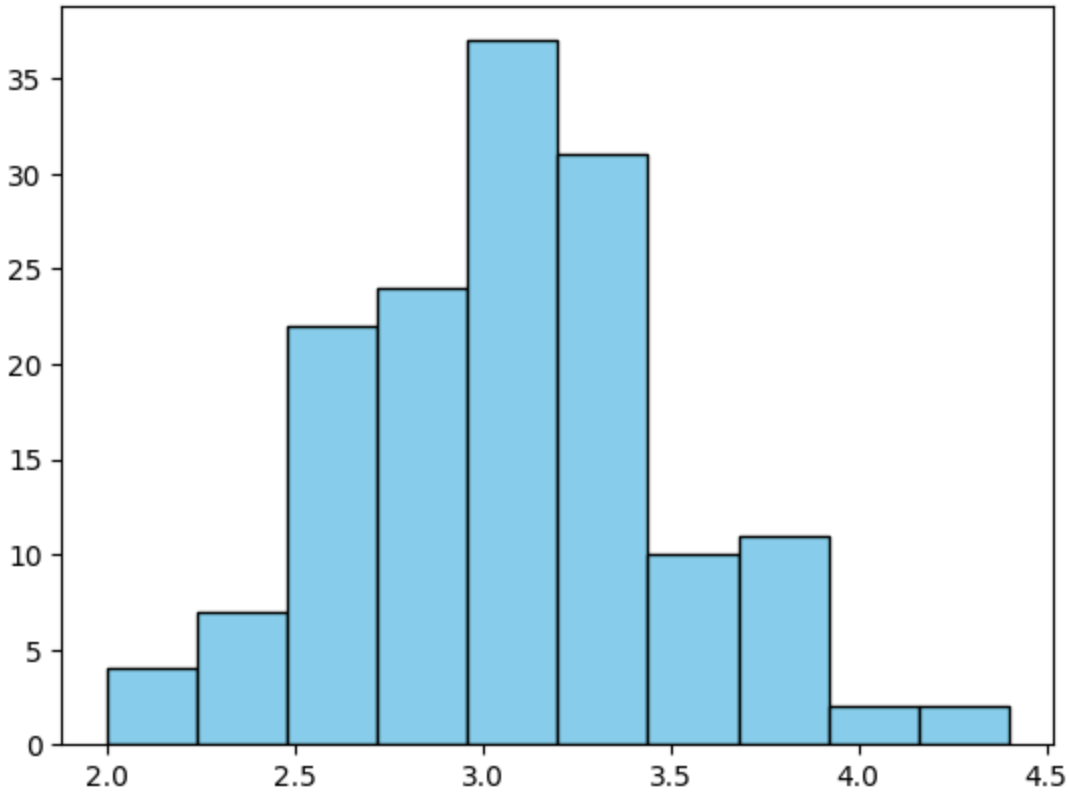
```
from sklearn import datasets
iris = datasets.load_iris()
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

df = pd.DataFrame(iris.data, columns=iris.feature_names)
df['']
data = { "weight": [4.17, 5.58, 5.18, 6.11, 4.50, 4.61, 5.17, 4.53, 5.33, 5.14, 4.81, 4.17,
PlantGrowth = pd.DataFrame(data)
#df['target'] = iris.target


plt.hist(df['sepal width (cm)'], color='skyblue', edgecolor='black')
```

```
(array([ 4.,  7., 22., 24., 37., 31., 10., 11.,  2.,  2.]),
 array([2.  , 2.24, 2.48, 2.72, 2.96, 3.2 , 3.44, 3.68, 3.92, 4.16, 4.4 ]),
 <BarContainer object of 10 artists>)
```



I would expect the mean to be higher because the histogram is right skewed.

```
mean = np.mean(df['sepal width (cm)'])
median = np.median(df['sepal width (cm)'])
print("mean: ", mean)
print("median: ", median)
```

```
mean:  3.0573333333333337
median:  3.0
```

