Q1. Train a **single perceptron and SVM** to learn an AND gate with two inputs x1 and x2. Assume that all the weights of the perceptron are initialized as 0. Show the calulation for each step and also draw the decision boundary for each updation.

Importing the necessary libraries

In [ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from cvxopt import matrix, solvers
```

Single Perceptron

Defining the necessary functions used

In [ ]:
```
def decision_plot(w, x_min, x_max, y_min, y_max):
    inp_0 = []
    inp_1 = []

    x_min = x_min - np.abs(((x_max - x_min) / 10))
    y_min = y_min - np.abs(((y_max - y_min) / 10))

    x_max = x_max + np.abs(((x_max - x_min) / 10))
    y_max = y_max + np.abs(((y_max - y_min) / 10))

    for i in np.arange(x_min, x_max, np.abs(x_max - x_min) / 100):
        for j in np.arange(y_min, y_max, np.abs(x_max - x_min) / 100):
            y_pred = np.dot(np.array([[1, i, j]]), w)

            if y_pred < 0:
                inp_0.append(np.array((i, j)))

            else:
                inp_1.append(np.array((i, j)))

    return inp_0, inp_1
```

In [ ]:
```
def perceptron_training_alg(X, Y, learning_rate):
  # Adding Bias to the input and Randomly initializing weights
  num_samples, num_features = X.shape

  X = np.hstack((np.ones((num_samples, 1)), X))
  W = np.zeros(num_features + 1).reshape((-1, 1))

  weight_vector = []
  weight_vector.append(W)

  print(W, "\n")

  # Running the algorithm until all the data points are correctly classified
  samples_misclassified = True
  cnt=0
  while samples_misclassified and cnt < 100000:
    # Initially we'll assume that there are no misclassified samples
    samples_misclassified = False

    for index, x in enumerate(X):
      if cnt >= 100000:
        return W, np.array(weight_vector)
```

```
      x = Y[index] * x

      if np.dot(x.reshape((1, -1)), W) <= 0: # if samples misclassified
        samples_misclassified = True
        W = W + (learning_rate * x.reshape((-1,1)))     # Gradient Descent Step
        weight_vector.append(W)

        print(W, "\n")
        cnt += 1

  return W, np.array(weight_vector)
```

In [ ]:

```
def sorter(X, Y):
  X_new0 = []
  Y_new0 = []

  X_new1 = []
  Y_new1 = []

  for x, y in zip(X, Y):
    if y == 0:
      Y_new0.append(y)
      X_new0.append(x)

    else:
      Y_new1.append(y)
      X_new1.append(x)

  X0 = []
  Y0 = []

  X1 = []
  Y1 = []

  if len(X_new0) > 0:
    X0 = np.stack(X_new0)
    Y0 = np.stack(Y_new0)

  elif len(X_new1) > 0:
    X1 = np.stack(X_new1)
    Y1 = np.stack(Y_new1)

  if len(X0) > 0 and len(X1) > 0:
    X = np.concatenate((X0, X1))
    Y = np.concatenate((Y0, Y1))

  elif len(X0) > 0:
    X = X0
    Y = Y0

  else:
    X = X1
    Y = Y1

  return X, Y
```

In [ ]:

```
def decision_boundary(W, X, Y):
  X, Y = sorter(X, Y)
  num_samples, num_features = X.shape
  np2 = np.hstack((np.ones((num_samples, 1)), X))

  X=np2

  x1, y1 = decision_plot(W, min(X[:,1]), max(X[:,1]), min(X[:,2]), max(X[:,2]))

  plt.scatter([i[0] for i in x1], [i[1] for i in x1])
```

```
    plt.scatter([i[0] for i in y1], [i[1] for i in y1])
    plt.scatter(X[:np.argmax(Y), 1], X[:np.argmax(Y), 2])
    plt.scatter(X[np.argmax(Y):, 1], X[np.argmax(Y):, 2])
    plt.show()
```

Code

In [ ]:

```
X = np.array([[1, 1], [0, 1], [1, 0], [0, 0]])
Y = np.array([[1], [-1], [-1], [-1]])
ans = np.array([[1], [0], [0], [0]])

len(X)
```

Out[ ]: 4

In [ ]:

```
W, weight_vector = perceptron_training_alg(X, Y, 0.5)
```

```
[[0.]
 [0.]
 [0.]]

[[0.5]
 [0.5]
 [0.5]]

[[0. ]
 [0.5]
 [0. ]]

[[-0.5]
 [ 0. ]
 [ 0. ]]

[[0. ]
 [0.5]
 [0.5]]

[[-0.5]
 [ 0.5]
 [ 0. ]]

[[-1.]
 [ 0.]
 [ 0.]]

[[-0.5]
 [ 0.5]
 [ 0.5]]

[[-1. ]
 [ 0.5]
 [ 0. ]]

[[-0.5]
 [ 1. ]
 [ 0.5]]

[[-1.]
 [ 1.]
 [ 0.]]

[[-1.5]
 [ 0.5]
 [ 0. ]]

[[-1. ]
 [ 1. ]
 [ 0.5]]

[[-1.5]
```

```
 [ 0.5]
 [ 0.5]]

[[-1.]
 [ 1.]
 [ 1.]]

[[-1.5]
 [ 1. ]
 [ 0.5]]

[[-1. ]
 [ 1.5]
 [ 1. ]]

[[-1.5]
 [ 1.5]
 [ 0.5]]

[[-2. ]
 [ 1. ]
 [ 0.5]]

[[-1.5]
 [ 1.5]
 [ 1. ]]

[[-2.]
 [ 1.]
 [ 1.]]

[[-1.5]
 [ 1.5]
 [ 1.5]]

[[-2. ]
 [ 1.5]
 [ 1. ]]
```

In [ ]:

```
W
```

Out[ ]:

```
array([[-2. ],
       [ 1.5],
       [ 1. ]])
```

In [ ]:

```
weight_vector
```

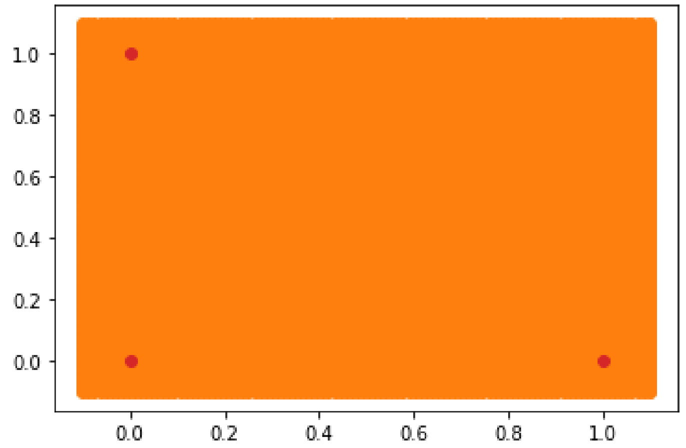
Out[ ]:

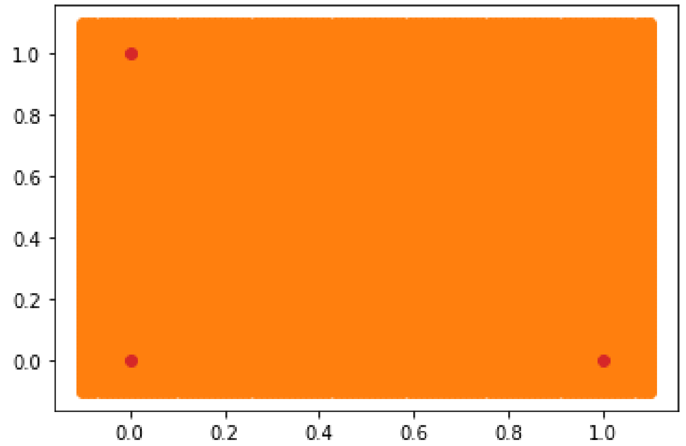
```
array([[[ 0. ],
        [ 0. ],
        [ 0. ]],

       [[ 0.5],
        [ 0.5],
        [ 0.5]],

       [[ 0. ],
        [ 0.5],
        [ 0. ]],

       [[-0.5],
        [ 0. ],
        [ 0. ]],

       [[ 0. ],
        [ 0.5],
        [ 0.5]],

       [[-0.5],
        [ 0.5],
        [ 0. ]],
```

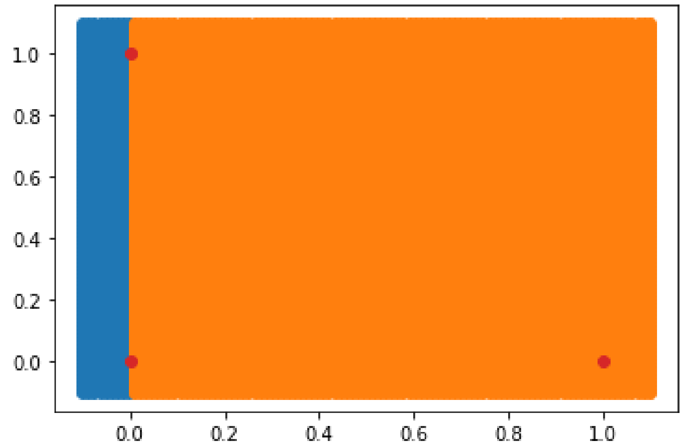
```
        [[-1. ],
         [ 0. ],
         [ 0. ]],

