Artificial Intelligence Lab Work (2)

レポート解答用紙（Report Answer Sheet）

学生証番号（Student ID): 18521071

名前(Name): ズオン・ミン・ルオン（Duong Minh Luong）

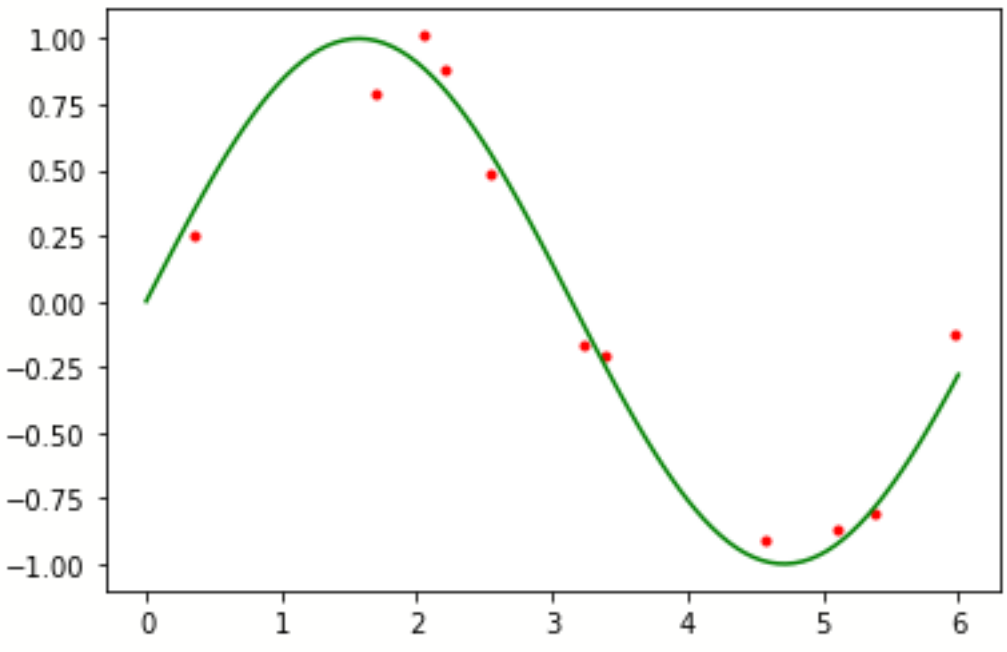
```
#Data D
X = [0.349526784, 1.6974435, 5.384308891, 2.044150596,
     4.578814506, 3.241690807, 2.535931731, 2.210580888,
     3.397474351, 5.972933146, 5.114704101]

Y = [0.254020646, 0.790556868, -0.81239532, 1.012143475,
     -0.904558188, -0.167456361, 0.482547054, 0.878514378,
     -0.210093715, -0.128786937, -0.866501299]
```

問 1.

(プログラム)

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

plt.scatter(X,Y,color = 'red', marker = '.')

def f_sin(x):
  return np.sin(x)

X_new = np.linspace(0, 6, 100)
Y_new = f_sin(X_new)