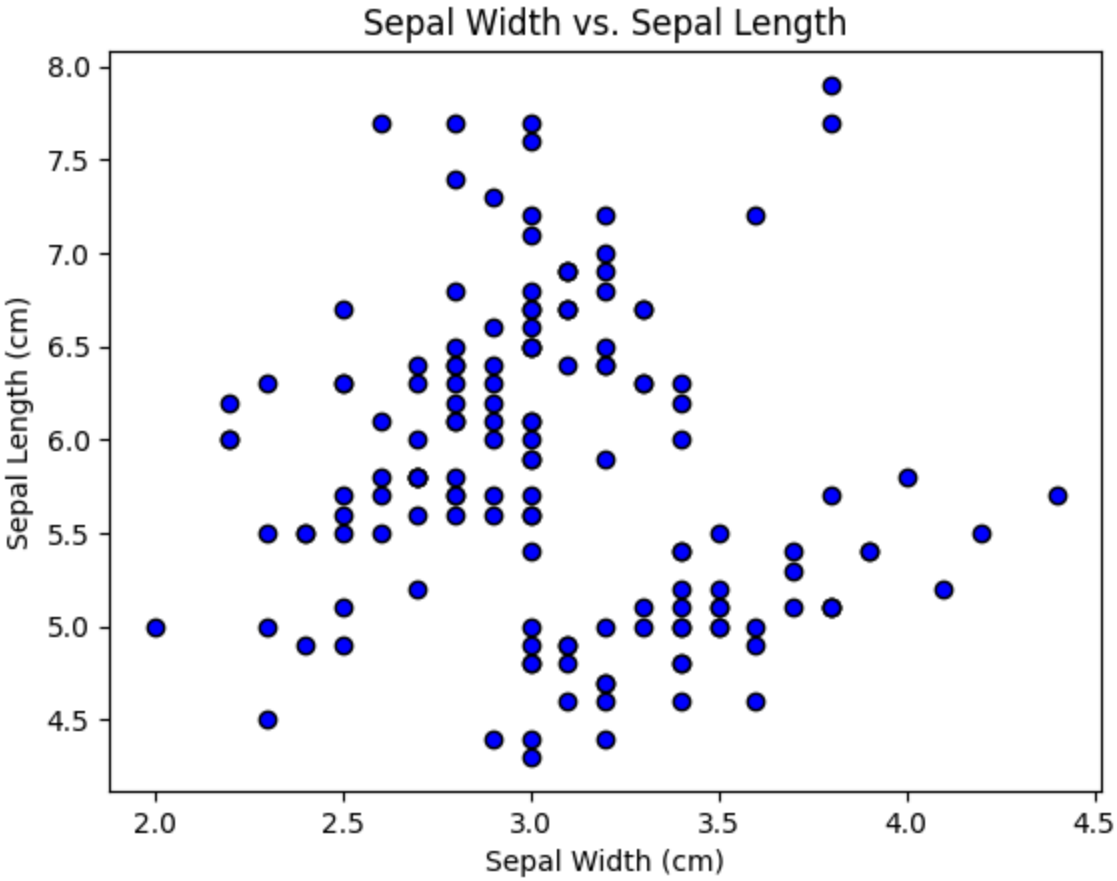
```
print(np.percentile(df['sepal width (cm)'],73))
```

