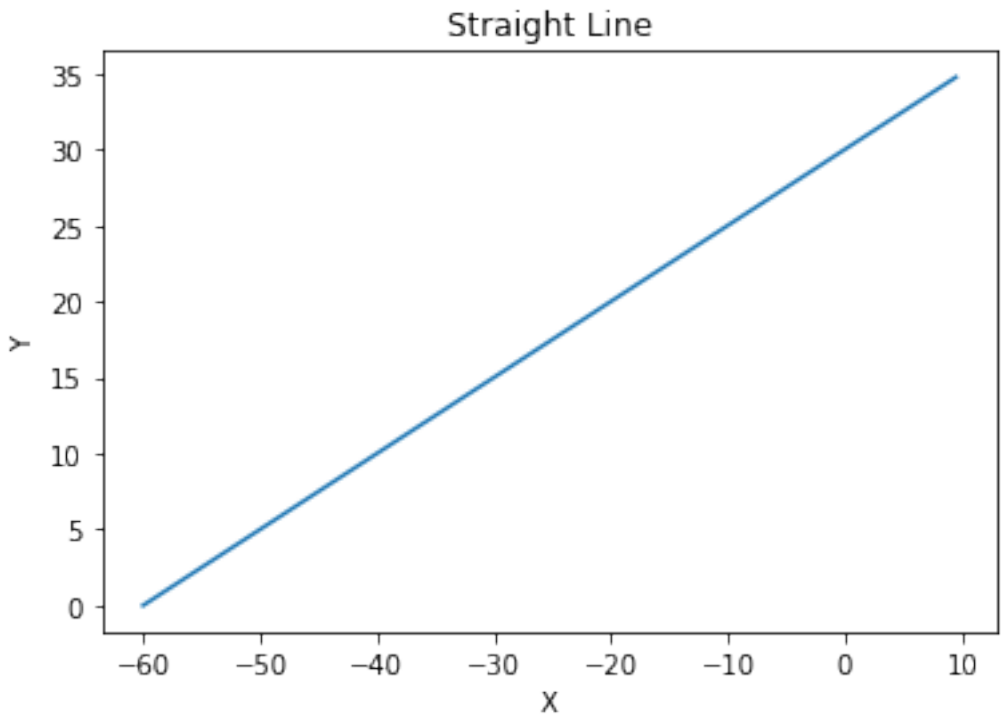
Homework1

January 17, 2019

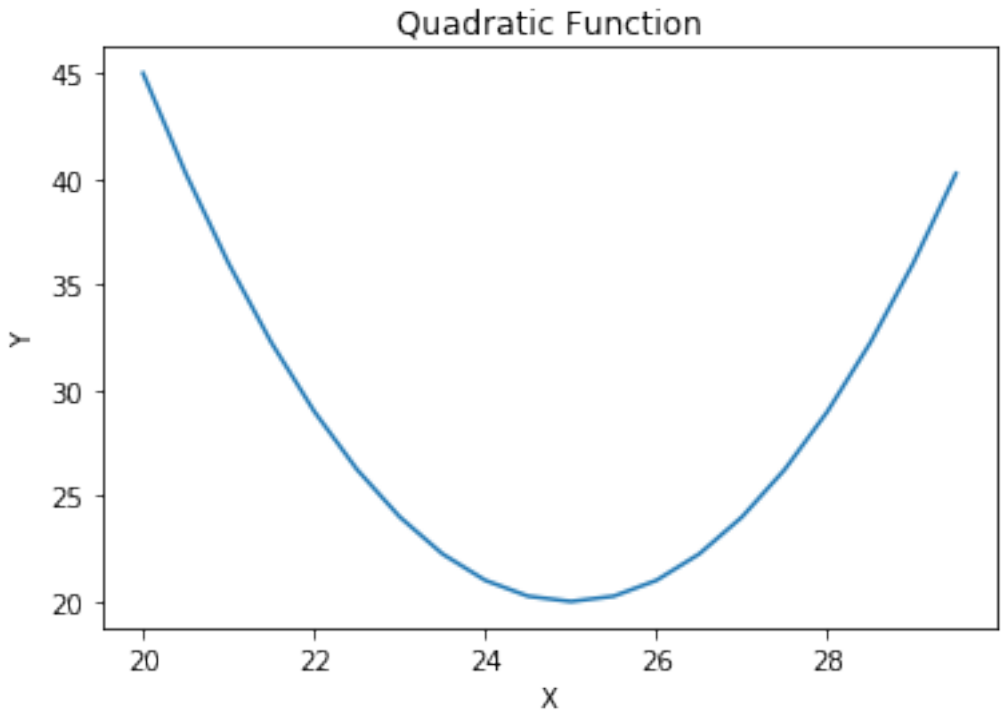
Ross Lewis Machine Learning Homework 1

1 Question 1

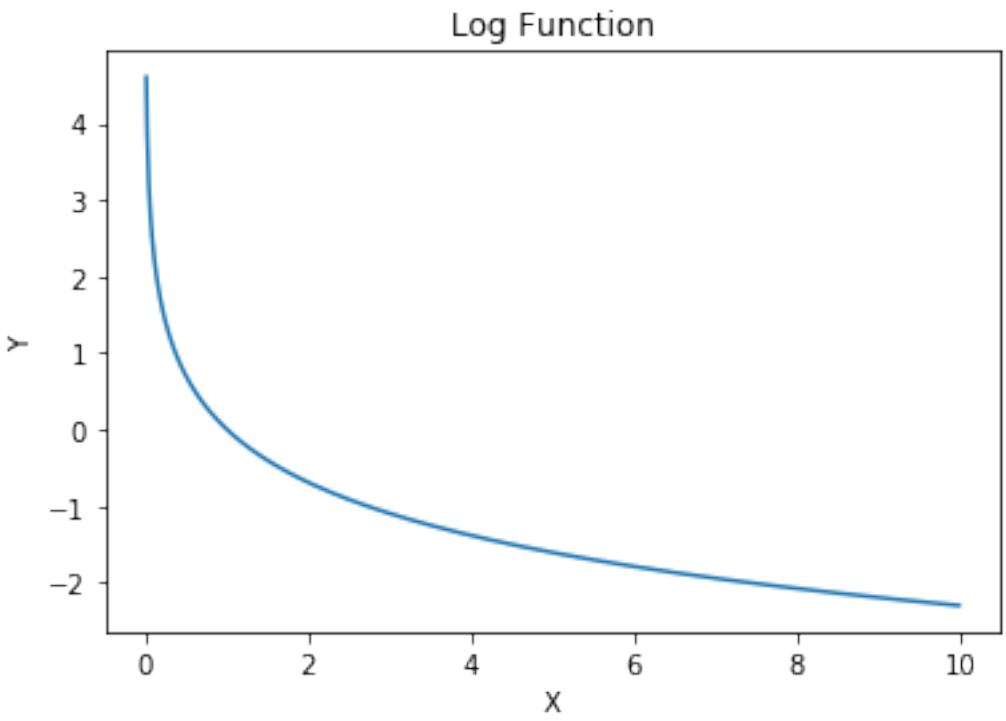
In [28]:
```
%matplotlib inline

import numpy as np
import matplotlib.pyplot as plt
from matplotlib import pylab

x = np.arange(-60,10,.5)
b = 30
m = .5
y = m*x + b