        [[-0.5],
         [ 0.5],
         [ 0.5]],

        [[-1. ],
         [ 0.5],
         [ 0. ]],

        [[-0.5],
         [ 1. ],
         [ 0.5]],

        [[-1. ],
         [ 1. ],
         [ 0. ]],

        [[-1.5],
         [ 0.5],
         [ 0. ]],

        [[-1. ],
         [ 1. ],
         [ 0.5]],

        [[-1.5],
         [ 0.5],
         [ 0.5]],

        [[-1. ],
         [ 1. ],
         [ 1. ]],

        [[-1.5],
         [ 1. ],
         [ 0.5]],

        [[-1. ],
         [ 1.5],
         [ 1. ]],

        [[-1.5],
         [ 1.5],
         [ 0.5]],

        [[-2. ],
         [ 1. ],
         [ 0.5]],

        [[-1.5],
         [ 1.5],
         [ 1. ]],

        [[-2. ],
         [ 1. ],
         [ 1. ]],

        [[-1.5],
         [ 1.5],
         [ 1.5]],

        [[-2. ],
         [ 1.5],
         [ 1. ]]]])
```

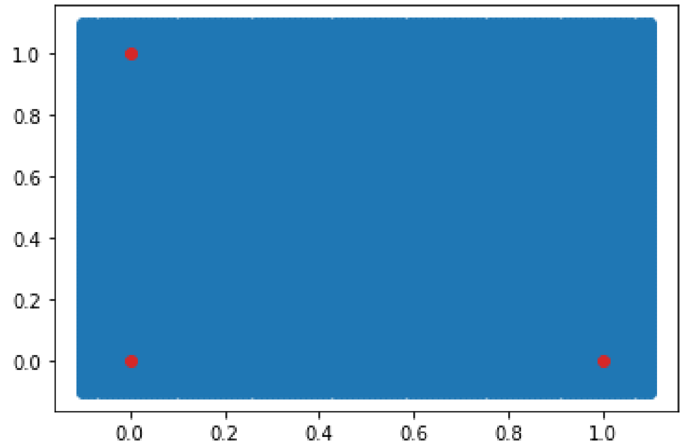
In [ ]:

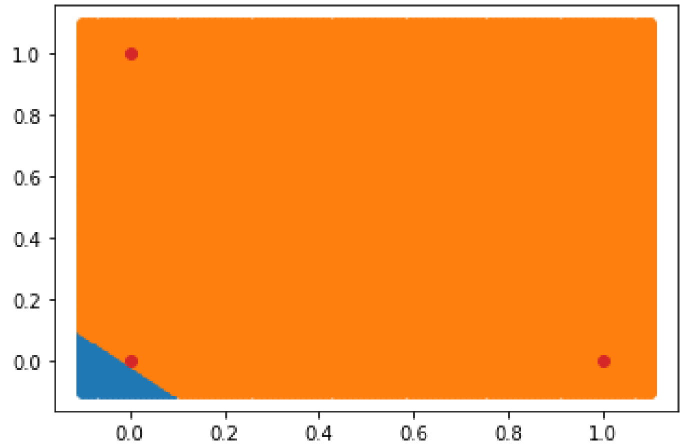
```
for w in weight_vector:
  decision_boundary(w, X, ans)
```

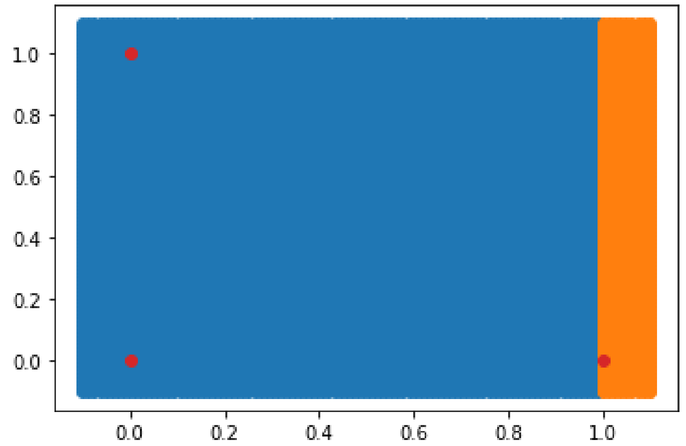

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

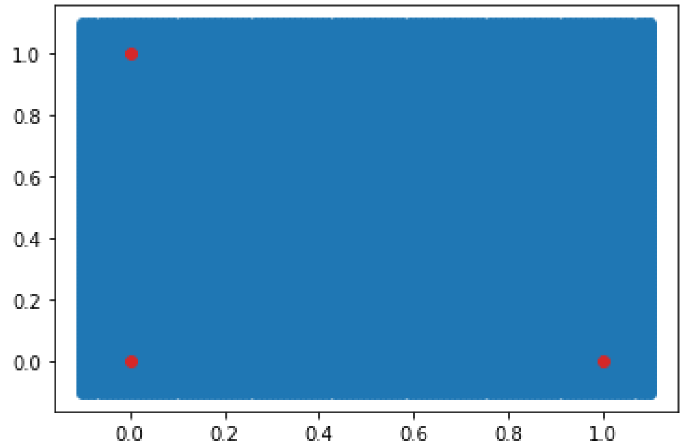

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

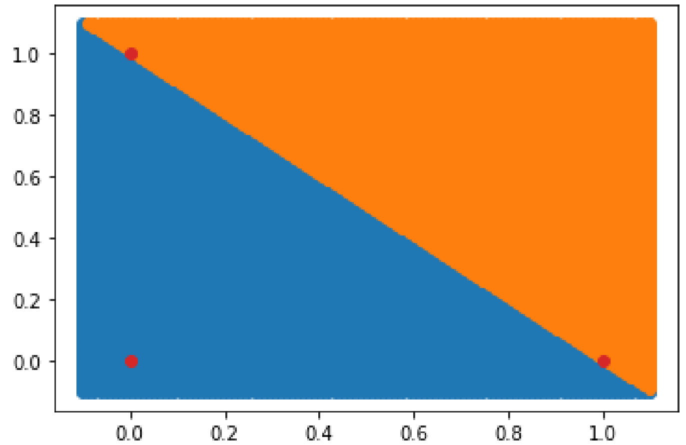

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

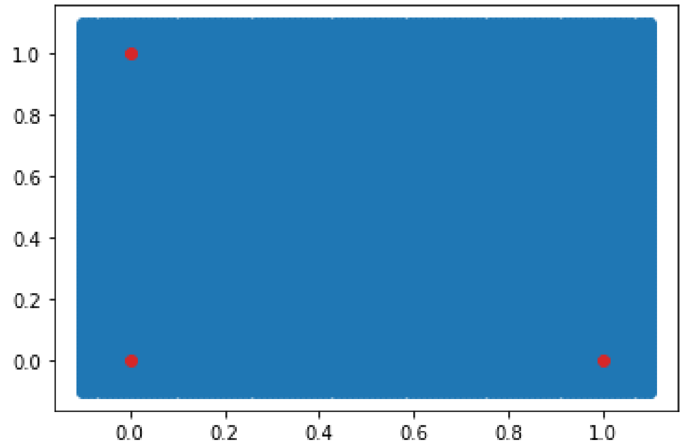

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

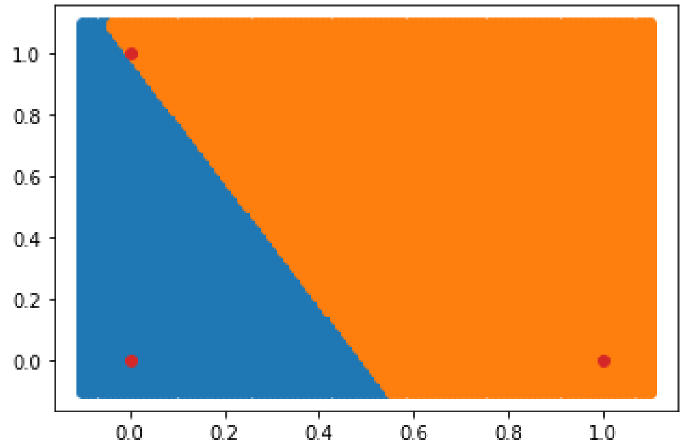

1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0