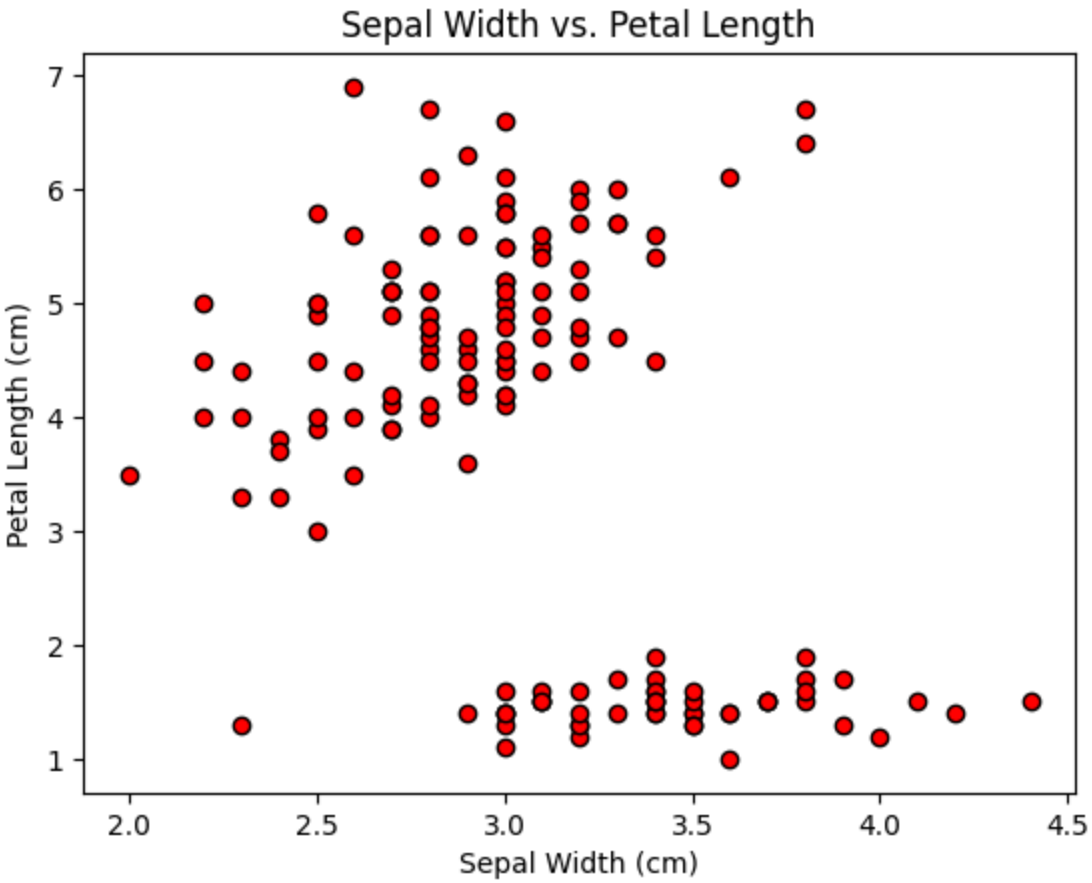
```
3.3
```

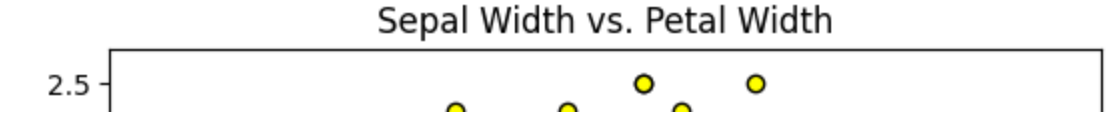
Only 27% of the flowers have a Sepal.Width higher than 3.3 cm.

