Grad505_Assignment_Week3

February 1, 2026

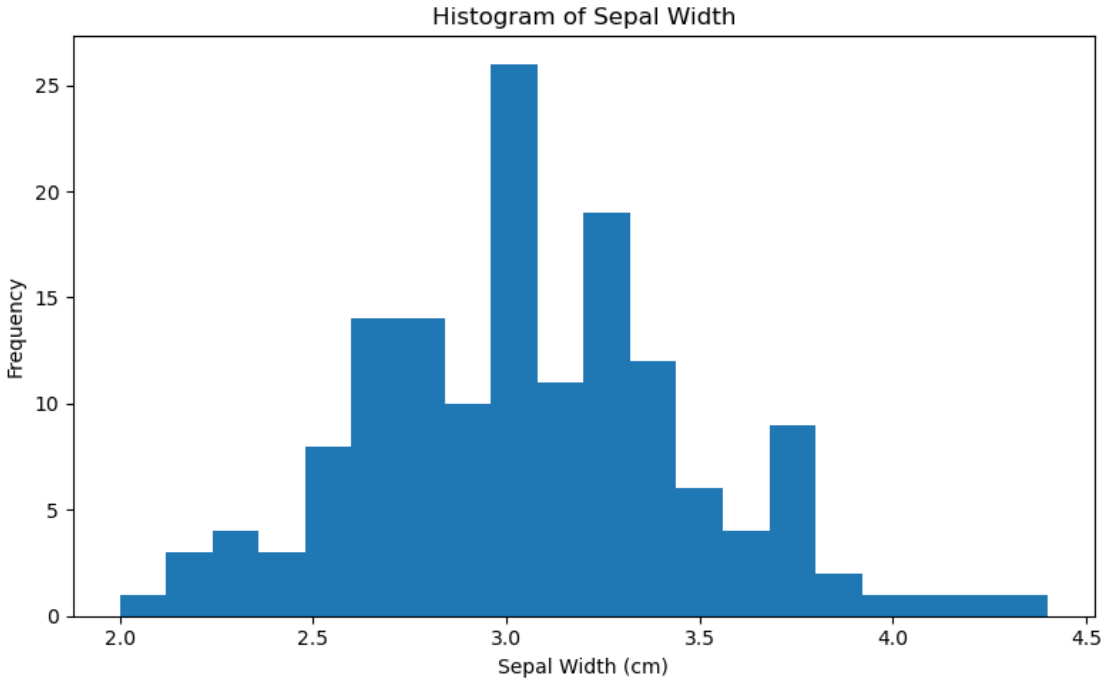
[74]:
```
#Author: Reid Sroda
#Assignment: Week 3 Assignment 1 of 2
#Date: 01/31/2026
```

[33]:
```
from sklearn import datasets
import pandas as pd
import matplotlib.pyplot as plt
import itertools
```

[15]:
```
iris = datasets.load_iris(as_frame = True)
df = iris.frame

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, 4.41, 3.59, 5.87, 3.83, 6.03, 4.89, 4.32, 4.69, 6.31, 5.12, 5.
 ↪54, 5.50, 5.37, 5.29, 4.92, 6.15, 5.80, 5.26], "group": ["ctrl"] * 10 +␣
 ↪["trt1"] * 10 + ["trt2"] * 10}
PlantGrowth = pd.DataFrame(data)
```

1 Question 1

[25]:
```
#1a
plt.figure(figsize=(8,5))
plt.hist(df['sepal width (cm)'], bins = 20)
plt.xlabel('Sepal Width (cm)')
plt.ylabel('Frequency')
plt.title('Histogram of Sepal Width')

plt.tight_layout()
plt.show()
```



#1b Based on the graph, because the data seems to be right skewed I would assume the mean is higher than the median

[24]:
```
#1c
print(f"Mean: {df['sepal width (cm)'].mean()}")
print(f"Median: {df['sepal width (cm)'].median()}")
```