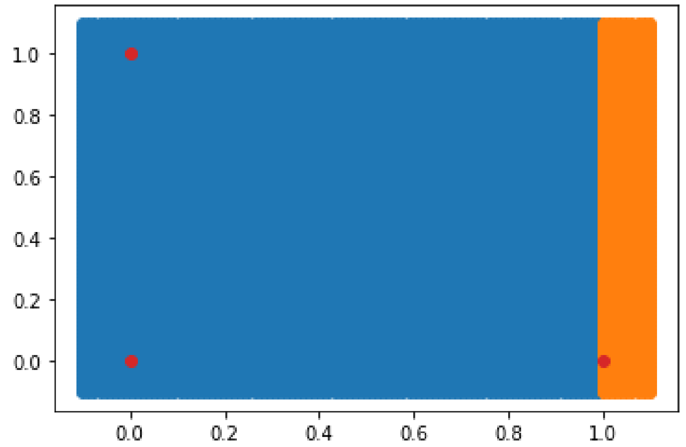

1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0

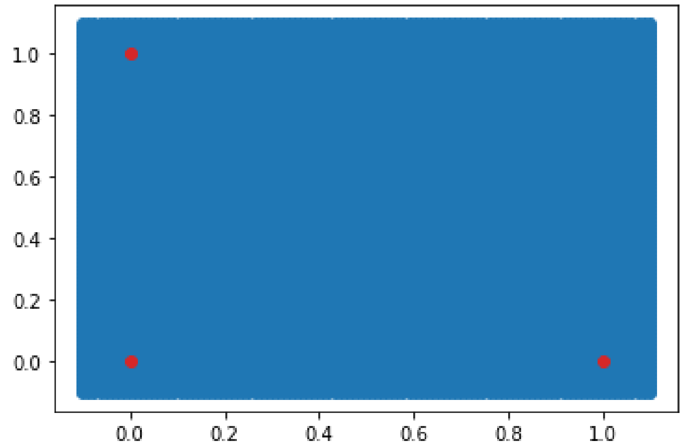

1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0

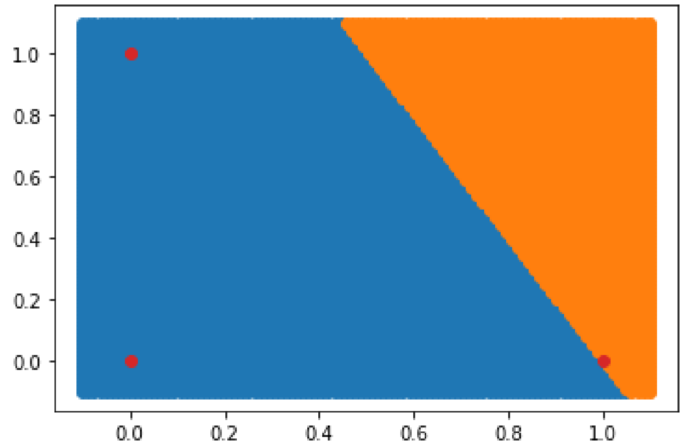

1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0

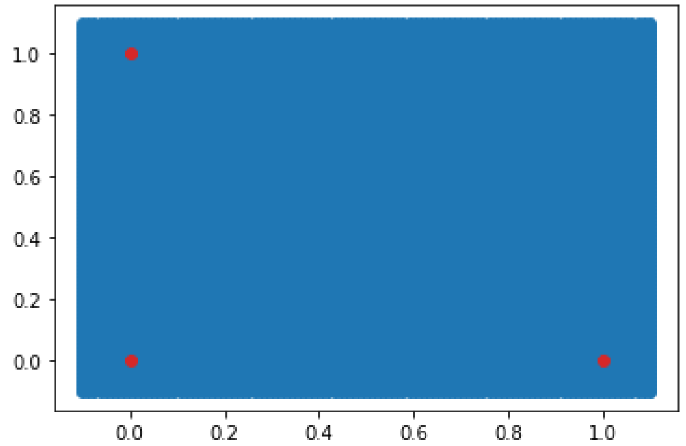

1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0

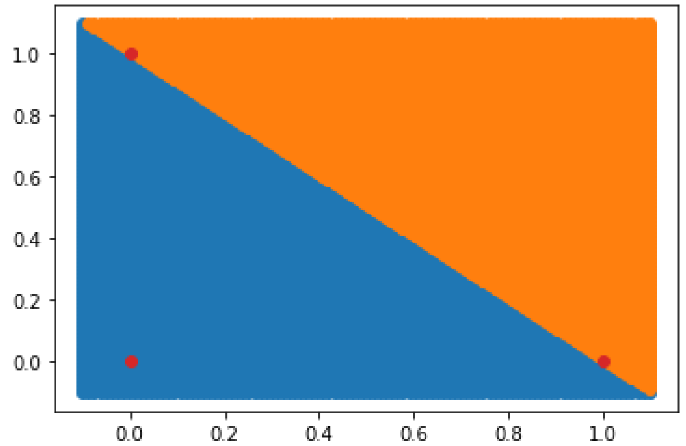

1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0

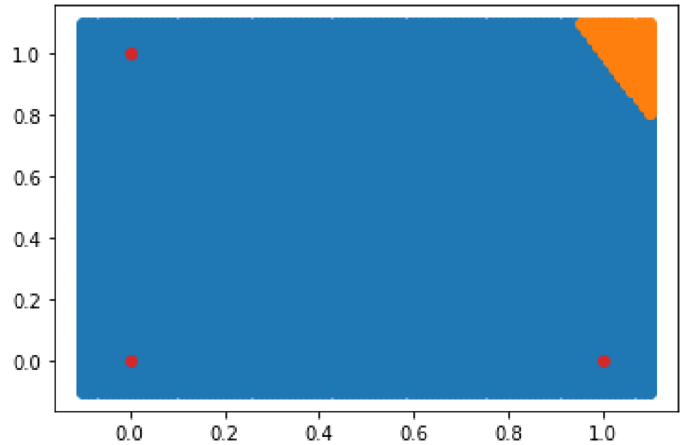

1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0

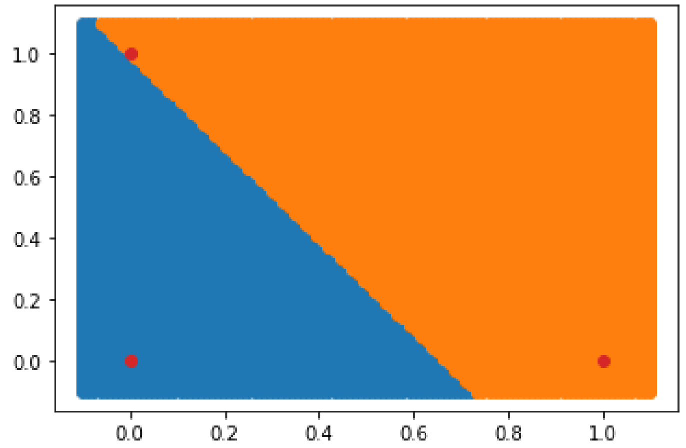

1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0

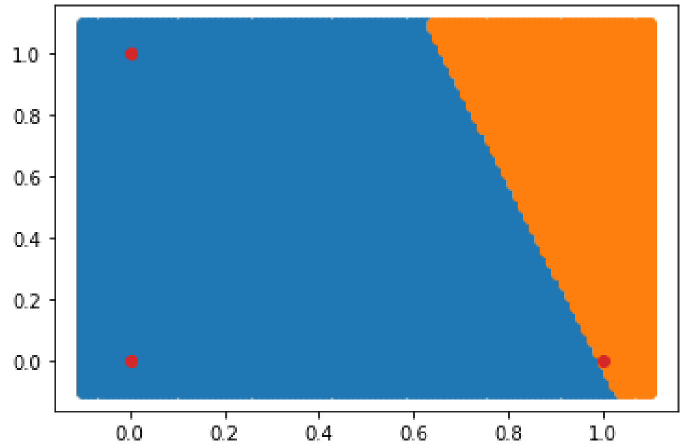

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

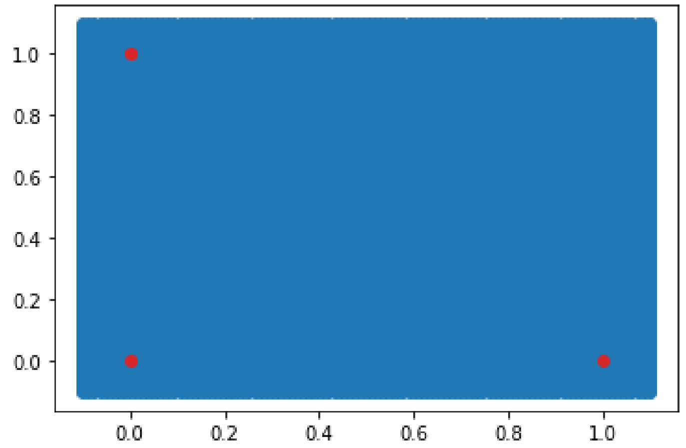

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

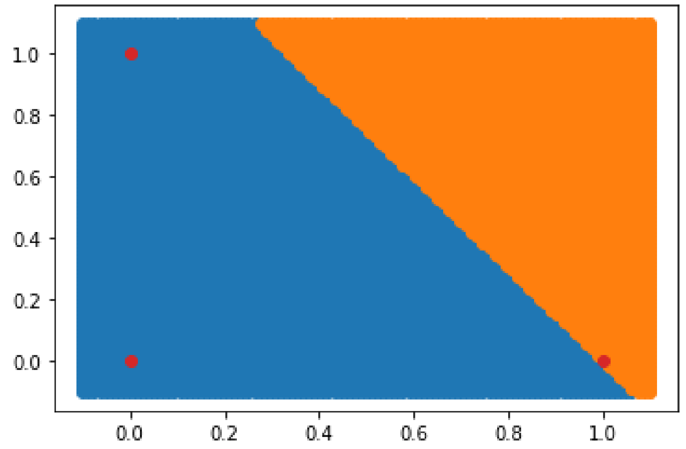

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

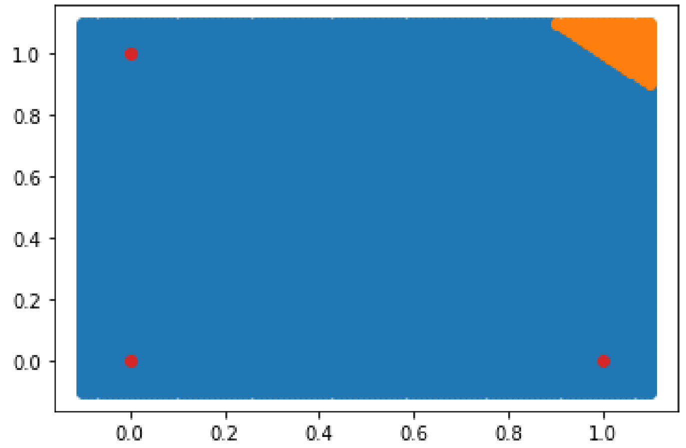

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

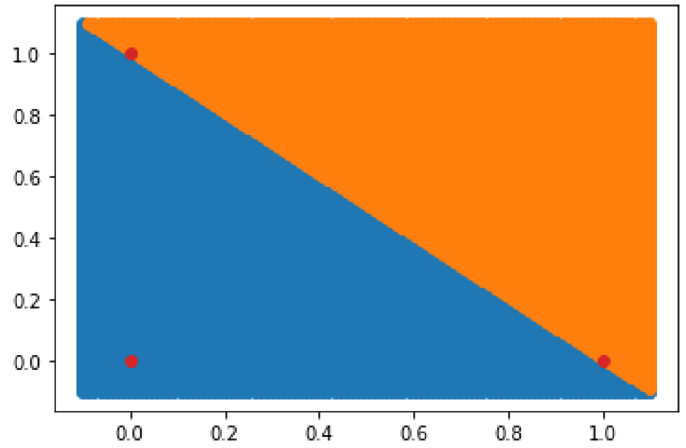

1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0

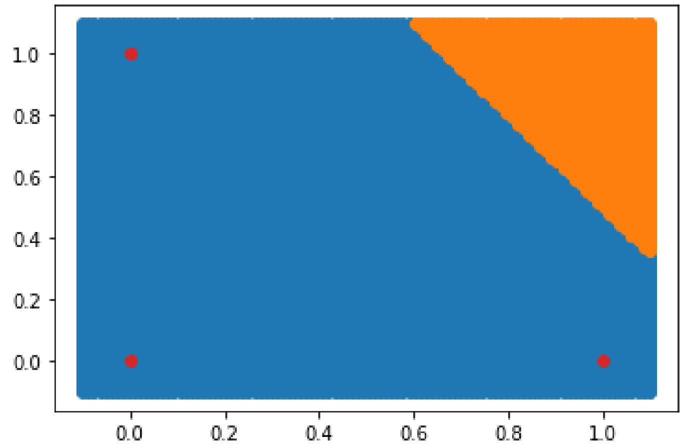

1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0


1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0


1.0
0.8
0.6
0.4
0.2
0.0
0.0 0.2 0.4 0.6 0.8 1.0







SVM

Defining the necessary functions used

In [ ]:

```
def SVM(C1, C2):
  x1 = np.array(C1.to_numpy())
  x2 = np.array(C2.to_numpy())

  x = np.vstack((x1, x2))
  y = np.hstack((np.ones(len(x1)), np.ones(len(x2)) * -1))

  X = np.dot(x, np.transpose(x))
  Y = np.outer(y, y)

  n = X.shape[0]

  H = matrix(Y * X)
  f = matrix(np.ones(n) * -1)
  A = matrix(np.diag(np.ones(n) * -1))
  a = matrix(np.zeros(n))
```

```
    B = matrix(y, (1, n))
    b = matrix(0.0)

    # solve QP problem
    solution = solvers.qp(H, f, A, a, B, b)
    print('')

    # Lagrange multipliers
    t1 = np.ravel(solution['x'])

    # Support vectors have non zero lagrange multipliers
    for i in range(0, len(t1)):
      if(t1[i] < 1e-4):
        t1[i] = 0.0
      else:
        t1[i] = round(t1[i], 4)

    t1 = np.array(t1)
    print(t1)

    # Weight vector
    w = np.zeros(X.shape[1])
    for i in range(X.shape[1]):
      w[i] = t1[i] * y[i]

    w = np.dot(w, x)

    # Intercept
    b = 0
    for i in range(len(t1)):
      if(t1[i] != 0.0):
        b = round(1 - np.dot(w, np.transpose(x[i])), 4)
        break

    return w, b
```

Code

In [ ]:
```
X = [[0, 0], [0, 1], [1, 0]]

C1 = pd.DataFrame(X)
C1
```

Out[ ]:

	0	1
0	0	0
1	0	1
2	1	0

In [ ]:
```
X = [[1, 1]]

C2 = pd.DataFrame(X)
C2
```

Out[ ]:

	0	1
0	1	1

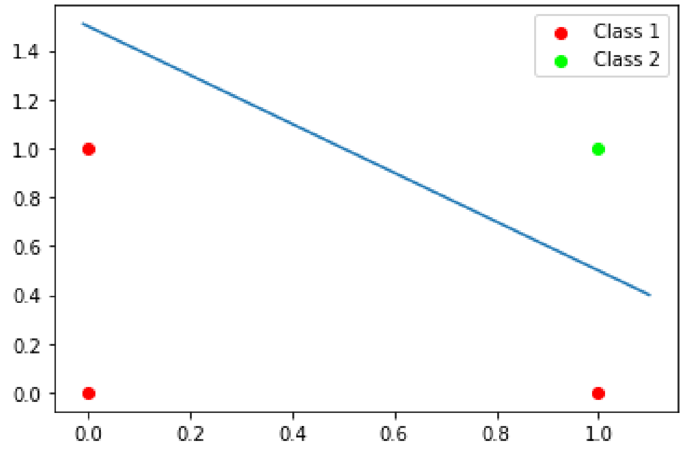
In [ ]:
```
w, b = SVM(C1, C2)


print(w)
```

```
print('')
print(b)
```

```
     pcost       dcost       gap    pres   dres
 0: -1.7500e+00 -4.2500e+00  8e+00  2e+00  2e+00
 1: -3.6653e+00 -4.3941e+00  1e+00  3e-01  3e-01
 2: -3.9945e+00 -4.0256e+00  3e-02  2e-15  3e-16
 3: -3.9999e+00 -4.0003e+00  3e-04  1e-15  4e-16
 4: -4.0000e+00 -4.0000e+00  3e-06  2e-15  4e-16
Optimal solution found.

[0. 2. 2. 4.]
[-2. -2.]

3.0
```

In [ ]:

```
x1 = np.arange(-0.01, 1.1, 0.01)

plt.scatter(C1.iloc[:, [0]], C1.iloc[:, [1]], label = 'Class 1', color = '#FF0000')
plt.scatter(C2.iloc[:, [0]], C2.iloc[:, [1]], label = 'Class 2', color = '#00FF00')
plt.legend()

slope, c = -w[0] / w[1], -b / w[1]
plt.plot(x1, slope * x1 + c)
plt.show()
```



Q2. Train a **single perceptron and SVM** to learn the two classes in the following table.

x_1	x_2	ω
2	2	1
-1	-3	0
-1	2	1
0	-1	0
1	3	1
-1	-2	0
1	-2	0
-1	-1	1

where x_1 and x_2 are the inputs and ω is the target class. Assume that all the weights of the perceptron are initialized as 0 with learning rate 0.01 and 0.5 separately. Also, tabulate the number of iterations required to converge the perception algorithm with these two learning rates.

Importing the necessary libraries

In [ ]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from cvxopt import matrix, solvers
```

Single Perceptron

Defining the necessary functions used

In [ ]:

```
def decision_plot(w, x_min, x_max, y_min, y_max):
    inp_0 = []
    inp_1 = []

    x_min = x_min - np.abs(((x_max - x_min) / 10))
    y_min = y_min - np.abs(((y_max - y_min) / 10))

    x_max = x_max + np.abs(((x_max - x_min) / 10))
    y_max = y_max + np.abs(((y_max - y_min) / 10))

    for i in np.arange(x_min, x_max, np.abs(x_max - x_min) / 100):
        for j in np.arange(y_min, y_max, np.abs(x_max - x_min) / 100):
            y_pred = np.dot(np.array([[1, i, j]]), w)

            if y_pred < 0:
                inp_0.append(np.array((i, j)))

            else:
                inp_1.append(np.array((i, j)))

    return inp_0, inp_1
```

In [ ]:

```
def perceptron_training_alg(X, Y, learning_rate):
```

```
  # Adding Bias to the input and Randomly initializing weights
  num_samples, num_features = X.shape

  X = np.hstack((np.ones((num_samples, 1)), X))
  W = np.zeros(num_features + 1).reshape((-1, 1))

  weight_vector = []
  weight_vector.append(W)

  print(W, "\n")

  # Running the algorithm until all the data points are correctly classified
  samples_misclassified = True
  cnt=0
  while samples_misclassified and cnt < 100000:
    # Initially we'll assume that there are no misclassified samples
    samples_misclassified = False

    for index, x in enumerate(X):
      if cnt >= 100000:
        return W, np.array(weight_vector)

      x = Y[index] * x

      if np.dot(x.reshape((1, -1)), W) <= 0: # if samples misclassified
        samples_misclassified = True
        W = W + (learning_rate * x.reshape((-1, 1)))     # Gradient Descent Step
        weight_vector.append(W)

        print(W, "\n")
        cnt += 1

  return W, np.array(weight_vector)
```

In [ ]:

```
def sorter(X, Y):
  X_new0 = []
  Y_new0 = []

  X_new1 = []
  Y_new1 = []

  for x, y in zip(X, Y):
    if y == 0:
      Y_new0.append(y)
      X_new0.append(x)

    else:
      Y_new1.append(y)
      X_new1.append(x)

  X0 = []
  Y0 = []

  X1 = []
  Y1 = []

  if len(X_new0) > 0:
    X0 = np.stack(X_new0)
    Y0 = np.stack(Y_new0)

  elif len(X_new1) > 0:
    X1 = np.stack(X_new1)
    Y1 = np.stack(Y_new1)

  if len(X0) > 0 and len(X1) > 0:
    X = np.concatenate((X0, X1))
    Y = np.concatenate((Y0, Y1))
```

```
  elif len(X0) > 0:
    X = X0
    Y = Y0

  else:
    X = X1
    Y = Y1

  return X, Y
```

In [ ]:

```
def decision_boundary(W, X, Y):
  X, Y = sorter(X, Y)
  num_samples, num_features = X.shape
  np2 = np.hstack((np.ones((num_samples, 1)), X))

  X=np2

  x1, y1 = decision_plot(W, min(X[:,1]), max(X[:,1]), min(X[:,2]), max(X[:,2]))

  plt.scatter([i[0] for i in x1], [i[1] for i in x1])
  plt.scatter([i[0] for i in y1], [i[1] for i in y1])
  plt.scatter(X[:np.argmax(Y), 1], X[:np.argmax(Y), 2])
  plt.scatter(X[np.argmax(Y):, 1], X[np.argmax(Y):, 2])
  plt.show()
```

Code

In [ ]:

```
X = np.array([[2, 2], [-1, -3], [-1, 2], [0, -1], [1, 3], [-1, -2], [1, -2], [-1, -1]])
Y = [[1], [-1], [1], [-1], [1], [-1], [-1], [1]]
ans = [[1], [0], [1], [0], [1], [0], [0], [1]]

len(X)
```

