

Intro to matplotlib

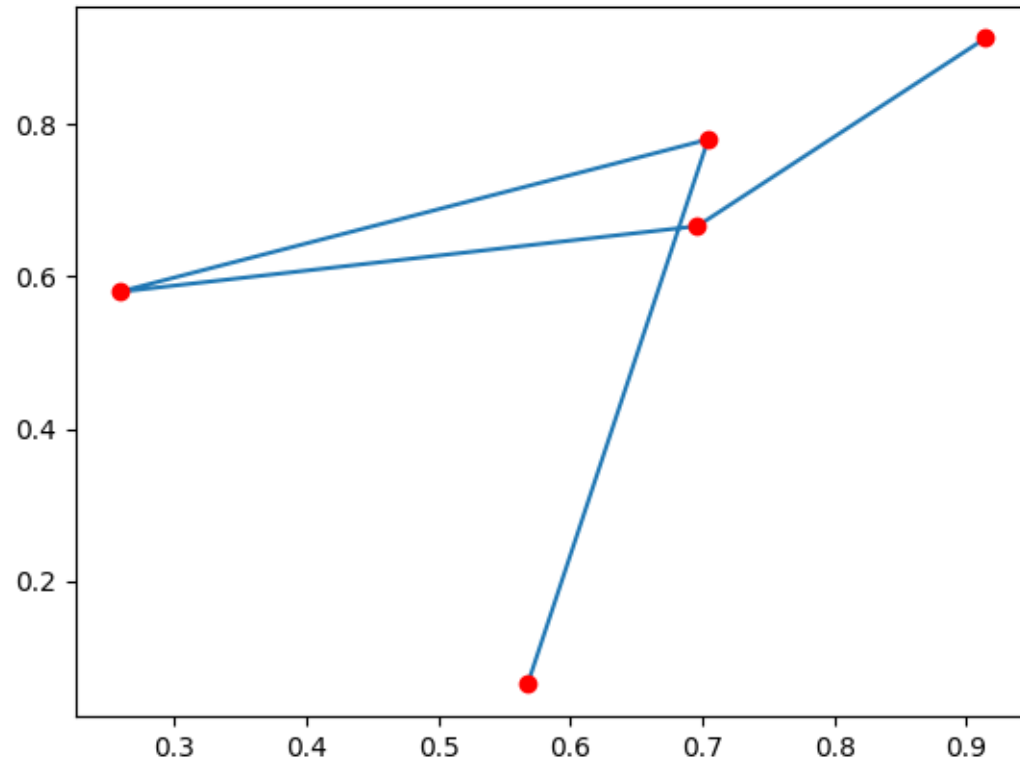
- matplotlib is “MATLAB (subset) in Python”
- Relies on **numpy** for vectors, math
- Relies on **tkinter** for display graphics
- Usage:

```
import matplotlib.pyplot as plt
```

Sample Program

```
import numpy as np
import matplotlib.pyplot as plt
fig = plt.figure()
ax = fig.add_subplot(111)
rand = np.random.random
xvals = rand(5); yvals = rand(5)
ax.plot(xvals, yvals)
ax.plot(xvals, yvals, 'ro',
        linestyle='None')
plt.pause(0.01)
```

Plotting Points



PlotDrive (1)

```
from cozmo_fsm import *
```

```
class PlotDrive(StateMachineProgram):
```

```
    class SetupPlot(StateNode):
```

```
        def start(self, event=None):
```

```
            super().start()
```

```
            self.parent.fig = plt.figure()
```

```
            self.parent.ax = ax = self.parent.fig.add_subplot(111)
```

```
            ax.set_xlim(-5,5)
```

```
            ax.set_ylim(23,29)
```

```
            ax.plot([-5,5],[25,25], 'g--')
```

```
            ax.plot([0,0],[23,29], 'r--')
```

```
            ax.set_xlabel('Drift')
```

```
            ax.set_ylabel('Travel Distance')
```

```
            self.parent.old_pos = self.robot.pose.position
```

PlotDrive (2)

```
class AddPoint(StateNode):  
    def start(self,event=None):  
        super().start()  
        fwd = self.robot.pose.position.x - self.parent.old_pos.x  
        drift = self.robot.pose.position.y - self.parent.old_pos.y  
        print('fwd=',fwd,'drift=',drift)  
        self.parent.ax.plot(drift,fwd,'bo',linestyle='None')  
        self.parent.old_pos = self.robot.pose.position
```

```
class ShowPlot(StateNode):  
    def start(self,event=None):  
        super().start()  
        plt.pause(0.01)
```