plt.plot(X_new, Y_new, color = 'green')
plt.show()
```

(グラフ)



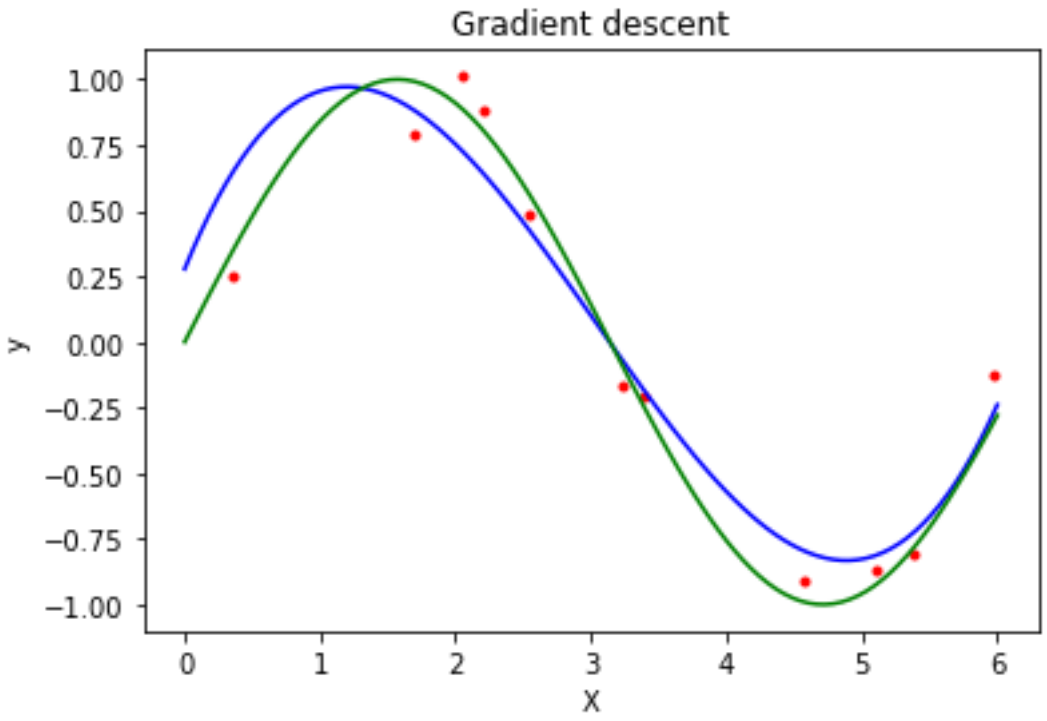
問 2.

(プログラム)

```
#初期化
epoch = 200000
lr = 0.000008
a = 0 ,b = 0 ,c = 0 ,d = 0
A = [a]
B = [b]
C = [c]
D = [d]
#a、b、c、d の最急降下法を計算します
for e in range(epoch):
  grad_a = 0
  grad_b = 0
  grad_c = 0
  grad_d = 0
  loss = 0

  for i in range(len(X)):
    x = X[i]
    y = Y[i]

    grad_a = grad_a - (y - a*x**3 - b*x**2 - c*x - d)*2*x**3
    grad_b = grad_b - (y - a*x**3 - b*x**2 - c*x - d)*2*x**2
    grad_c = grad_c - (y - a*x**3 - b*x**2 - c*x - d)*2*x
    grad_d = grad_d - (y - a*x**3 - b*x**2 - c*x - d)*2
    loss = loss + (y - a*x**3 - b*x**2 - c*x - d)**2


  a = a - lr*grad_a
  b = b - lr*grad_b
  c = c - lr*grad_c
  d = d - lr*grad_d

  print("epoch: {}, a: {}, b: {}, c: {}, d: {}, loss: {}".format(e, a, b, c
, d, loss ))

plt.scatter(X,Y,color = 'red', marker = '.')
```

```
def cubic_funtion(x):
  return a*x**3 + b*x**2 + c*x + d
X_new1 = np.linspace(0,6,100)
Y_new1 = cubic_funtion(X_new1)
plt.plot(X_new1, Y_new1, color = 'blue')

def f_sin(x):
  return np.sin(x)