Out[ ]: 8

Learning Rate = 0.01

In [ ]:

```
W1, weight_vector1 = perceptron_training_alg(X, Y, 0.01)
```

```
[[0.]
 [0.]
 [0.]]

[[0.01]
 [0.02]
 [0.02]]

[[0.02]
 [0.01]
 [0.01]]

[[0.01]
 [0.01]
 [0.02]]

[[0.02]
 [0.  ]
 [0.01]]

[[0.01]
 [0.  ]
 [0.02]]

[[ 0.02]
 [-0.01]
 [ 0.01]]
```

```
[[ 0.01]
 [-0.01]
 [ 0.02]]

[[ 0.02]
 [-0.02]
 [ 0.01]]

[[0.03]
 [0.  ]
 [0.03]]

[[0.02]
 [0.  ]
 [0.04]]

[[ 0.03]
 [-0.01]
 [ 0.03]]

[[ 0.02]
 [-0.01]
 [ 0.04]]

[[ 0.03]
 [-0.02]
 [ 0.03]]

[[ 0.02]
 [-0.02]
 [ 0.04]]

[[ 0.03]
 [-0.03]
 [ 0.03]]

[[ 0.02]
 [-0.03]
 [ 0.04]]
```

In [ ]:

```
W1
```

Out[ ]:

```
array([[ 0.02],
       [-0.03],
       [ 0.04]])
```

In [ ]:

```
weight_vector1
```

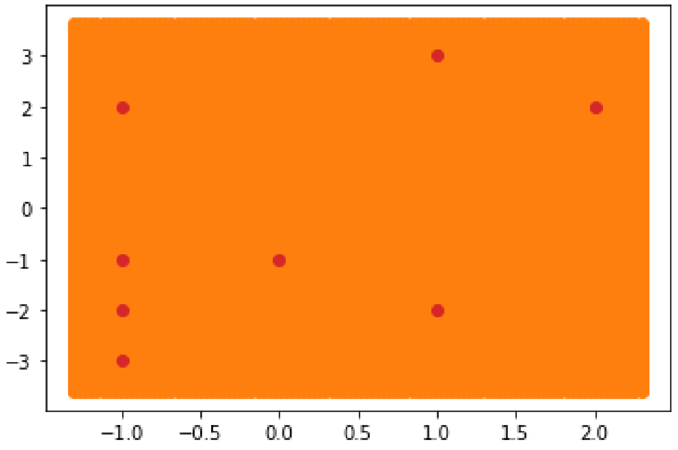
Out[ ]:

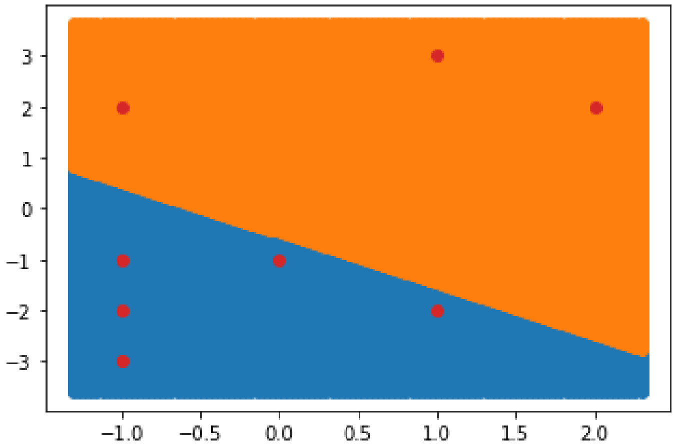
```
array([[[ 0.  ],
        [ 0.  ],
        [ 0.  ]],

       [[ 0.01],
        [ 0.02],
        [ 0.02]],

       [[ 0.02],
        [ 0.01],
        [ 0.01]],

       [[ 0.01],
        [ 0.01],
        [ 0.02]],

       [[ 0.02],
        [ 0.  ],
        [ 0.01]],

       [[ 0.01],
        [ 0.  ],
        [ 0.02]],
```

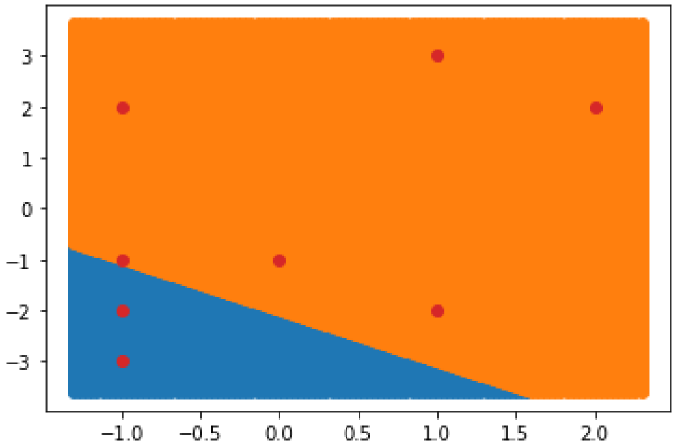
```
[[ 0.02],
 [-0.01],
 [ 0.01]],

[[ 0.01],
 [-0.01],
 [ 0.02]],

[[ 0.02],
 [-0.02],
 [ 0.01]],

[[ 0.03],
 [ 0.  ],
 [ 0.03]],

[[ 0.02],
 [ 0.  ],
 [ 0.04]],

[[ 0.03],
 [-0.01],
 [ 0.03]],

[[ 0.02],
 [-0.01],
 [ 0.04]],

[[ 0.03],
 [-0.02],
 [ 0.03]],

[[ 0.02],
 [-0.02],
 [ 0.04]],

[[ 0.03],
 [-0.03],
 [ 0.03]],

[[ 0.02],
 [-0.03],
 [ 0.04]]]])
```

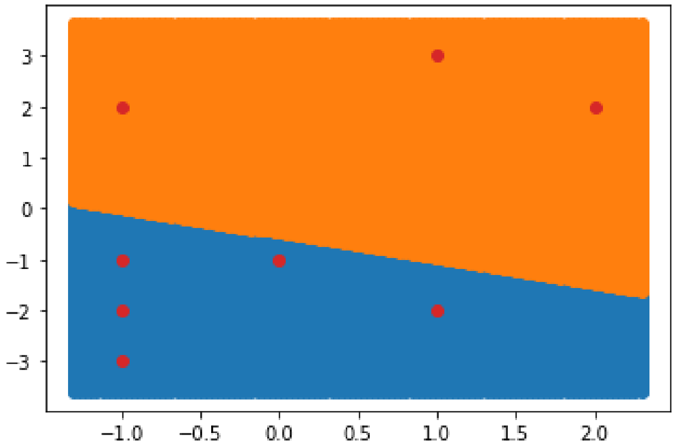
In [ ]:

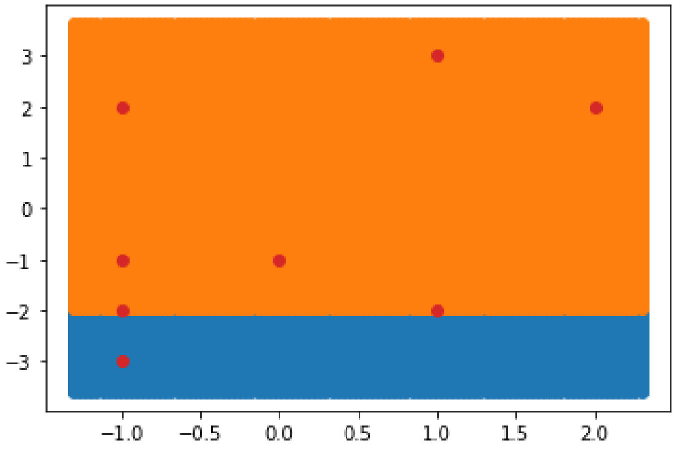
```
for w in weight_vector1:
  decision_boundary(w, X, ans)
```