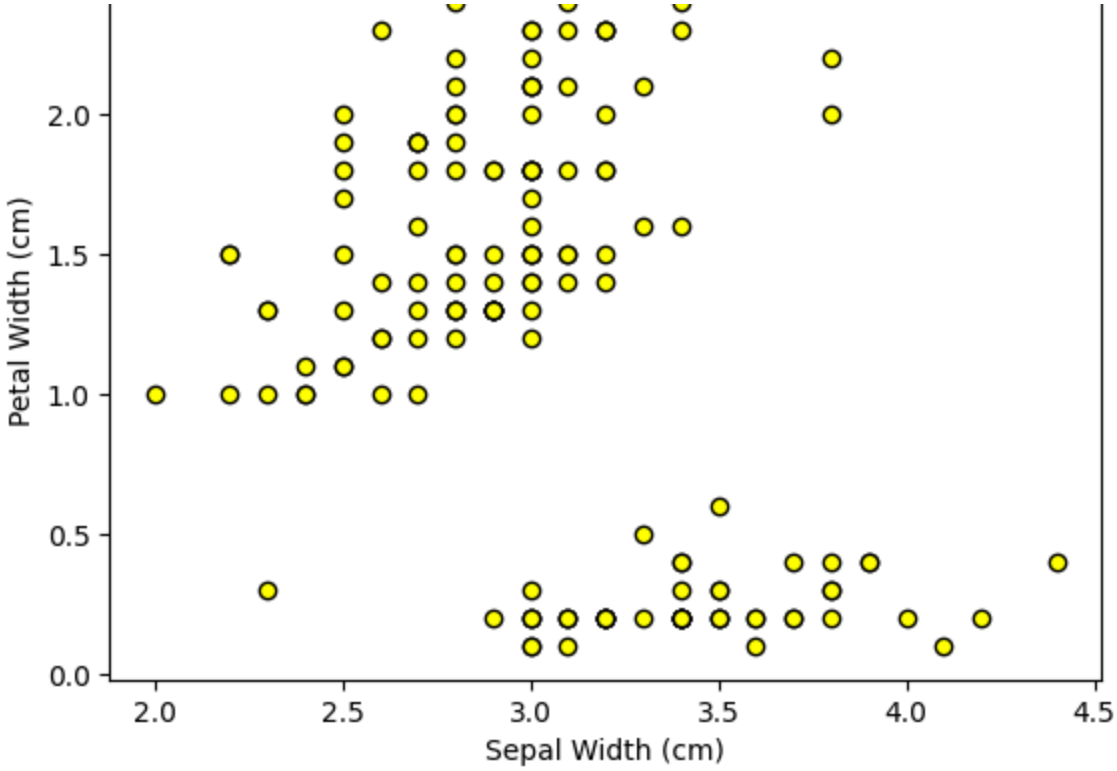
```
plt.scatter(df['sepal width (cm)'], df['sepal length (cm)'], color='blue', edgecolor='black'
plt.xlabel('Sepal Width (cm)')
plt.ylabel('Sepal Length (cm)')
plt.title('Sepal Width vs. Sepal Length')

plt.show()

plt.scatter(df['sepal width (cm)'], df['petal length (cm)'], color='red', edgecolor='black')
plt.xlabel('Sepal Width (cm)')
plt.ylabel('Petal Length (cm)')
plt.title('Sepal Width vs. Petal Length')

plt.show()

plt.scatter(df['sepal width (cm)'], df['petal width (cm)'], color='yellow', edgecolor='black
plt.xlabel('Sepal Width (cm)')
plt.ylabel('Petal Width (cm)')
plt.title('Sepal Width vs. Petal Width')

plt.show()

plt.scatter(df['sepal length (cm)'], df['petal length (cm)'], color='green', edgecolor='blac
plt.xlabel('Sepal Length (cm)')
plt.ylabel('Petal Length (cm)')
plt.title('Sepal Length vs. Petal Length')

plt.show()

plt.scatter(df['sepal length (cm)'], df['petal width (cm)'], color='orange', edgecolor='blac
plt.xlabel('Sepal Length (cm)')
plt.ylabel('Petal Width (cm)')
plt.title('Sepal Length vs. Petal Width')

plt.show()

plt.scatter(df['petal length (cm)'], df['petal width (cm)'], color='pink', edgecolor='black'
plt.xlabel('Petal Length (cm)')
plt.ylabel('Petal Width (cm)')
plt.title('Petal Length vs. Petal Width')
```