fig, ax = plt.subplots()

ax.plot(x,y)
ax.set_title("Straight Line")
ax.set_xlabel("X")
ax.set_ylabel("Y")

plt.show()
```



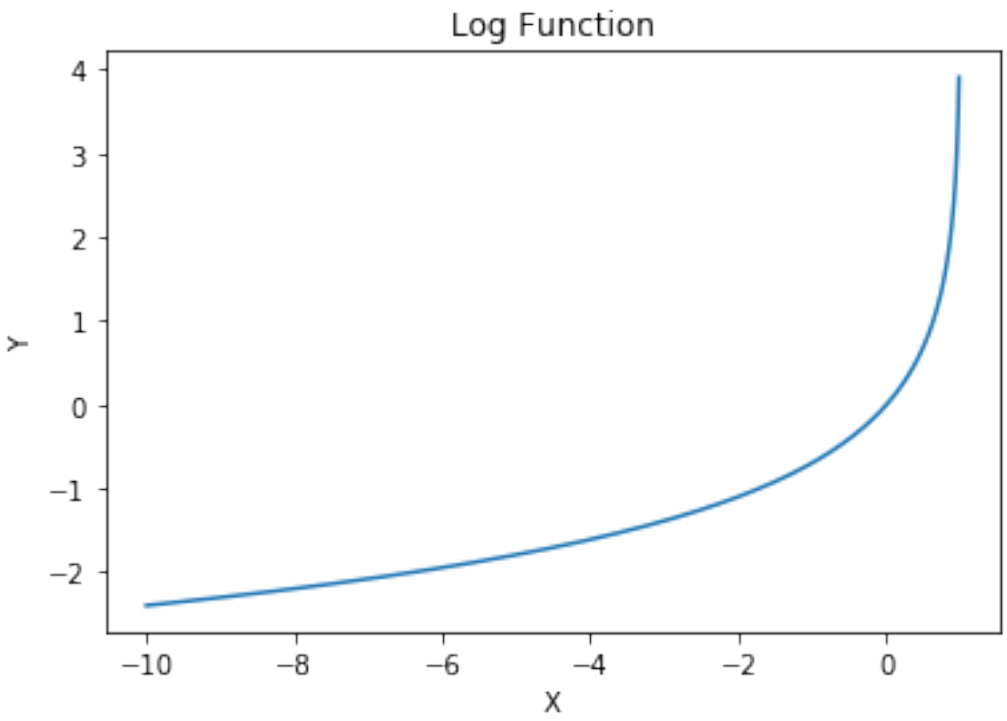
In [33]:
```
x = np.arange(20,30,.5)
t1 = 25
t2 = 20
y = (x - t1)**2 + t2

fig, ax = plt.subplots()

ax.plot(x,y)
ax.set_title("Quadratic Function")
ax.set_xlabel("X")
ax.set_ylabel("Y")

plt.show()
```



