Lab04 Solutions

PSTAT 5A, compiled by Ethan

April 25, 2023

1 Task 1

[1]:
```
## part (a)
import numpy as np
```

[2]:
```
np.sin(0) # should be 0
```

[2]:
```
0.0
```

[3]:
```
## part (b)
import datascience as ds
```

[4]:
```
ds.Table().with_columns(
  "Col1", [1, 2, 3],
  "Col2", [2, 3, 4]
)
```

[4]:
```
Col1 | Col2
1    | 2
2    | 3
3    | 4
```

2 Task 2

[5]:
```
x_list = [1, 2, 3]
x_array = ds.make_array([1, 2, 3])
```

[6]:
```
np.mean(x_list)
```

[6]:
```
2.0
```

[7]:
```
np.mean(x_array)
```

[7]:
```
2.0
```

3 Task 3

`np.ptp()` computes the **p**eak **t**o **p**eak (i.e. Range) of a dataset.

```
[8]: np.ptp(x_list)
```

```
[8]: 2
```

```
[9]: np.ptp(x_array)
```

```
[9]: 2
```

4 Task 4

Part (a) The mean of `x_list` is 2, so

$$s_x^2 = \frac{1}{3-1}[(1-2)^2 + (2-2)^2 + (3-2)^2] = \frac{1}{2}(1+1) = 1$$

meaning the standard deviation of `x` is $\sqrt{1} = \boxed{1}$

Part (b)

```
[10]: np.std(x_list)
```

```
[10]: 0.816496580927726O3
```

This does **not** agree with our answer to part (a) above.

Part (c): Following the instructions, we compute

$$\frac{1}{3}[(1-2)^2 + (2-2)^2 + (3-2)^2] = \frac{1}{3}(1+1) = \frac{2}{3}$$

so this modified standard deviation would be $\sqrt{2/3}$:

```
[11]: np.sqrt(2/3)
```

```
[11]: 0.816496580927726O3
```

This agrees with our answer obtained from `np.std()`, so it seems `np.std()` uses a denominator of n as opposed to $n-1$.

Part (d)

```
[12]: np.std(x_list, ddof = 1)
```

```
[12]: 1.0
```

This **does** match with our answer to part (a) above.

5 Plotting Code

[13]:

```
%matplotlib inline
import matplotlib
import matplotlib.pyplot as plt
plt.style.use('seaborn-v0_8-whitegrid')
```

