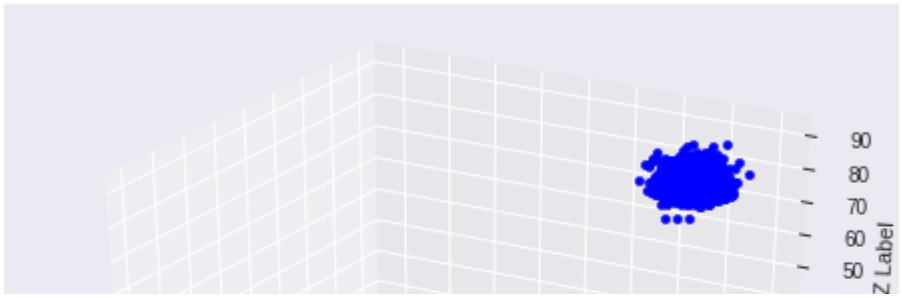
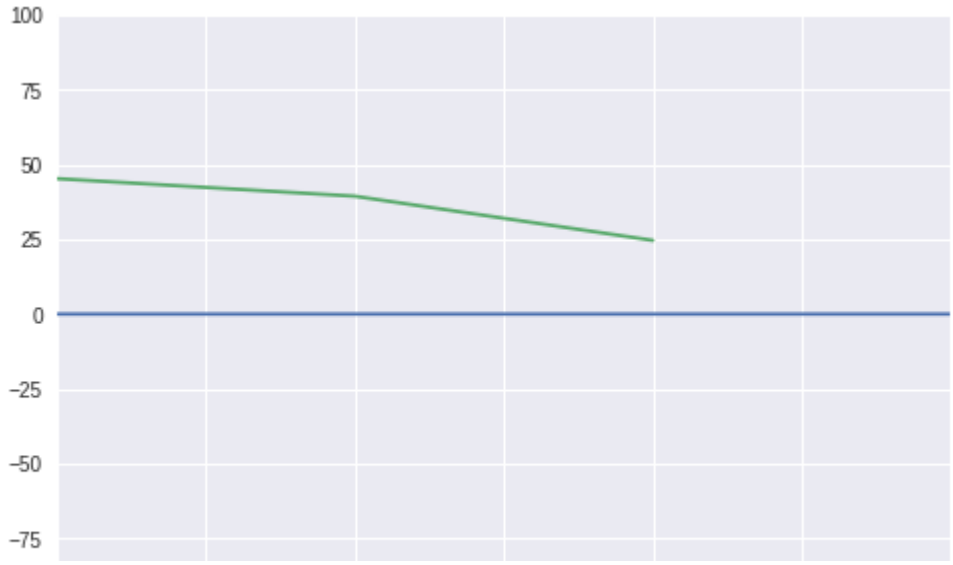
```
#Programmer 1: Akhileshwar Bomma (CWID: 893581496)
#Programmer 2: Roberto Perez Mendoza (CWID: 892037052)
#Project 1
#CPSC585

import numpy as np

class_0 = []
n=1000
for i in range(n):
  for j in range(3):
    a=np.random.normal(20,3,3)
    a=np.append(a,[0])
    #print(a)
  class_0.append(a)

class_1 = []
for i in range(n):
  for j in range(3):
    a=np.random.normal(80,3,3)
    a=np.append(a,[1])
    #print(a)
  class_1.append(a)



from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt

fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

for i in class_0:
    x = i[0]
    y = i[1]
    z = i[2]
    ax.scatter(x, y, z, c="red")

ax.set_xlabel('X Label')
ax.set_ylabel('Y Label')
ax.set_zlabel('Z Label')


for i in class_1:
    x = i[0]
    y = i[1]
    z = i[2]
    ax.scatter(x, y, z, c="blue")

ax.set_xlabel('X Label')
ax.set_ylabel('Y Label')
ax.set_zlabel('Z Label')
plt.show()
```




```
import random
import numpy as np
import matplotlib
import matplotlib.pyplot as plt

random.shuffle(class_0)
random.shuffle(class_1)

total_set = class_0 + class_1
random.shuffle(total_set)
#20% is the validation set
k = 2*n//5
validation_set = total_set[0:k]
#print(len(validation))
training_set = total_set[k:]
#print(len(training_set))

w = np.zeros(4)
eta = 0.1
epochs = 3
error_all = np.zeros(0)
error_validation = np.zeros(0)

def prediction(x):
    prediction = np.dot(w[1:],x[0:3]) + w[0]
    if prediction > 0:
      return 1
    return 0

for e in range(epochs):
  error_count_training = 0
  for x in training_set:
    y = prediction(x)
    if y != x[3]:
      error_count_training += 1
    w[1:] = w[1:] + eta*(x[3] - y)* x[0:3]
    w[0] =  w[0] + eta*(x[3] - y)
  error_all = np.append(error_all, error_count_training/len(training_set)*100)
  print(error_count_training/len(training_set)*100)
  error_count_validation = 0
  for v in validation_set:
    y_valid = prediction(v)
    if y_valid != v[3]:
      error_count_validation += 1
    error_validation = np.append(error_validation, error_count_validation/len(validation_set)

#fig, ax = plt.subplots()
plt.plot(error_validation)
plt.xlim(0, epochs)
plt.ylim(-100, 100);
plt.plot(error_all)
plt.show()
```

```
45.25
39.375
24.625
```




```
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt

fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

for i in class_0:
    x = i[0]
    y = i[1]
    z = i[2]
    ax.scatter(x, y, z, c="red")

ax.set_xlabel('X Label')
ax.set_ylabel('Y Label')
ax.set_zlabel('Z Label')


for i in class_1:
    x = i[0]
    y = i[1]
    z = i[2]
    ax.scatter(x, y, z, c="blue")

#ax.set_xlabel('X Label')
#ax.set_ylabel('Y Label')
#ax.set_zlabel('Z Label')

x = np.linspace(-20, 100, 10)
y = np.linspace(-20, 100, 10)
X, Y = np.meshgrid(x,y)
#Z = -w[0]/w[3] - w[1]/w[3]*X - w[2]/w[3]*Y
Z = -w[1]/w[3]*X - w[2]/w[3]*Y - w[0]*w[3]

#ax.plot_surface(X, Y, Z, rstride=1, cstride=1,cmap='viridis', edgecolor='none')
#ax.set_title('surface');
#ax.plot_wireframe(X, Y, Z, color='black')
surf = ax.plot_surface(X, Y, Z, antialiased=False, alpha=0.75)

ax.legend()

plt.show()
```

```
No handles with labels found to put in legend.
```