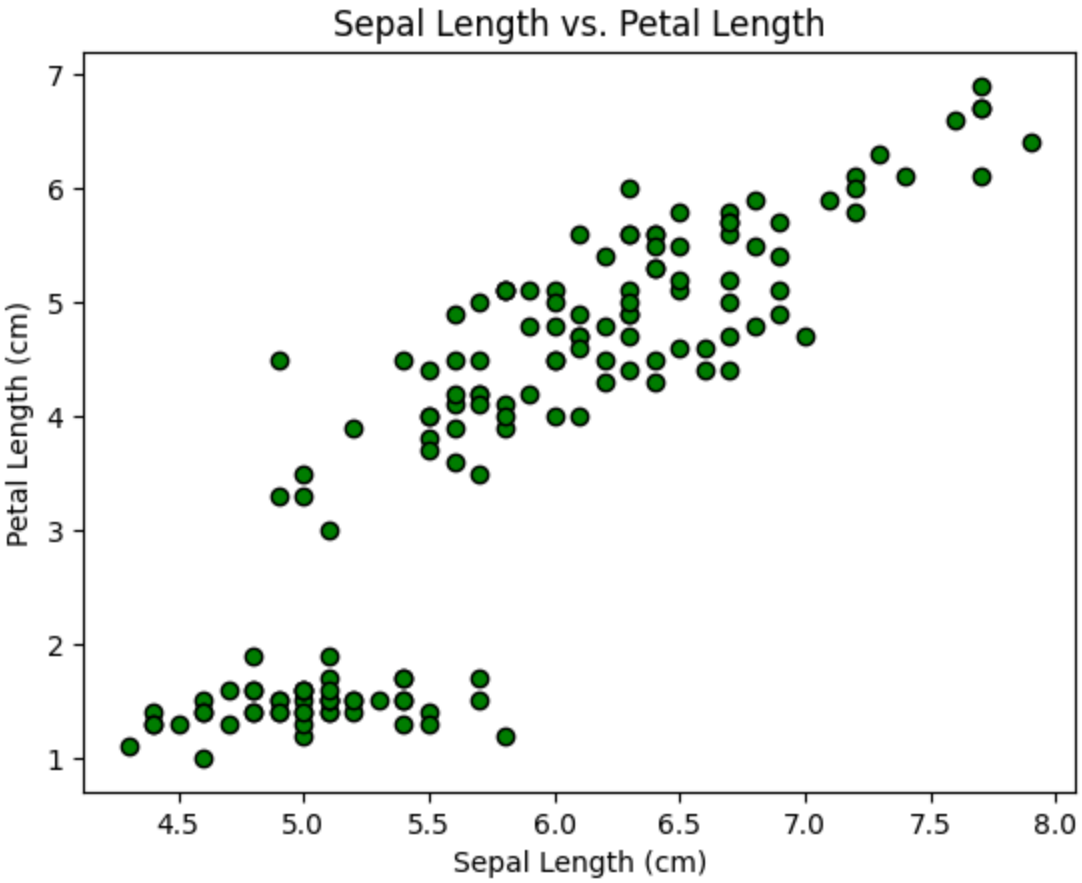
```
    plt.show()
```

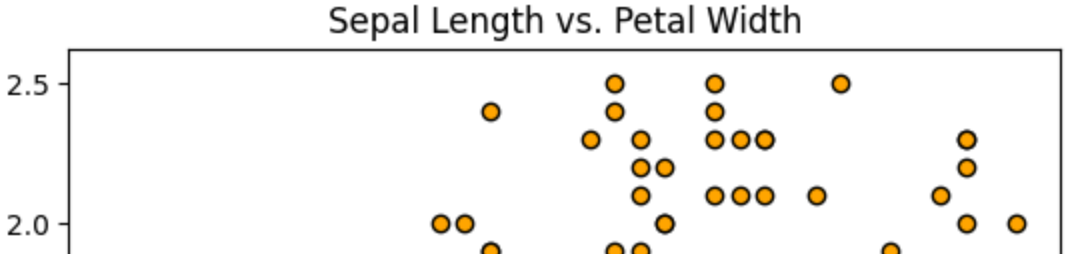

Sepal Width vs. Sepal Length
Sepal Length (cm)
Sepal Width (cm)
8.0
7.5
7.0
6.5
6.0
5.5
5.0
4.5
2.0
2.5
3.0
3.5
4.0
4.5

