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)  Ảnh có chứa văn bản, thiết bị điện tử  Mô tả được tạo tự động |