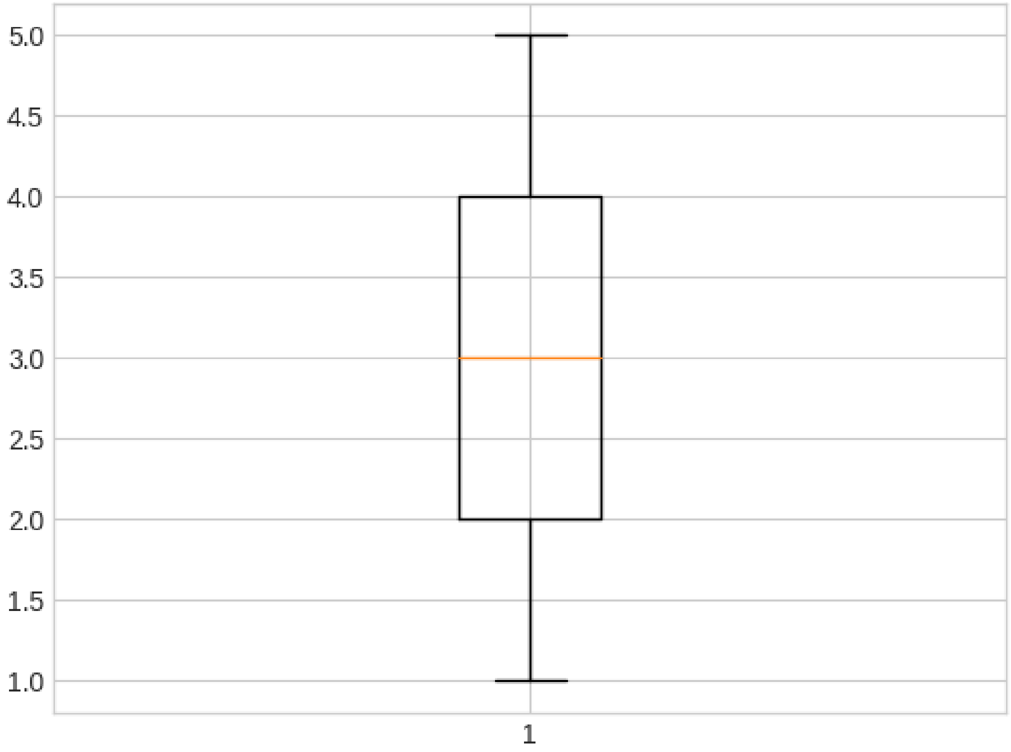
6 Task 5

[14]:

```
# part (a)
y = [1, 2, 3, 4, 5, 4, 3, 5, 4, 1, 2]
```

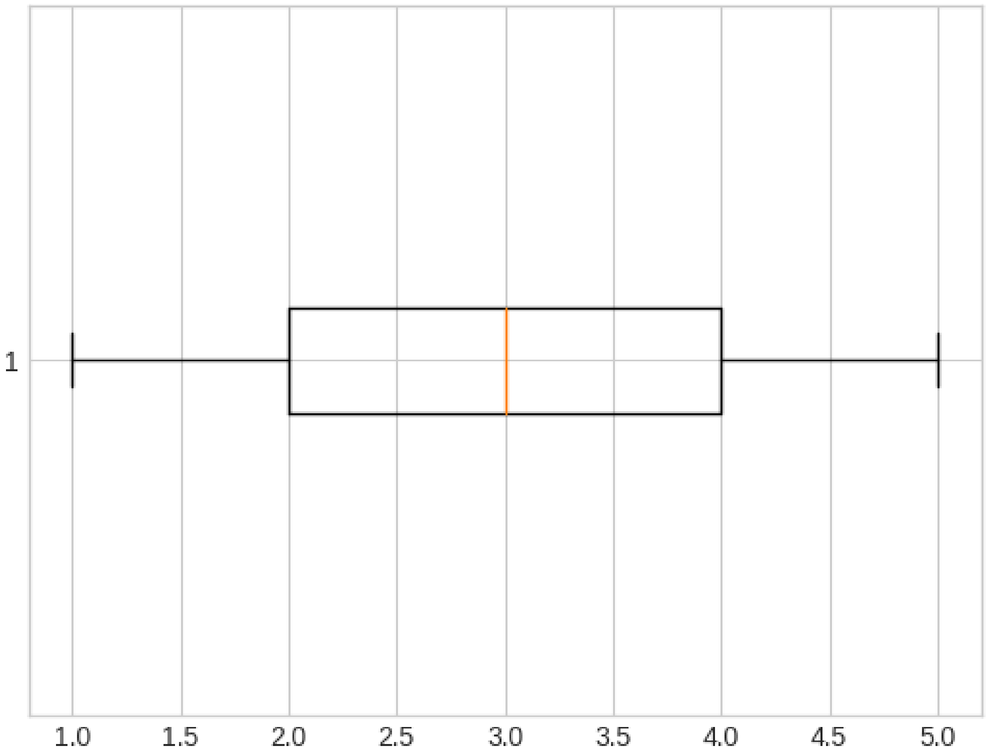
[15]:

```
# part (b)
plt.boxplot(y);
```