PlotDrive (3)

```
$setup{  
    self.SetupPlot() =N=> loop  
  
    loop: Iterate(range(5))  
    loop =D=> Forward(25) =C=> self.AddPoint() =Next=> loop  
    loop =C=> show  
  
    show: self.ShowPlot()  
  
}
```

PlotDrive

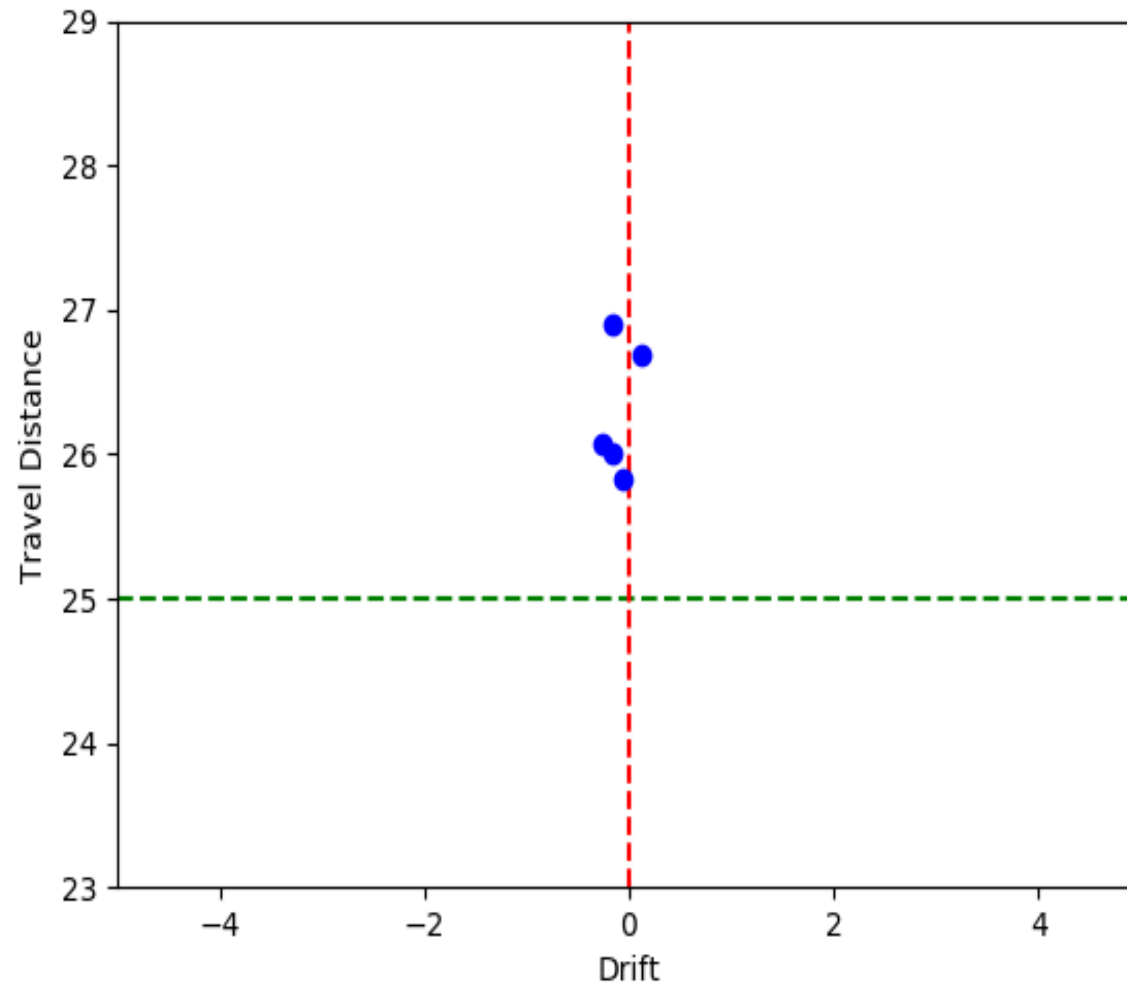


Image Display

```
rimg = robot.world.latest_image.raw_image
```

```
img = np.array(rimg)
```

```
plt.imshow(img)
```

Intensity Histogram

```
pixels = tuple(img[:, :, 1].flat)
```

```
plt.hist(pixels, bins=25)
```

```
plt.pause(0.01)
```