X_new = np.linspace(0, 6, 100)
Y_new = f_sin(X_new)
plt.xlabel('X')
plt.ylabel('y')
plt.title('Gradient descent')
plt.plot(X_new, Y_new, color = 'green')
plt.show()
```

(グラフ)

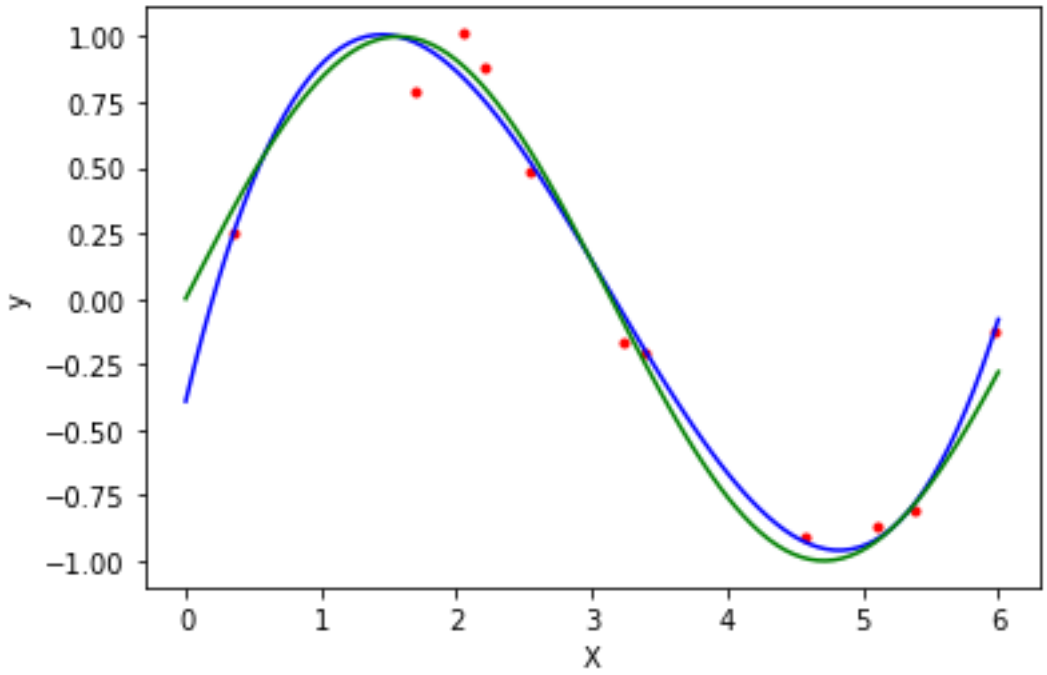


問 3. (a)

(プログラム)

```
X3 = []

for x in X:
  t = []
  for i in range(4):
    t.append(x**i)
  X3 = X3 + [t]

X3 = np.array(X3)
Y3 = np.array([Y]).T
Z1 = np.matmul(X3.T,X3)
Z2 = np.linalg.inv(Z1)
Z3 = np.matmul(Z2,X3.T)
w = np.matmul(Z3, Y3)
print(w)
plt.scatter(X,Y,color = 'red', marker = '.')

def cubic_funtion(x):
  return w[3][0]*x**3 + w[2][0]*x**2 + w[1][0]*x +w[0][0]

X_new1 = np.linspace(0,6,100)
Y_new1 = cubic_funtion(X_new1)

plt.plot(X_new1, Y_new1, color = 'blue')

def f_sin(x):
  return np.sin(x)

X_new = np.linspace(0, 6, 100)
Y_new = f_sin(X_new)
plt.xlabel('X')
plt.ylabel('y')
plt.plot(X_new, Y_new, color = 'green')
plt.show()
```

（グラフ）



問 3. (b)

(プログラム)

```
X9 = []
for x in X:
  t = []
  for i in range(10):
    t.append(x**i)
  X9 = X9 + [t]

X9 = np.array(X9)
Y9 = np.array([Y]).T
Z1 = np.matmul(X9.T,X9)
Z2 = np.linalg.inv(Z1)
Z3 = np.matmul(Z2,X9.T)
w = np.matmul(Z3, Y9)
print(w)
plt.scatter(X,Y,color = 'red', marker = '.')
def ninth_funtion(x):
  t = 0
  for i in range(10):
    t = t + w[i][0]*x**i
  return t

X_new1 = np.linspace(0,6,100)
Y_new1 = ninth_funtion(X_new1)
plt.plot(X_new1, Y_new1, color = 'blue')

def f_sin(x):
  return np.sin(x)

X_new = np.linspace(0, 6, 100)
Y_new = f_sin(X_new)
plt.xlabel('X')
plt.ylabel('y')
plt.plot(X_new, Y_new, color = 'green')
plt.show()
```

（グラフ）