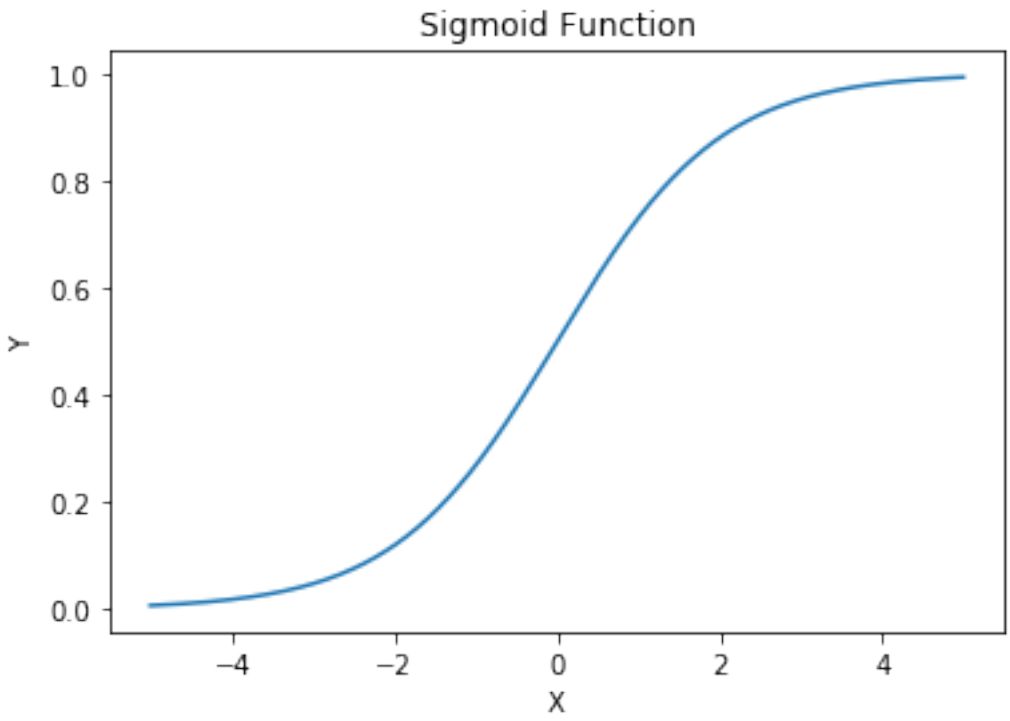

```
In [49]: import math
         x = np.arange(.01,10,.01)
         y1 = [-math.log(val) for val in x]

         fig, ax = plt.subplots()

         ax.plot(x,y1)
         ax.set_title("Log Function")
         ax.set_xlabel("X")
         ax.set_ylabel("Y")

         plt.show()
```




```
In [50]: x = np.arange(-10,.99,.01)
         y2 = [-math.log(1-val) for val in x]
         fig, ax = plt.subplots()

         ax.plot(x,y2)
         ax.set_title("Log Function")
         ax.set_xlabel("X")
         ax.set_ylabel("Y")

         plt.show()
```




```
In [55]: x = np.arange(-5,5,.01)
         y = 1/(1+math.e**(-x))
         fig, ax = plt.subplots()

         ax.plot(x,y)
         ax.set_title("Sigmoid Function")
         ax.set_xlabel("X")
         ax.set_ylabel("Y")

         plt.show()
```



2 Question 2

i - labeled data (0,1) ii - labeled data (0,1) iii - unlabeled data

b is correct

i) supervised learning with discrete predictions; ii) supervised learning with discrete predictions; iii) unsupervised learning with discrete results

3 Question 3

When I was working at Talisman Systems Group, we had a process to verify doctors' and providers' addresses. It would be beneficial to reduce the number of calls the call center would need to make if there was a good chance that the doctor didn't move. We could implement a classification model to predict whether or not the provider is still at that location. Our training data would consist of features of that provider: Specialty, zip code, state, city, number of other providers at that location, time at current location. The label ("has moved") can be filled in for past providers that were at a location and are no longer practicing there. If we had this model, we could choose to only call providers who were likely to no longer be proacticing at the location in our data.

- Collect the dataset: It already exists in the Talisman system

- Traing/validation/testing data: This can all be retreived from the historic data
- Ground Truth: For any doctor or provider in which we have a verified history, we can label the last call to that doctor as "has moved" = True. All other calls will be labeled as "has moved" = False.