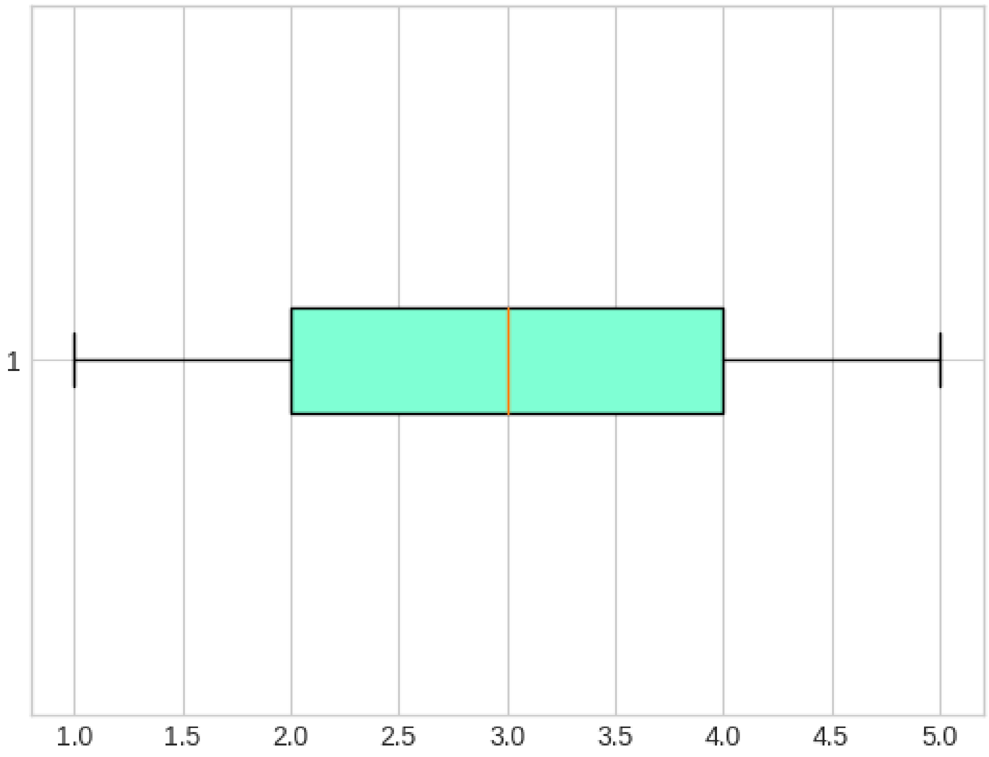


[16]:

```
# part (c)
plt.boxplot(y, vert = False);
```

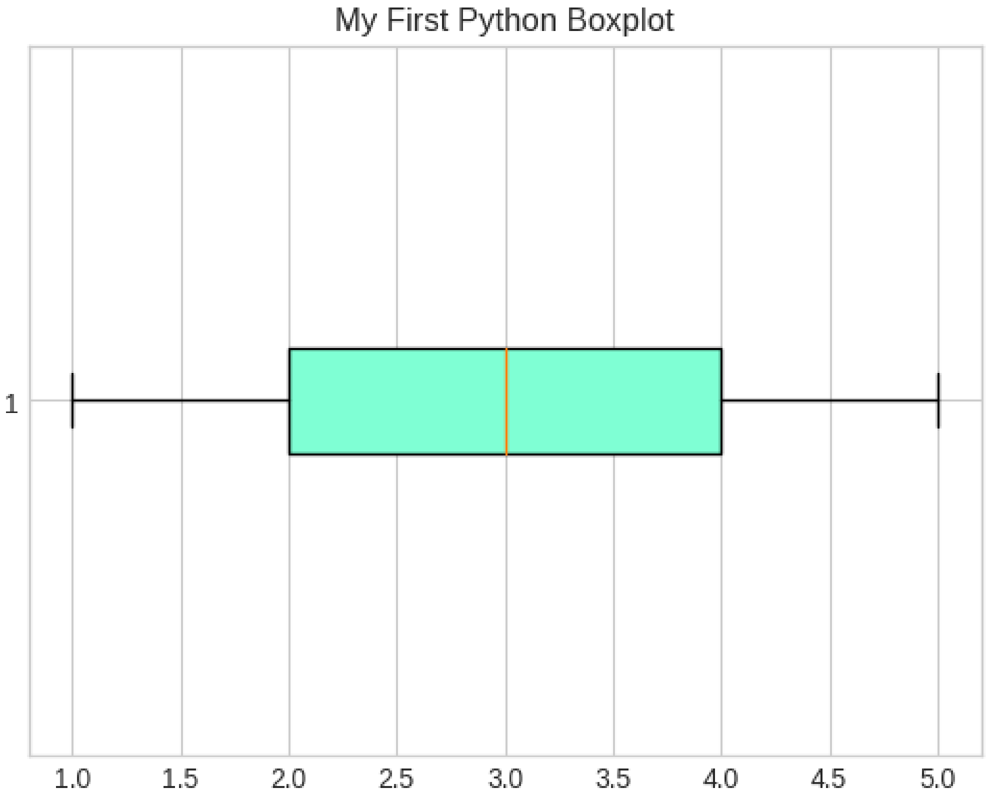


[17]:
```
# part (d)
plt.boxplot(y, vert = False,
            patch_artist=True,
            boxprops = dict(facecolor = "aquamarine"));
```