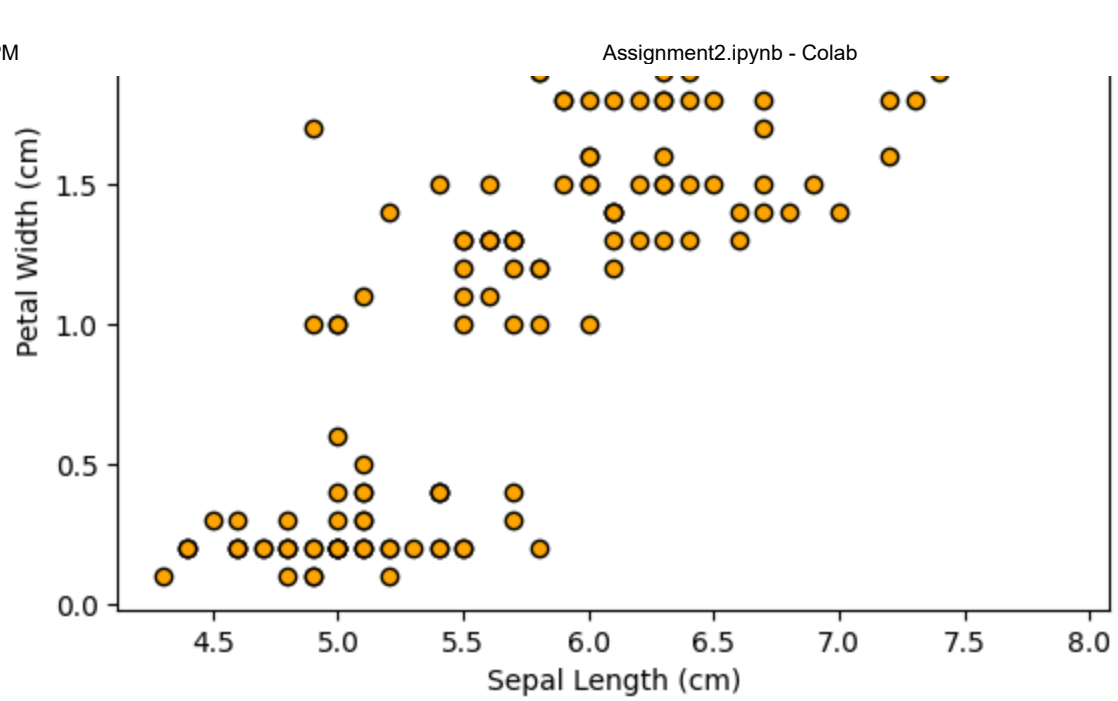

Sepal Width vs. Petal Length
Petal Length (cm)
Sepal Width (cm)
7
6
5
4
3
2
1
2.0
2.5
3.0
3.5
4.0
4.5