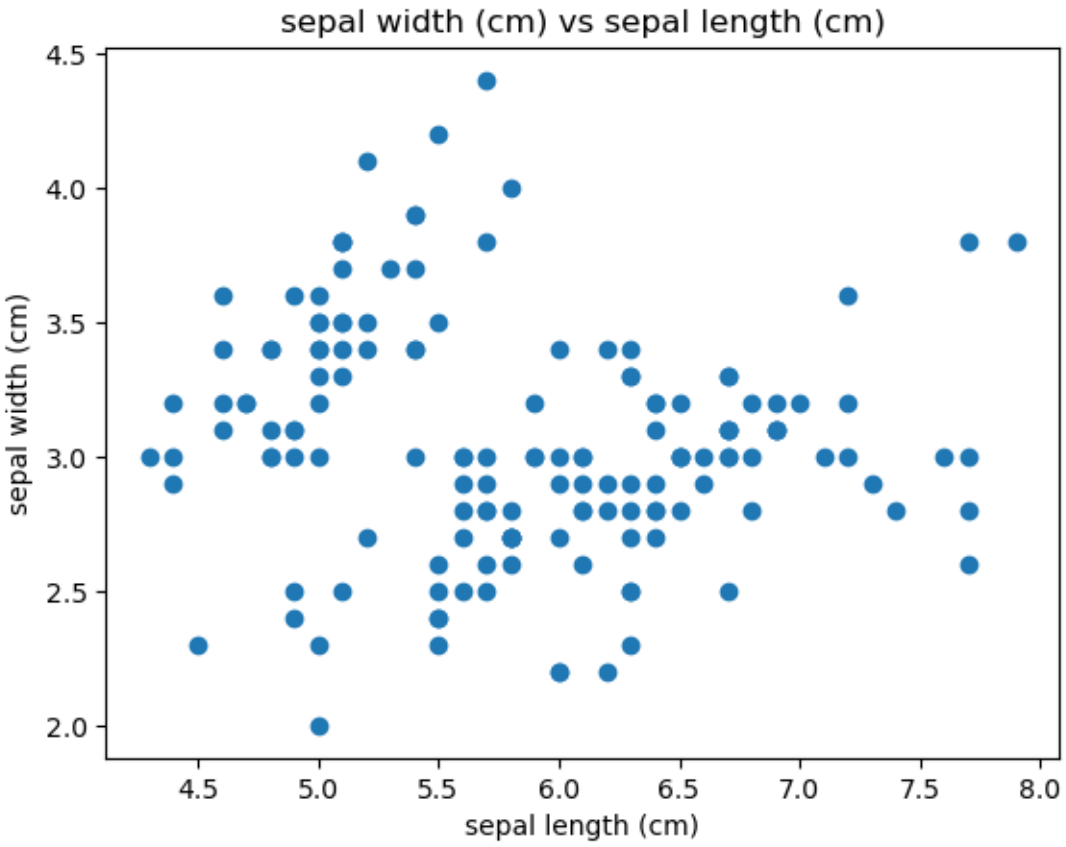
```
Mean: 3.057333333333334
Median: 3.0
```

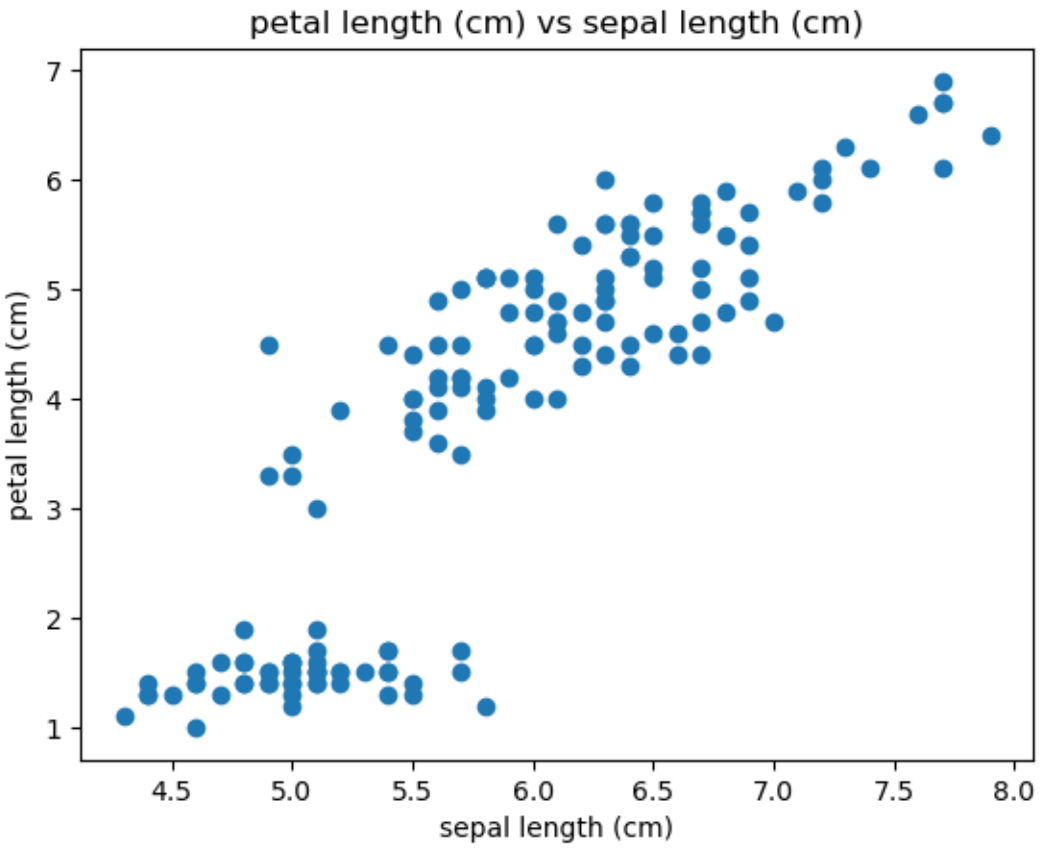
[28]:
```
#1d
import numpy as np
result = np.percentile(df['sepal width (cm)'], 100-27)
print(f"Only 27% of flower have a width higher than {result} cm")
```