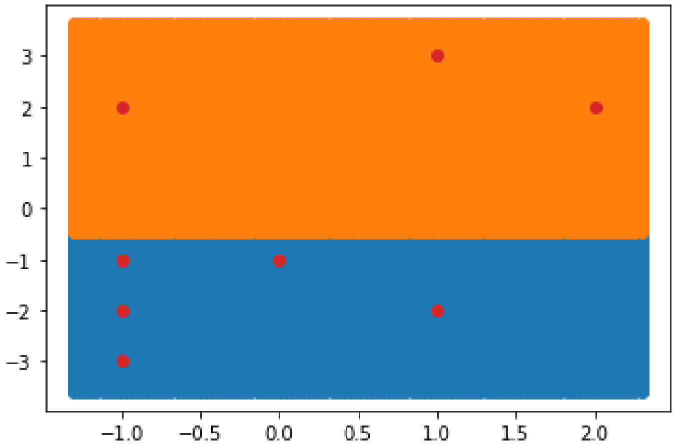



3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0

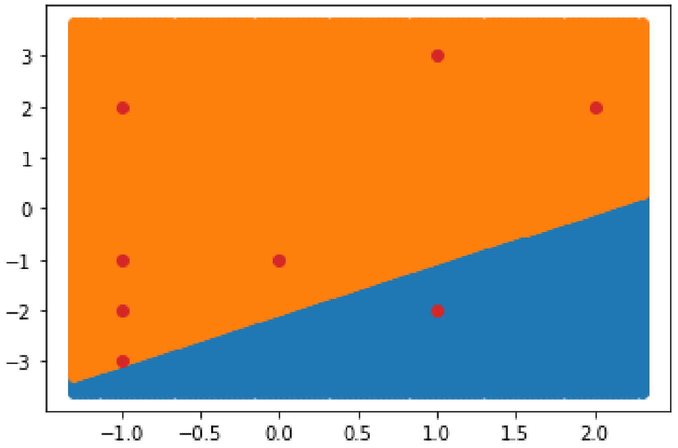

3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0


3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0

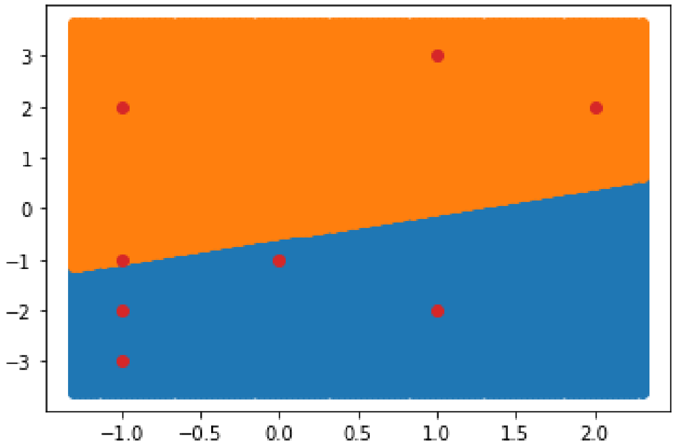

3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0



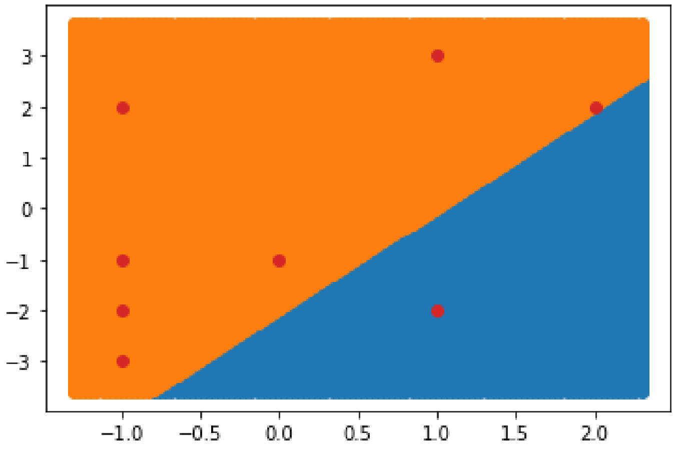
3
2
1
0
−1
−2
−3
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0



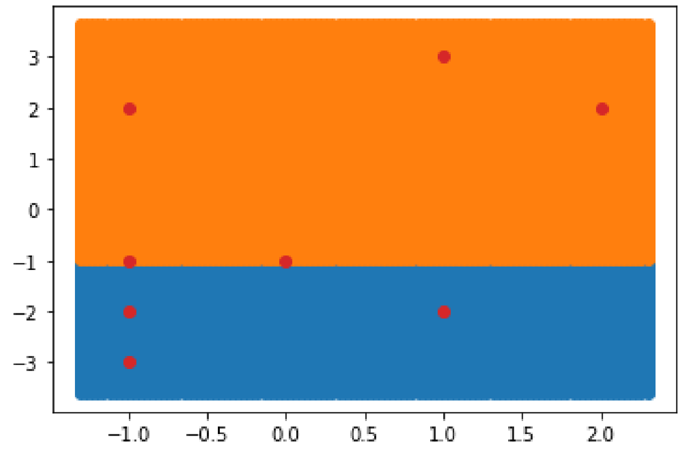
3
2
1
0
−1
−2
−3
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

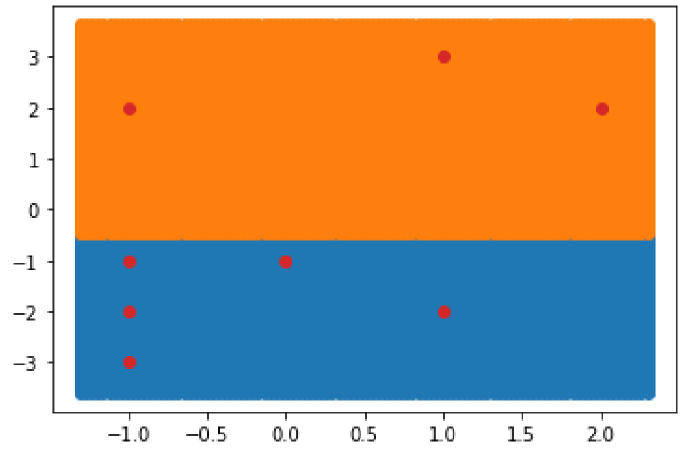


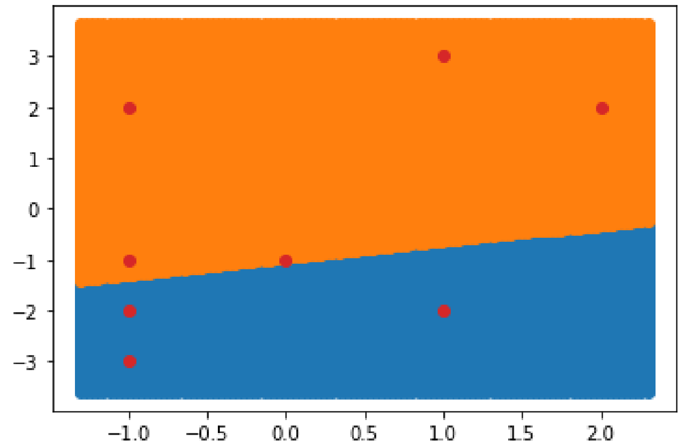
3
2
1
0
−1
−2
−3
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

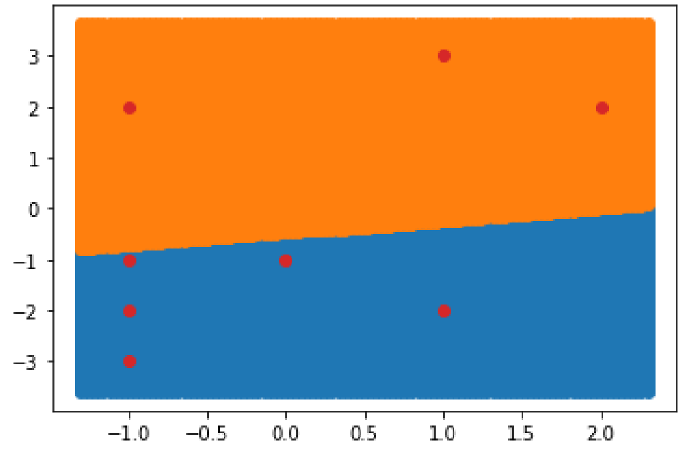


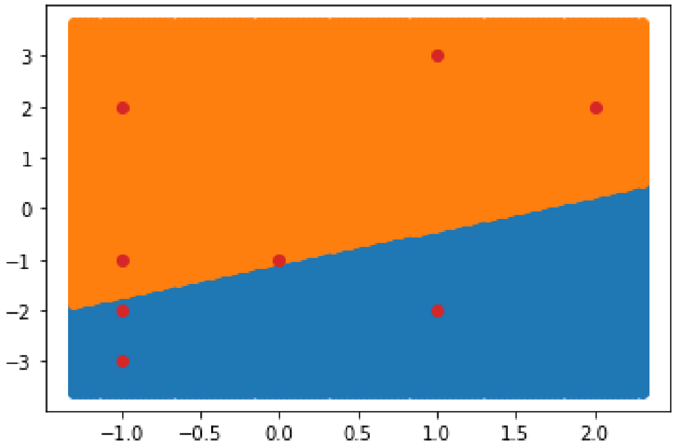
3
2
1
0
−1
−2
−3
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

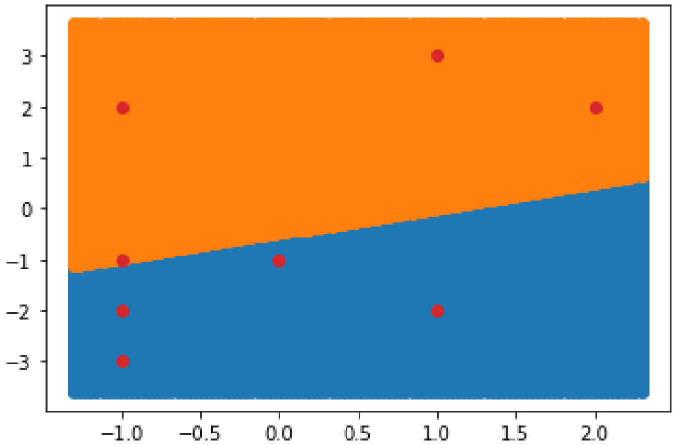


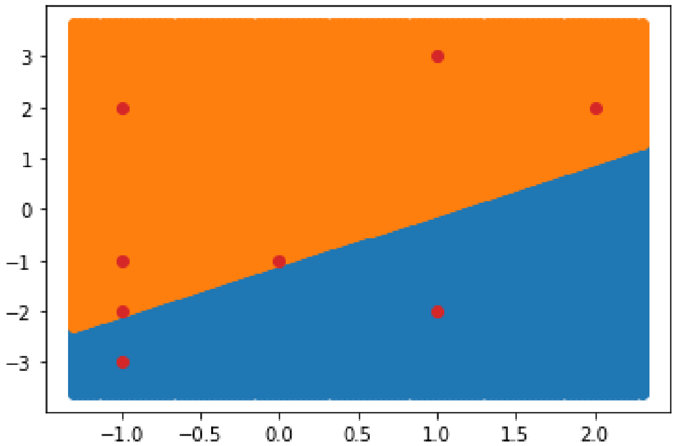


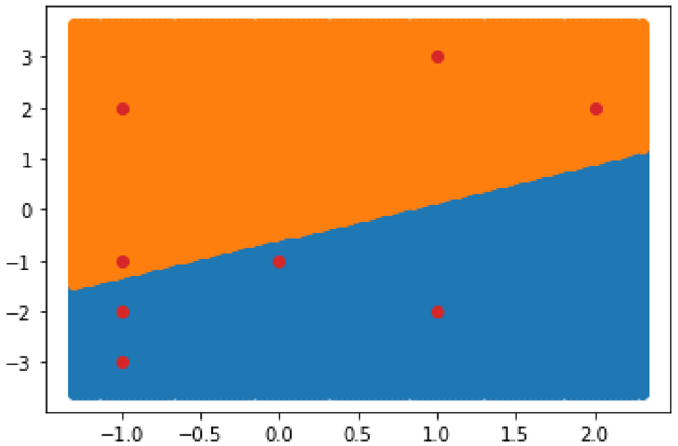












Learning Rate = 0.5

In [ ]:
```
W2, weight_vector2 = perceptron_training_alg(X,Y,0.5)
```

```
[[0.]
```

```
 [0.]
 [0.]]

[[0.5]
 [1. ]
 [1. ]]

[[1. ]
 [0.5]
 [0.5]]

[[0.5]
 [0.5]
 [1. ]]

[[1. ]
 [0. ]
 [0.5]]

[[0.5]
 [0. ]
 [1. ]]

[[ 1. ]
 [-0.5]
 [ 0.5]]

[[0.5]
 [0. ]
 [2. ]]

[[ 1. ]
 [-0.5]
 [ 1.5]]

[[ 1.5]
 [-1. ]
 [ 1. ]]

[[ 1. ]
 [-1. ]
 [ 1.5]]
```