[18]:

```
# part (e)
plt.boxplot(y, vert = False,
            patch_artist=True,
            boxprops = dict(facecolor = "aquamarine"));
plt.title("My First Python Boxplot");
```

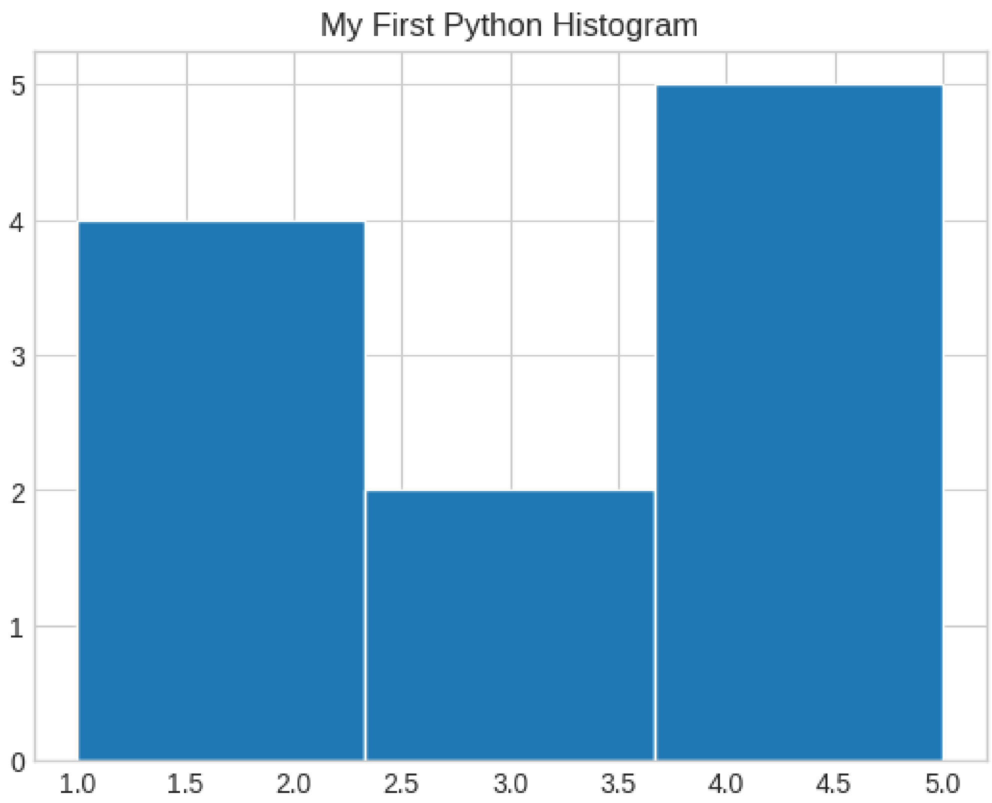


```
[19]: ## part (f)
      ## from the boxplot, IQR appears to be 2
      np.diff(np.percentile(y, [25,75]))[0]
```

```
[19]: 2.0
```