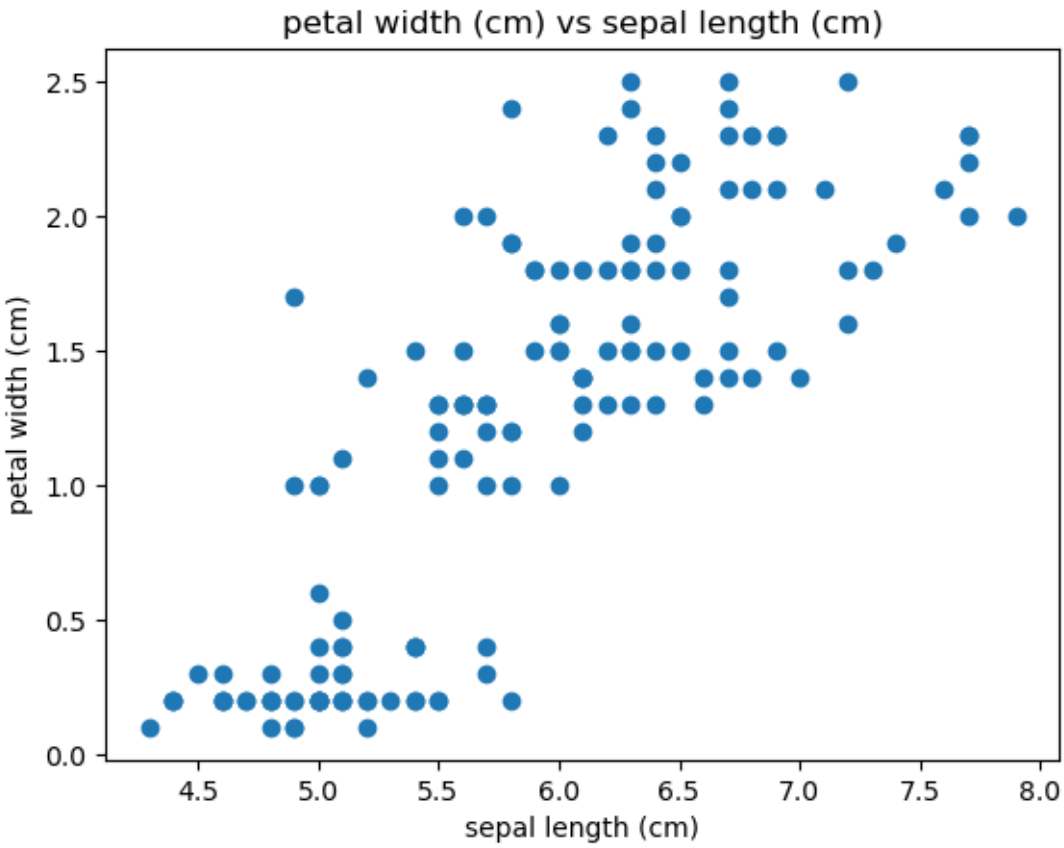
```
Only 27% of flower have a width higher than 3.3 cm
```

