

```
In [1]: print("19IT016: Mananv Butani")
```

19IT016: Mananv Butani

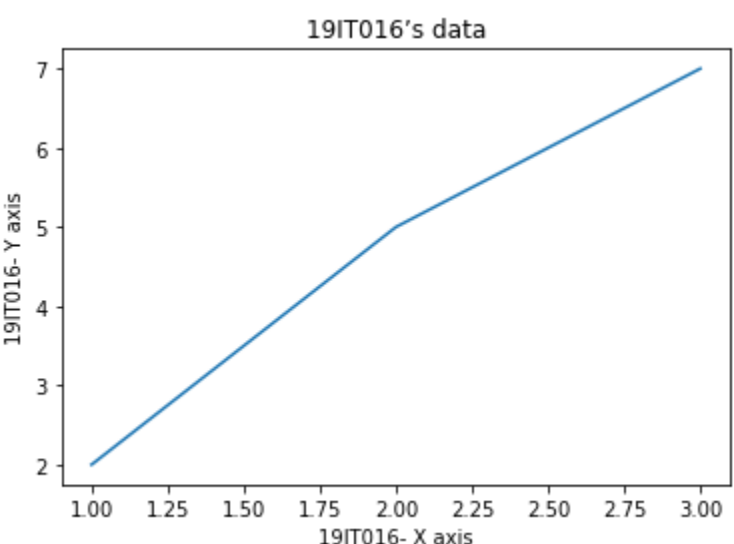
# Matplotlib

## Importing matplotlib

```
In [2]: import matplotlib as mt
from matplotlib import pyplot as plt
%matplotlib inline
```

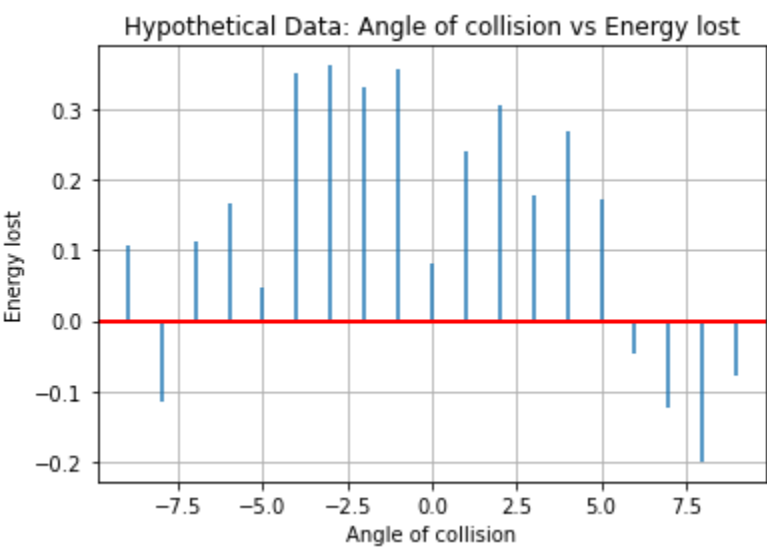
## Simple line chart

```
In [3]: plt.ylabel("19IT016- Y axis")
plt.xlabel("19IT016- X axis")
plt.title("19IT016's data")
plt.plot([1,2,3],[2,5,7])
plt.show()
```



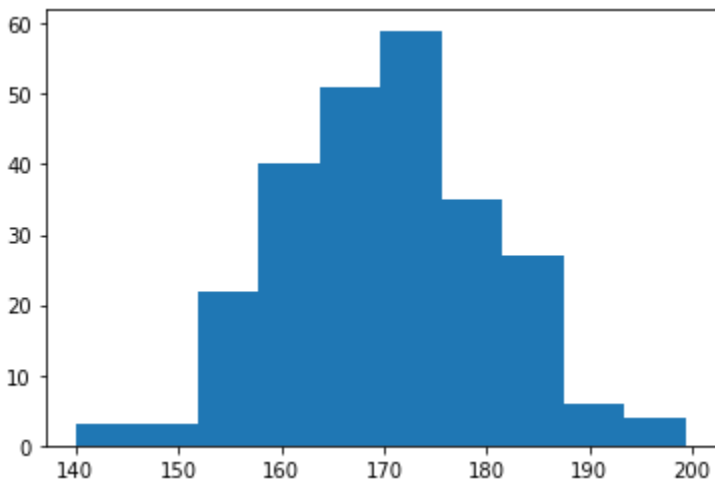
## Correlation chart

```
In [4]: import numpy as np
x_axis = np.array([24.40,10.25,20.05,22.00,16.90,7.80,15.00,22.80,34.90,13.30])
y_axis = np.array([-4.40,0.25,-0.05,2.00,6.90,-0.80,5.00,2.80,-4.90,3.30])
lines = plt.xcorr(x_axis, y_axis, maxlags=9, usevlines=True)
plt.title('Hypothetical Data: Angle of collision vs Energy lost')
plt.xlabel('Angle of collision')
plt.ylabel('Energy lost')
plt.grid(True)
plt.axhline(0, color='red', lw=2)
plt.show()
```



## Histogram

```
In [5]: import numpy as np
x = np.random.normal(170, 10, 250) # use NumPy to randomly generate an array with 250 values, where the values will concentrate around 170, and the
plt.hist(x)
plt.show()
```