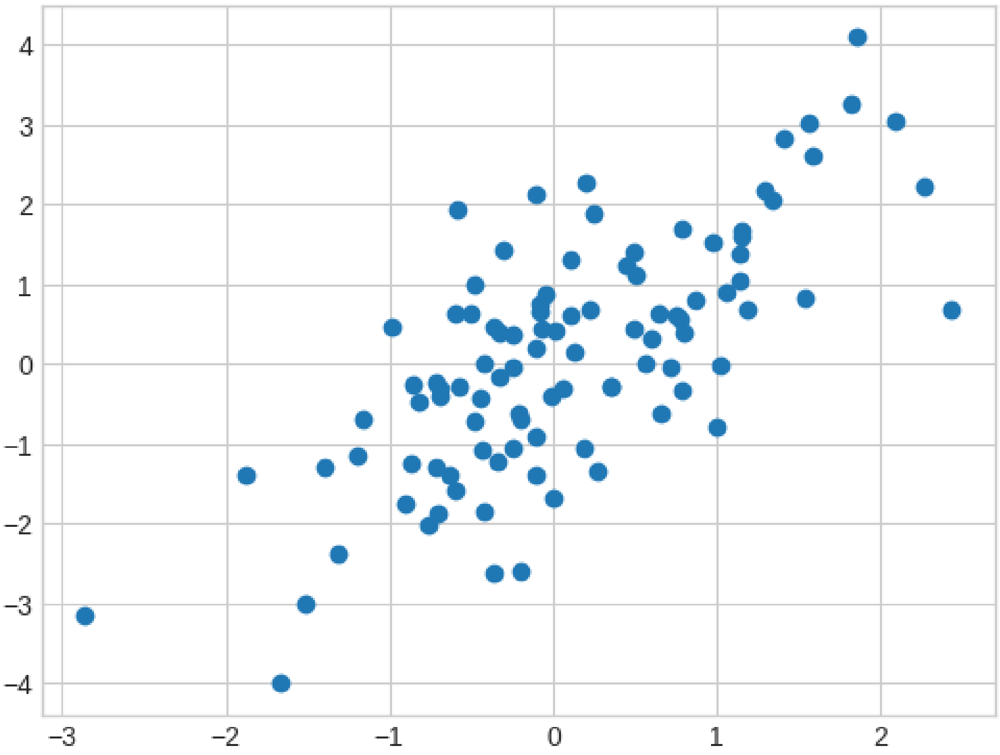
7 Task 6

```
[20]: plt.hist(y,
               bins = 3,
               edgecolor = "white");
      plt.title("My First Python Histogram");
```

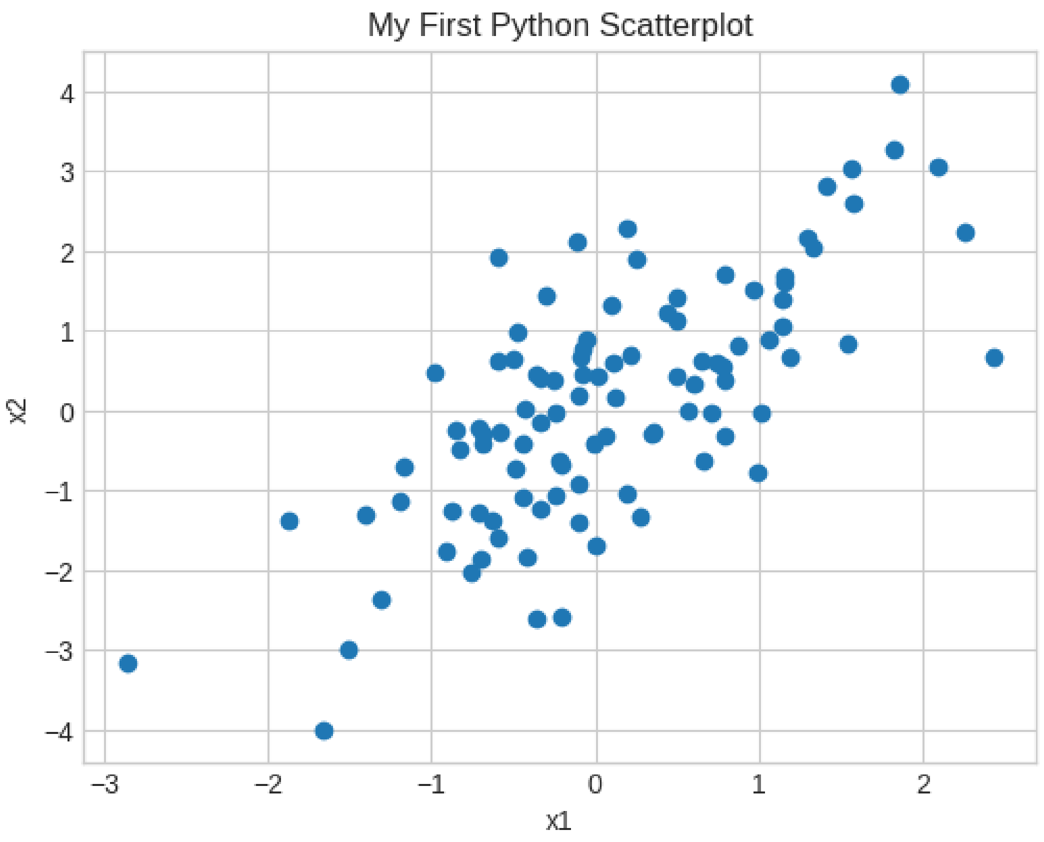


8 Task 7

[21]:
```
## part (a)

np.random.seed(5)

x1 = np.random.normal(0, 1, 100)
x2 = x1 + np.random.normal(0, 1, 100)

plt.scatter(x1, x2);
```