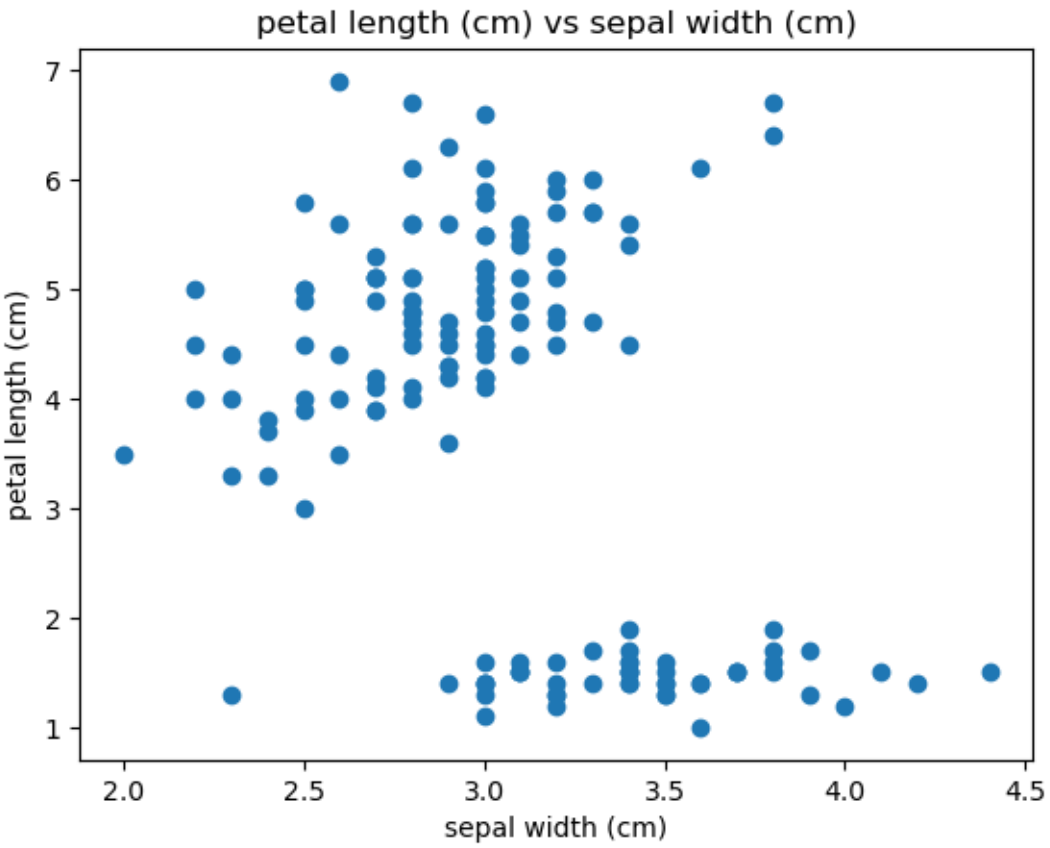
[34]:
```
#1e
features = [
    "sepal length (cm)",
    "sepal width (cm)",
    "petal length (cm)",
    "petal width (cm)"
]
```