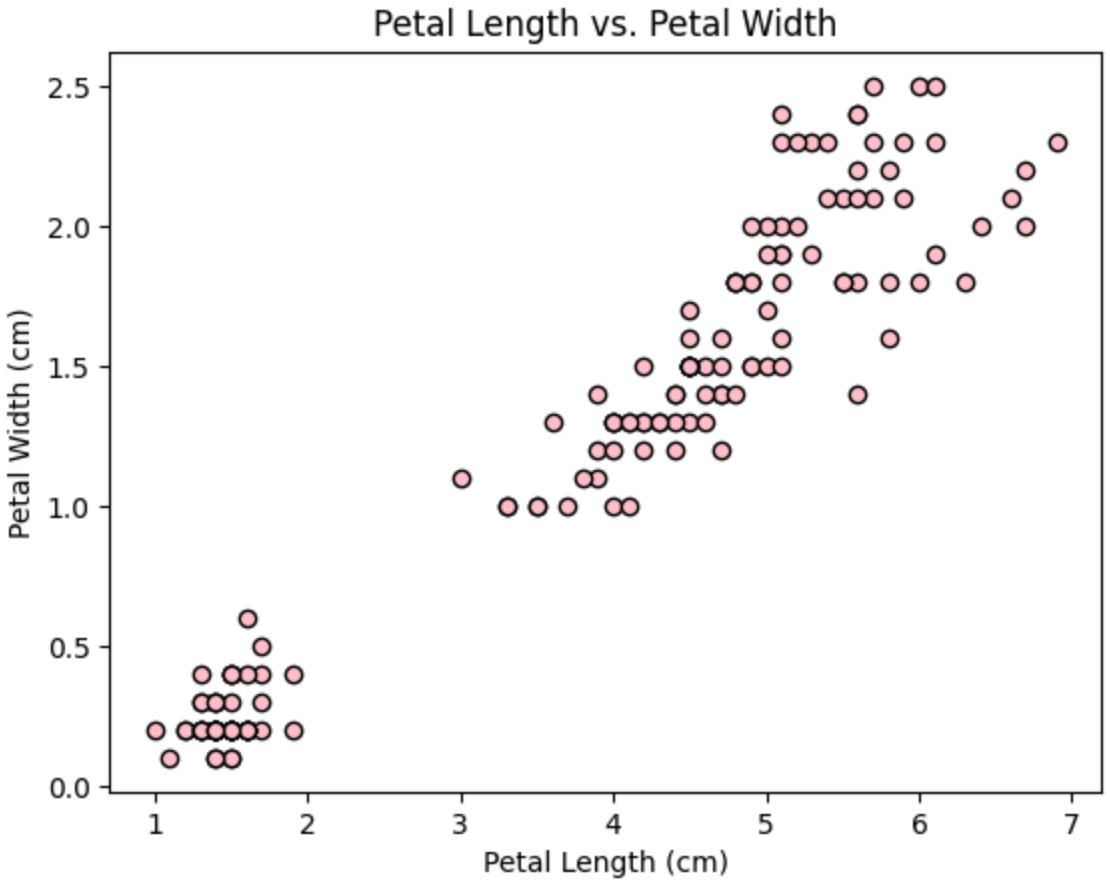

Sepal Width vs. Petal Width
2.5


Petal Width (cm)
Sepal Width (cm)
0.0
0.5
1.0
1.5
2.0
2.0
2.5
3.0
3.5
4.0
4.5


Sepal Length vs. Petal Length
Petal Length (cm)
Sepal Length (cm)
1
2
3
4
5
6
7
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0