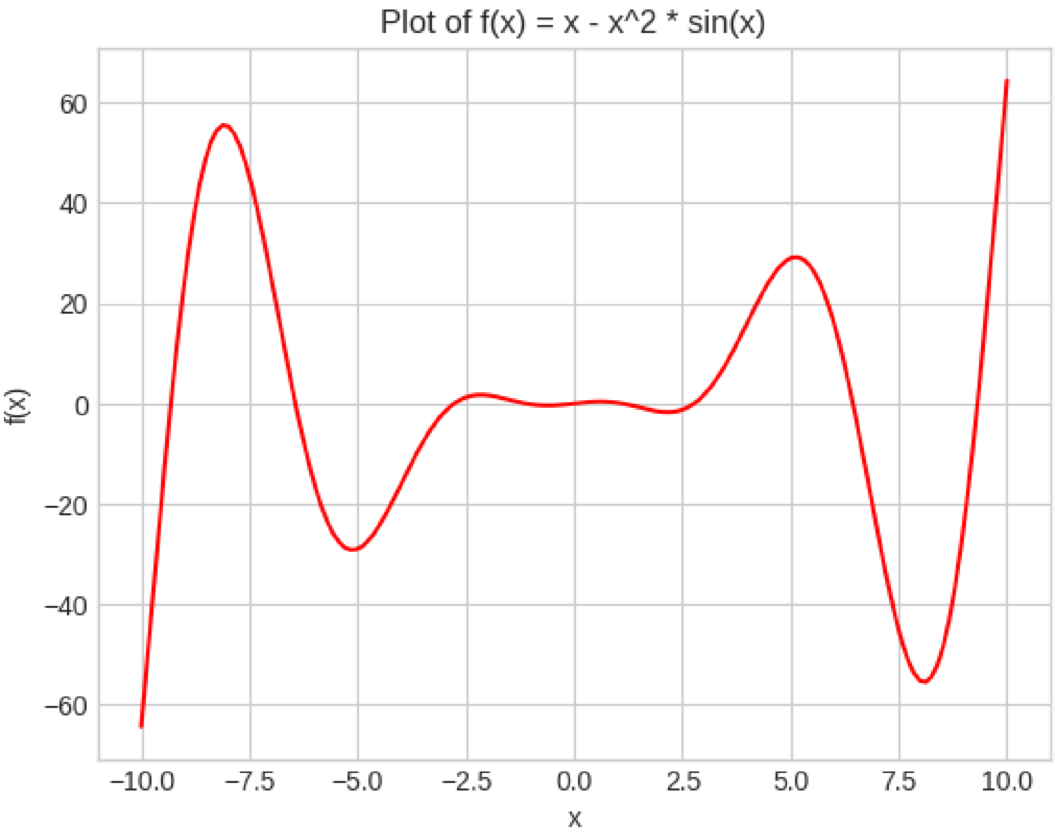
[22]:
```
## part (b)
plt.scatter(x1, x2);
plt.xlabel("x1");
plt.ylabel("x2");
plt.title("My First Python Scatterplot");
```



Part (c): Yes, there appears to be a positive linear trend.

9 Task 8

[23]:
```
def f(x):
    """return x - x^2 * sin(x)"""
    return x - (x ** 2) * np.sin(x)

x = np.linspace(-10, 10, 150)
plt.plot(x, f(x), color = "red");
plt.xlabel("x");
plt.ylabel("f(x)");
plt.title("Plot of f(x) = x - x^2 * sin(x)");
```


Plot of f(x) = x - x^2 * sin(x)
60
40
20
0
−20
−40
−60
f(x)
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
x