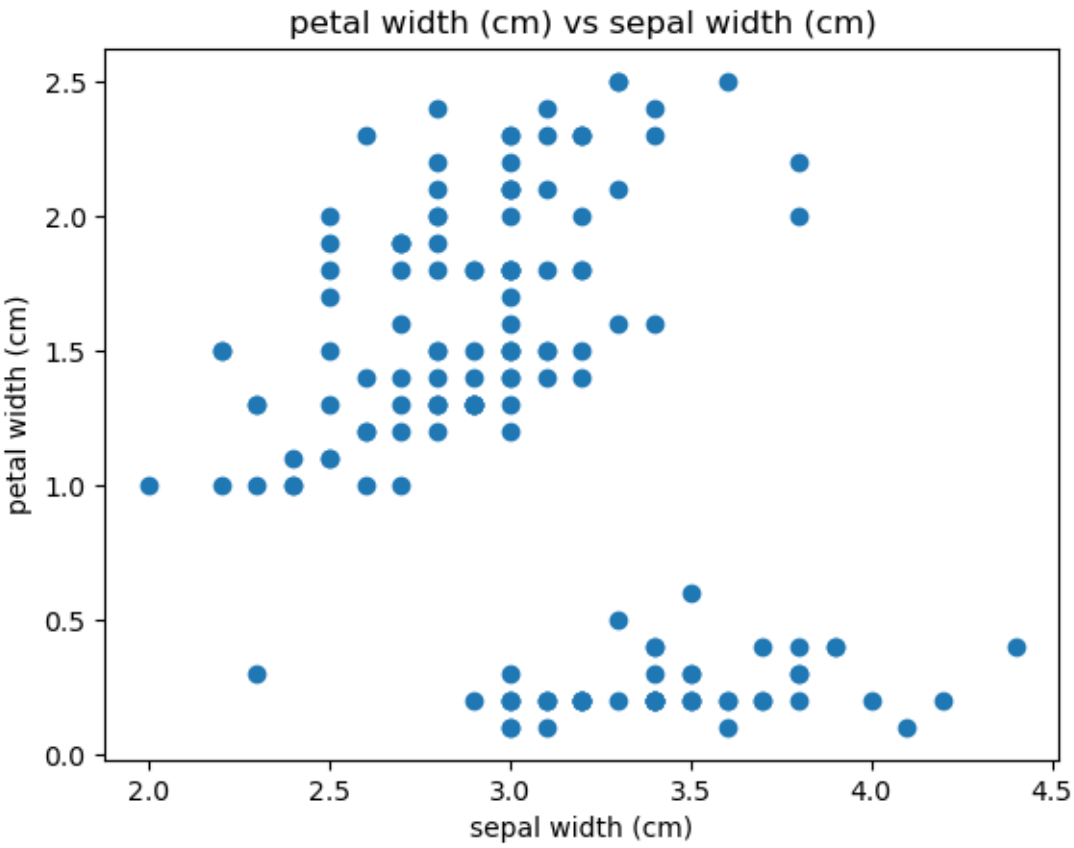
```
pairs = list(itertools.combinations(features, 2))

for x, y in pairs:
    plt.figure()
    plt.scatter(df[x], df[y])
    plt.xlabel(x)
    plt.ylabel(y)
    plt.title(f"{y} vs {x}")
    plt.show()
```