In [ ]:

```
W2
```

Out[ ]:

```
array([[ 1. ],
       [-1. ],
       [ 1.5]])
```

In [ ]:

```
weight_vector2
```

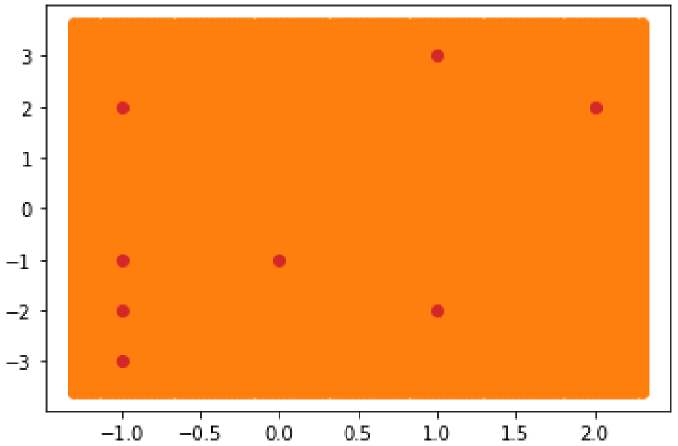
Out[ ]:

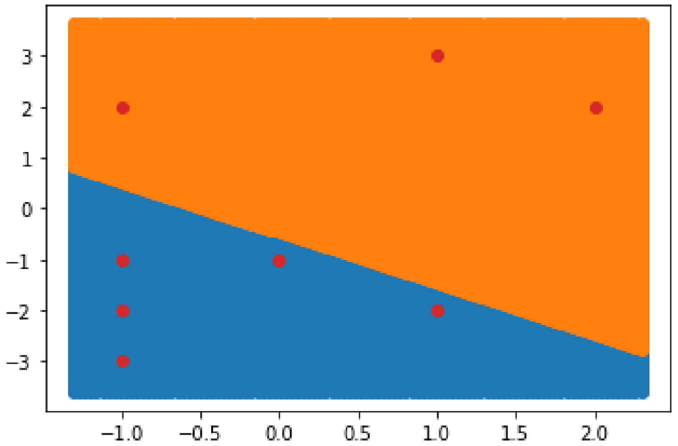
```
array([[[ 0. ],
        [ 0. ],
        [ 0. ]],

       [[ 0.5],
        [ 1. ],
        [ 1. ]],

       [[ 1. ],
        [ 0.5],
        [ 0.5]],

       [[ 0.5],
        [ 0.5],
        [ 1. ]],

       [[ 1. ],
        [ 0. ],
        [ 0.5]],
```

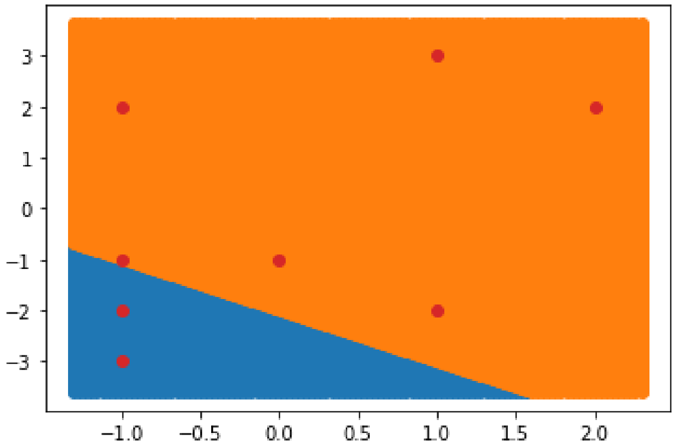
```
 [[ 0.5],
  [ 0. ],
  [ 1. ]],

 [[ 1. ],
  [-0.5],
  [ 0.5]],

 [[ 0.5],
  [ 0. ],
  [ 2. ]],

 [[ 1. ],
  [-0.5],
  [ 1.5]],

 [[ 1.5],
  [-1. ],
  [ 1. ]],

 [[ 1. ],
  [-1. ],
  [ 1.5]]]])
```

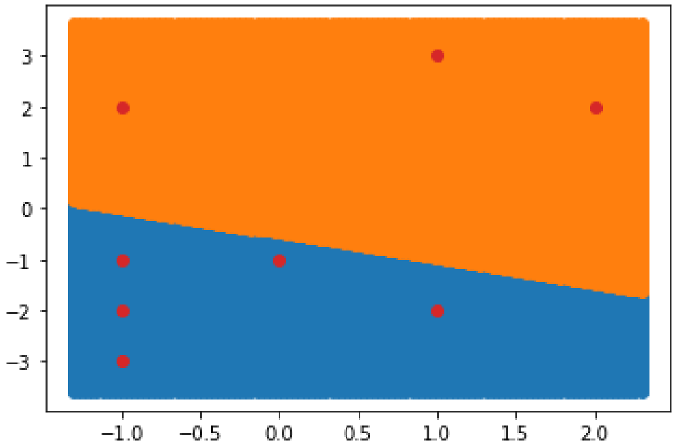
In [ ]:

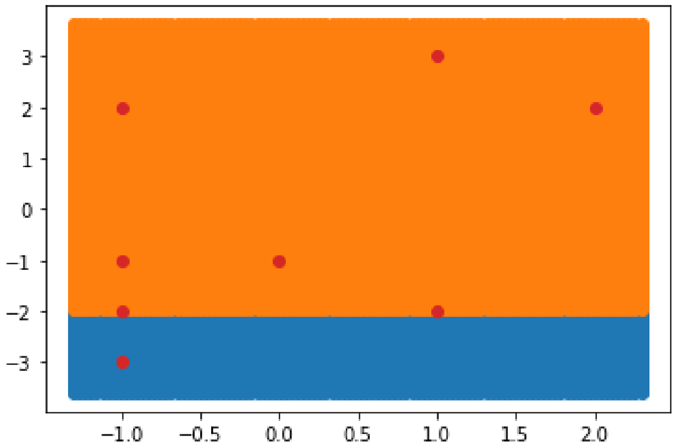
```
for w in weight_vector2:
  decision_boundary(w, X, ans)
```