## Plotting of Multivariate data

```
In [6]: import pandas as pd
import numpy as np
from IPython.display import HTML

# spin up a random dataframe of size 100
np.random.seed(13)
X = np.random.normal(50, 20, size=100)
Y = 2 * X + np.random.randint(25)
Z = np.random.choice(['pink', 'blue', 'green', 'red'], p=[0.1, 0.3, 0.4, 0.2], size=100)
df = pd.DataFrame({'X': X, 'Y': Y, 'Z': Z})
HTML(df.head().to_html(classes="table table-stripped table-hover table-dark"))
for size, color in zip([250, 80, 175, 30],[blue', 'green', 'pink', 'red']):
    color_index = df[df.Z == color].index
    color_array = np.random.normal(size, 10, size = len(color_index))
    for i in range(len(color_index)):
        df.at[color_index[i], 'A'] = color_array[i]
import matplotlib as mpl
plt.rcParams.update(mpl.rcParamsDefault)

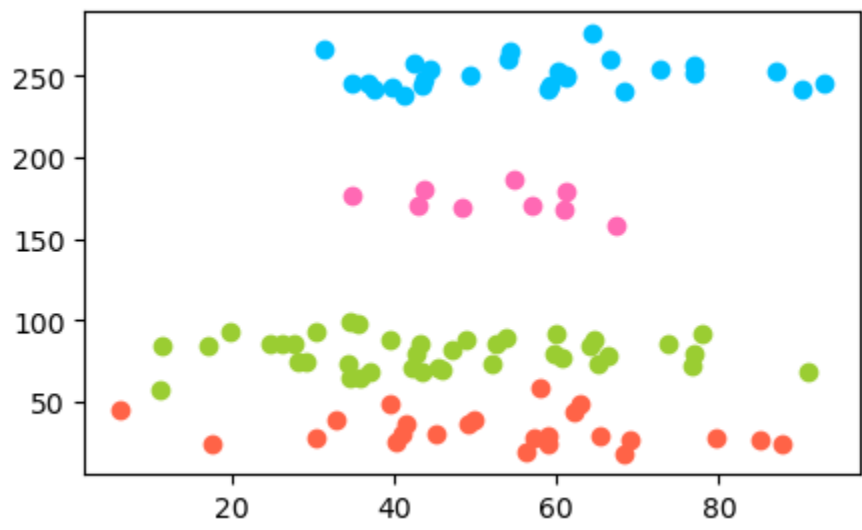
plt.figure(figsize= (5, 3))

plt.scatter(df[df['Z'] == 'green'].X, df[df['Z'] == 'green'].A
, color='yellowgreen');

plt.scatter(df[df['Z'] == 'blue'].X, df[df['Z'] == 'blue'].A
, color='deepskyblue');

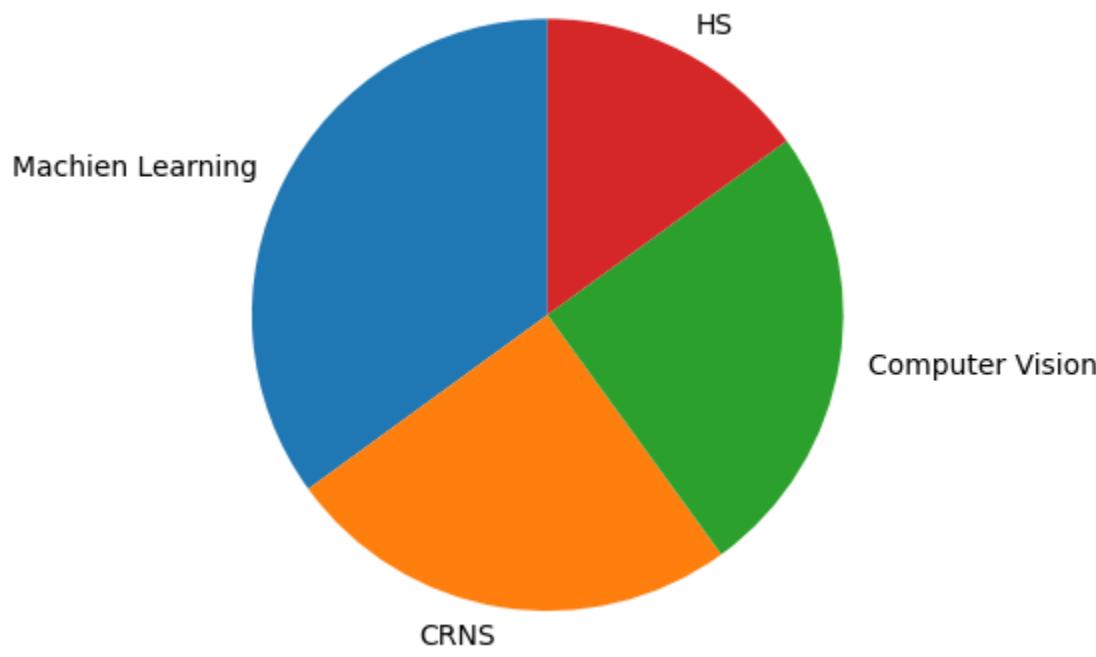
plt.scatter(df[df['Z'] == 'pink'].X, df[df['Z'] == 'pink'].A
, color='hotpink');

plt.scatter(df[df['Z'] == 'red'].X, df[df['Z'] == 'red'].A
, color='tomato');
```



## Plot Pi Chart

```
In [7]: marks = np.array([35, 25, 25, 15])
subjects = ["Machien Learning", "CRNS", "Computer Vision", "HS"]
plt.pie(marks, labels = subjects, startangle = 90)
plt.show()
```



```
In [ ]:
```

```
In [ ]:
```