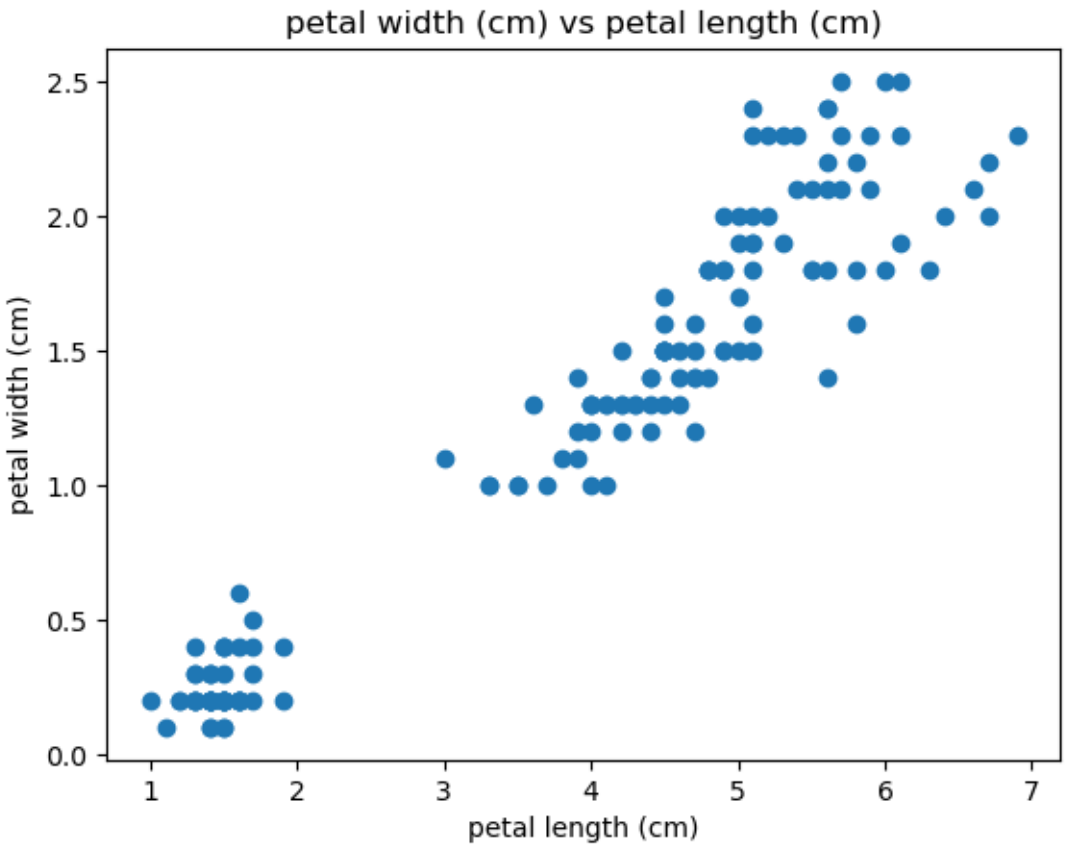



petal length (cm) vs sepal length (cm)
petal length (cm)
sepal length (cm)
1
2
3
4
5
6
7
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0


petal width (cm) vs sepal length (cm)
petal width (cm)
sepal length (cm)
0.0
0.5
1.0
1.5
2.0
2.5
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0


petal length (cm) vs sepal width (cm)
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

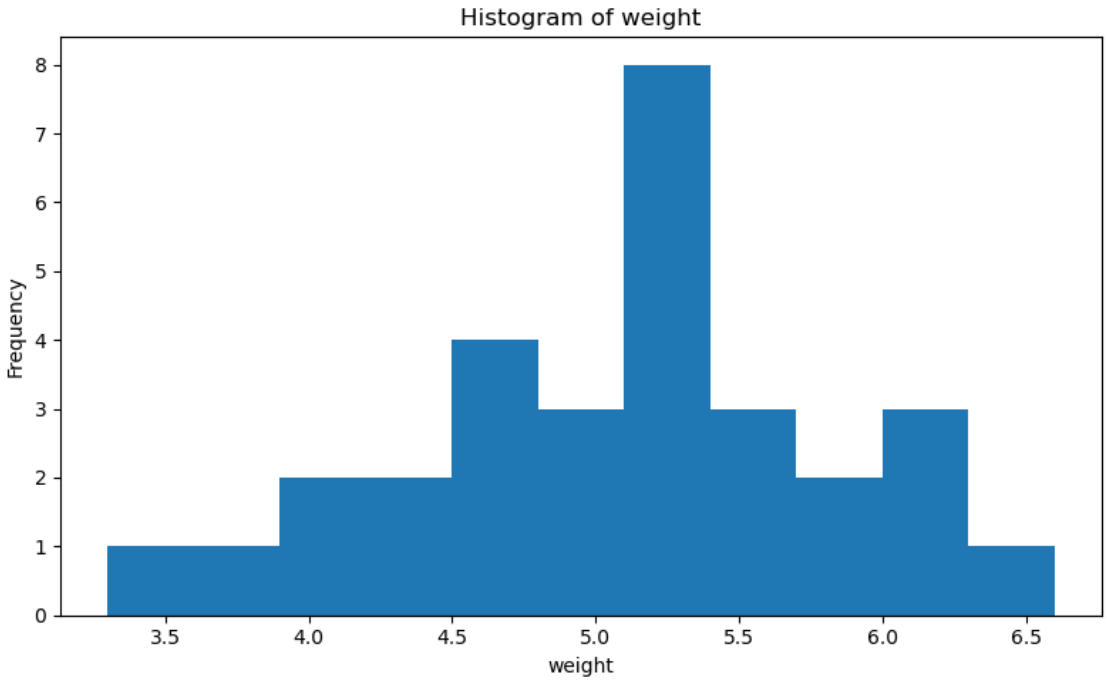

petal width (cm) vs sepal width (cm)
petal width (cm)
sepal width (cm)
2.5
2.0
1.5
1.0
0.5
0.0
2.0
2.5
3.0
3.5
4.0
4.5



1f. Based on the graphs, petal length and petal width seem to have the strongest relationship. Sepal width and sepal length seem to have the weakest relationship