Sepal Length vs. Petal Width
2.0
2.5

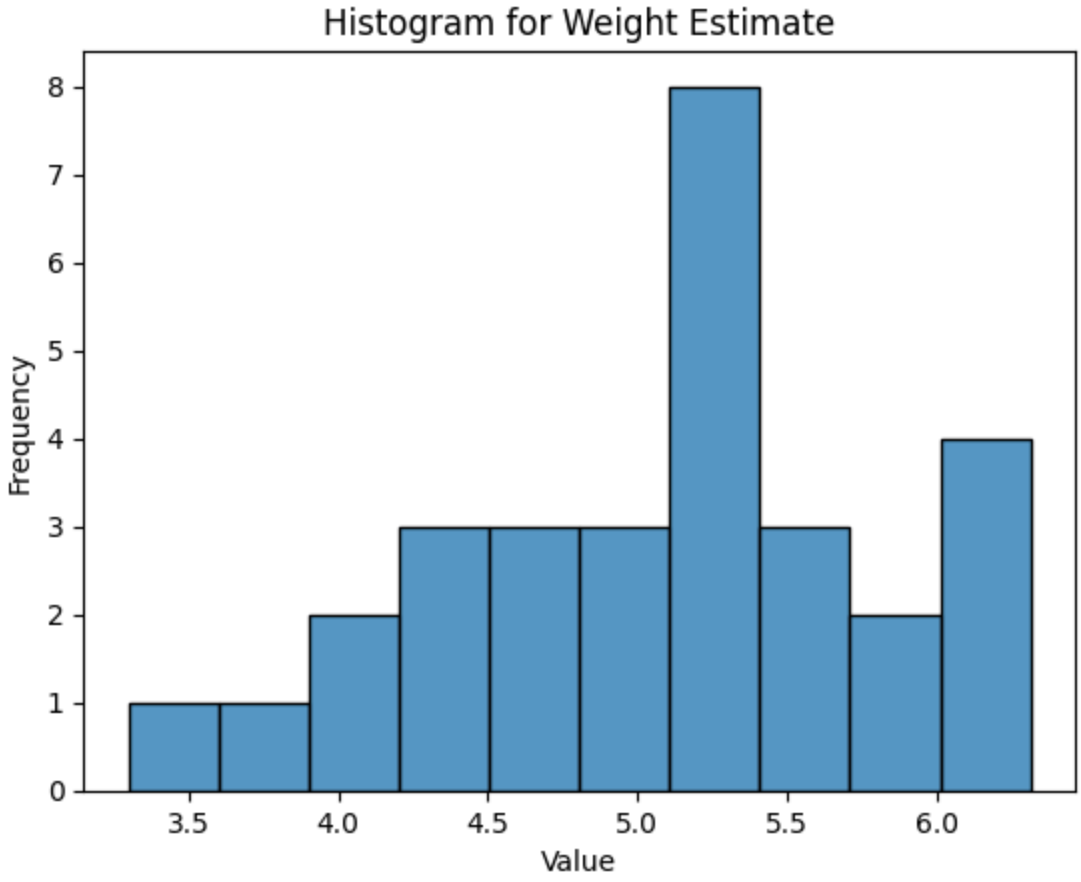

Petal Width (cm)
0.0
0.5
1.0
1.5
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
Sepal Length (cm)


Petal Length vs. Petal Width
Petal Width (cm)
0.0
0.5
1.0
1.5
2.0
2.5
1
2
3
4
5
6
7
Petal Length (cm)

Based on the scatterplots, petal length and petal width have the strongest relationship, and sepal length and sepal width have the weakest relationship.

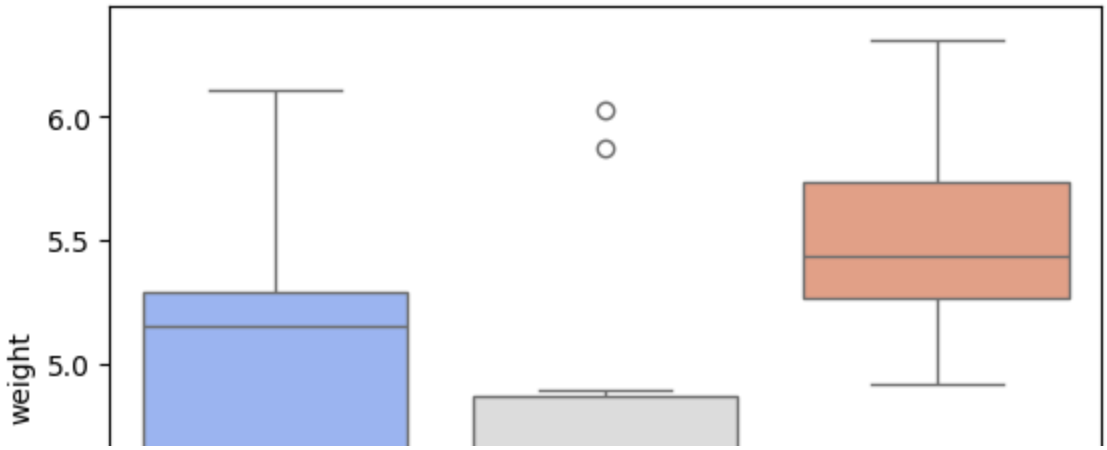
```
sns.histplot(PlantGrowth['weight'], binwidth=.3, binrange=(3.3,PlantGrowth['weight'].max()))
plt.title('Histogram for Weight Estimate')
plt.xlabel('Value')
plt.ylabel('Frequency')
```

```
Text(0, 0.5, 'Frequency')
```




```
sns.boxplot(x='group', y='weight', data=PlantGrowth, palette='coolwarm')
plt.show()
```

```
<ipython-input-25-bb99071beaff>:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0.

  sns.boxplot(x='group', y='weight', data=PlantGrowth, palette='coolwarm')
```



The IQR for trt1 based off of the boxplots above is approximately 4.85 - 4.2 = .65 and 1.5 * .65 = .975. Thus approximately 97.5% of trt1 weights are blow the minimum trt2 weight.