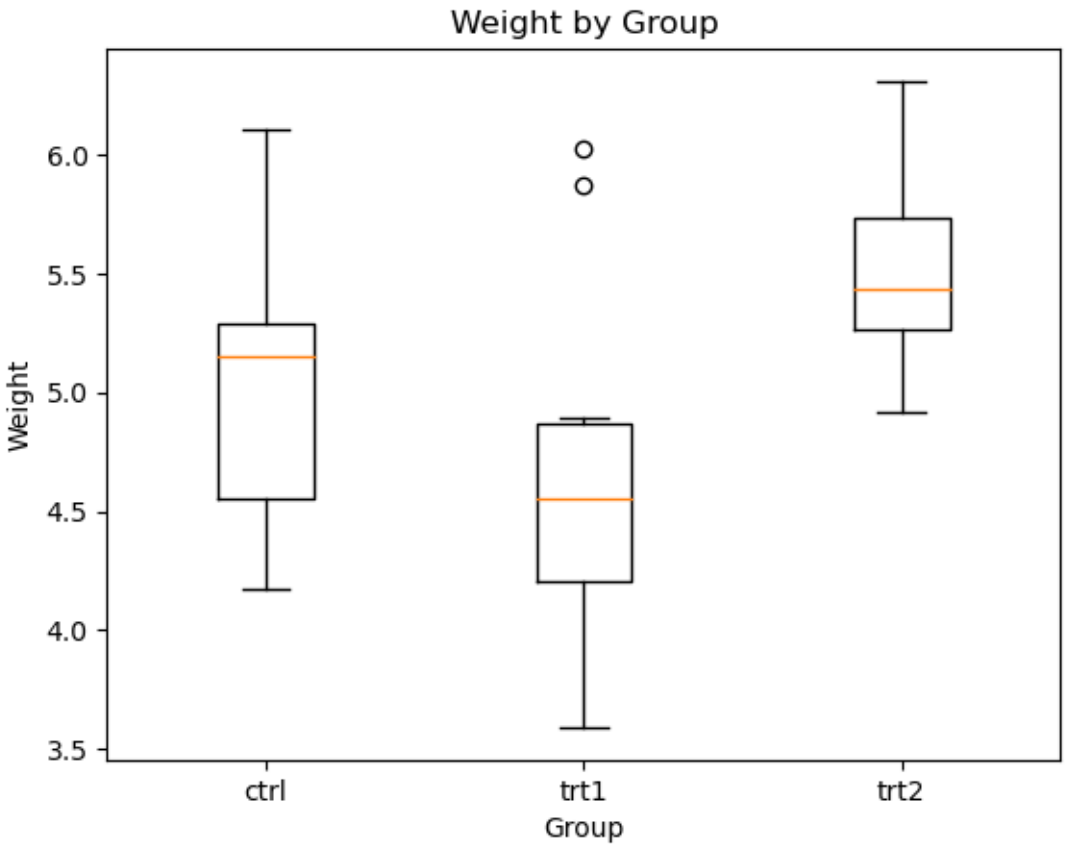
2 #2

[37]:
```
#2a
bins = np.arange(3.3, PlantGrowth["weight"].max() + 0.3, 0.3)

plt.figure(figsize=(8,5))
plt.hist(PlantGrowth['weight'], bins = bins)
plt.xlabel('weight')
plt.ylabel('Frequency')
plt.title('Histogram of weight')

plt.tight_layout()
plt.show()
```



[56]:
```
#2b
groups = PlantGrowth['group'].unique()

data = [PlantGrowth[PlantGrowth['group'] == i]['weight'] for i in groups]

plt.figure()
plt.boxplot(data, labels = groups)
plt.xlabel('Group')
plt.ylabel('Weight')
plt.title('Weight by Group')
```

[56]:
```
Text(0.5, 1.0, 'Weight by Group')
```



#2c: ~75% of weights

[69]:
```
#2d
min_trt2 = min(PlantGrowth[PlantGrowth['group'] == 'trt2']['weight'])
num = sum(PlantGrowth[PlantGrowth['group'] == 'trt1']['weight'] < min_trt2)
den = len(PlantGrowth[PlantGrowth['group'] == 'trt1'])
result = num / den * 100
print(f"There are {result}% of trt1 weights less than the minimum trt2 weight")
```

```
There are 80.0% of trt1 weights less than the minimum trt2 weight
```

[73]:
```
#2e
data = PlantGrowth[PlantGrowth['weight'] > 5.5]
group_counts = data['group'].value_counts()

colors = ['red', 'green', 'blue']

plt.figure(figsize=(10, 6))
plt.bar(group_counts.index, group_counts.values, color = colors)
plt.xlabel('Group')
plt.ylabel('Count')
```

```
plt.title('Plants with Weight  5.5 by Group')
plt.show()
```