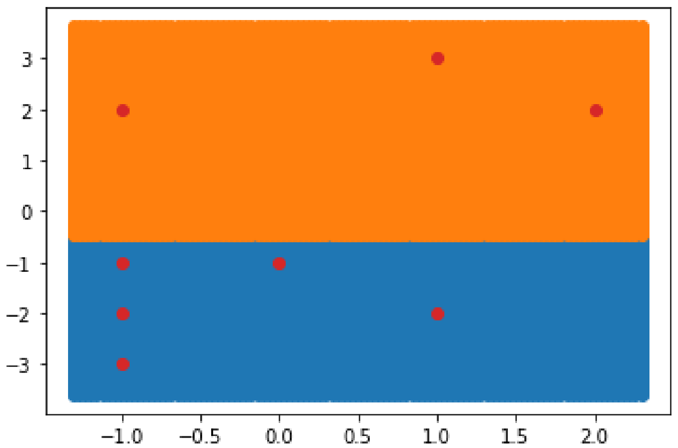


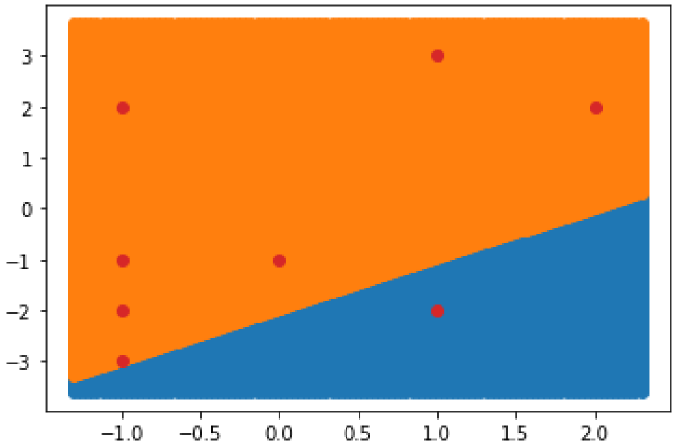



3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0

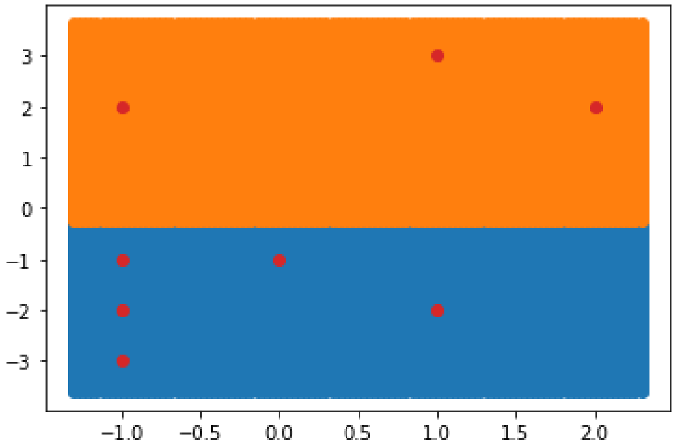

3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0


3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0

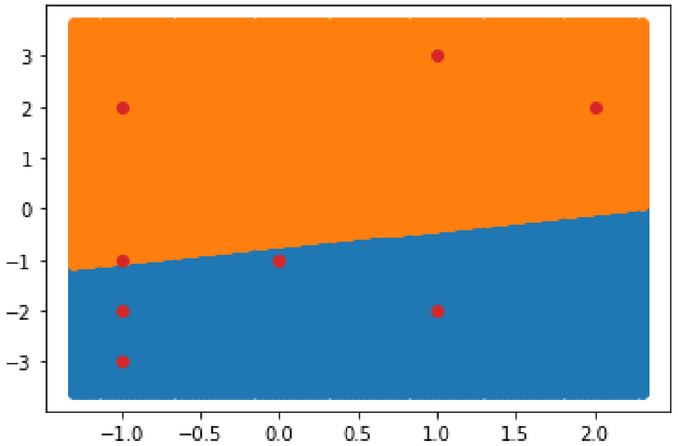

3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0



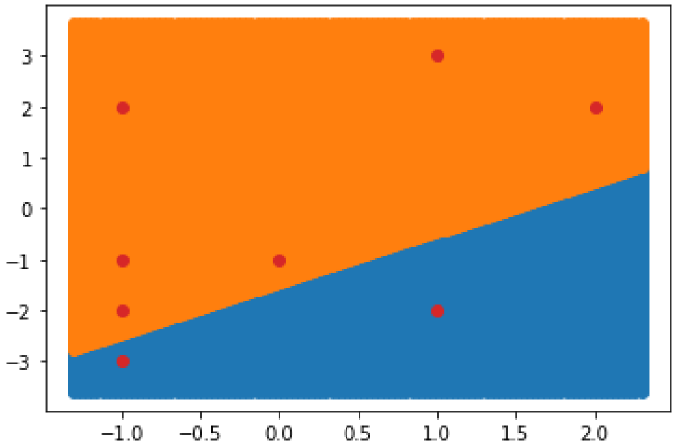
3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0



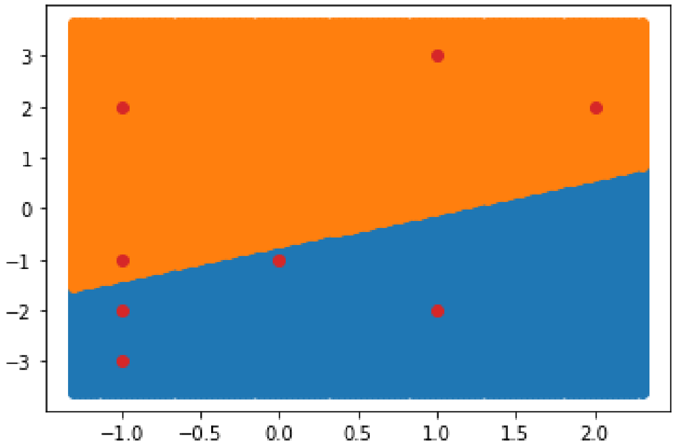
3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0



3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0



3
2
1
0
−1
−2
−3
−1.0 −0.5 0.0 0.5 1.0 1.5 2.0



SVM

Defining the necessary functions used

In [ ]:

```
def SVM(C1, C2):
  x1 = np.array(C1.to_numpy())
  x2 = np.array(C2.to_numpy())

  x = np.vstack((x1, x2))
  y = np.hstack((np.ones(len(x1)), np.ones(len(x2)) * -1))

  X = np.dot(x, np.transpose(x))
  Y = np.outer(y, y)

  n = X.shape[0]

  H = matrix(Y * X)
  f = matrix(np.ones(n) * -1)
  A = matrix(np.diag(np.ones(n) * -1))
  a = matrix(np.zeros(n))
  B = matrix(y, (1, n))
  b = matrix(0.0)

  # solve QP problem
  solution = solvers.qp(H, f, A, a, B, b)
  print('')

  # Lagrange multipliers
  t1 = np.ravel(solution['x'])

  # Support vectors have non zero lagrange multipliers
  for i in range(0, len(t1)):
    if(t1[i] < 1e-4):
      t1[i] = 0.0
    else:
      t1[i] = round(t1[i], 4)

  t1 = np.array(t1)
  print(t1)

  # Weight vector
  w = np.zeros(X.shape[1])
  for i in range(X.shape[1]):
    w[i] = t1[i] * y[i]

  w = np.dot(w, x)

  # Intercept
  b = 0
  for i in range(len(t1)):
    if(t1[i] != 0.0):
```

```
            b = round(1 - np.dot(w, np.transpose(x[i])), 4)
            break

    return w, b
```

Code

In [ ]:
```
X = [[-1, -3], [0, -1], [-1, -2], [1, -2]]

C1 = pd.DataFrame(X)
C1
```

Out[ ]:

	0	1
0	-1	-3
1	0	-1
2	-1	-2
3	1	-2

In [ ]:
```
X = [[2, 2], [-1, 2], [1, 3], [-1, -1]]

C2 = pd.DataFrame(X)
C2
```

Out[ ]:

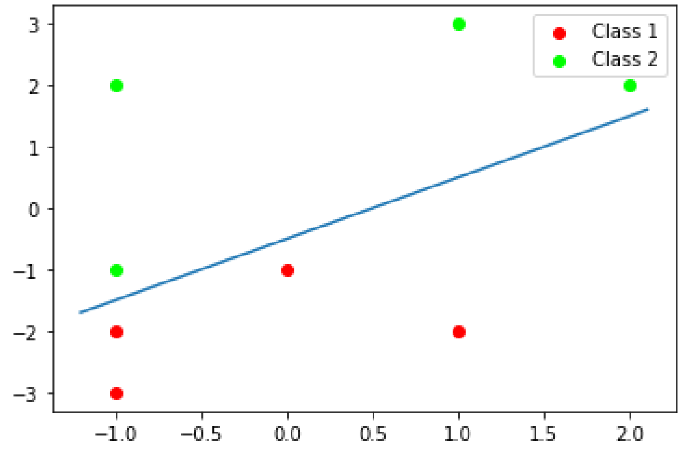
	0	1
0	2	2
1	-1	2
2	1	3
3	-1	-1

In [ ]:
```
w, b = SVM(C1, C2)


print(w)
print('')
print(b)
```

```
     pcost       dcost       gap    pres   dres
 0: -2.2863e+00 -5.2491e+00  2e+01  3e+00  2e+00
 1: -6.6436e+00 -7.0845e+00  6e+00  1e+00  7e-01
 2: -3.2806e+00 -4.6847e+00  1e+00  1e-15  1e-14
 3: -3.9727e+00 -4.0106e+00  4e-02  1e-15  3e-15
 4: -3.9997e+00 -4.0001e+00  4e-04  1e-15  9e-16
 5: -4.0000e+00 -4.0000e+00  4e-06  9e-16  1e-15
Optimal solution found.

[0.     3.3296 0.6704 0.     0.4432 0.     0.     3.5568]
[ 2. -2.]

-1.0
```

In [ ]:
```
x1 = np.arange(-1.2, 2.2, 0.1)

plt.scatter(C1.iloc[:, [0]], C1.iloc[:, [1]], label = 'Class 1', color = '#FF0000')
plt.scatter(C2.iloc[:, [0]], C2.iloc[:, [1]], label = 'Class 2', color = '#00FF00')
plt.legend()

slope, c = -w[0] / w[1], -b / w[1]
```

```
plt.plot(x1, slope * x1 + c)
plt.show()
```



Q3. In the given I set of images from poly1.png to poly14.png, let poly1 to poly 7 belong to class 1 and poly 8 to poly 14 belong to class 2. Assume that all the weights of the perceptron are initialized as 0 with the learning rate of 0.01.

- Identify two discriminant features x_1 and x_2 for the two target classes $\omega = \{\omega_1, \omega_2\}$. Here, ω_1 - class 1 and ω_2 - class 2.
- Generate an input feature vector X for all the images mapping them to a corresponding taget classes ω_i, where $i \in (1, 2)$.
- Train a **single perceptron and SVM** to learn the feature vector X mapping to ω.
- Plot and draw the final decision boundary separating the three classes

Importing the necessary libraries

In [ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import cv2 as cv
from cvxopt import matrix, solvers

from matplotlib.colors import ListedColormap
import matplotlib.patches as mpatches
```

In [ ]:
```
from google.colab import files
uploaded = files.upload()
```

Choose files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving poly1.png to poly1.png
Saving poly2.png to poly2.png
Saving poly3.png to poly3.png
Saving poly4.png to poly4.png
Saving poly5.png to poly5.png
Saving poly6.png to poly6.png
Saving poly7.png to poly7.png
Saving poly8.png to poly8.png
Saving poly9.png to poly9.png
Saving poly10.png to poly10.png
Saving poly11.png to poly11.png
Saving poly12.png to poly12.png
Saving poly13.png to poly13.png
Saving poly14.png to poly14.png
```

In [ ]:
```
ls=[]

class1 = ["poly1.png", "poly2.png", "poly3.png", "poly4.png", "poly5.png", "poly6.png", "poly7.png"]

for image in class1:
    img = cv.imread(image)
    img = cv.resize(img, (256, 256))

    b, g, r = cv.split(img)

    img = cv.merge((r, g, b))
    ls.append(img)
```

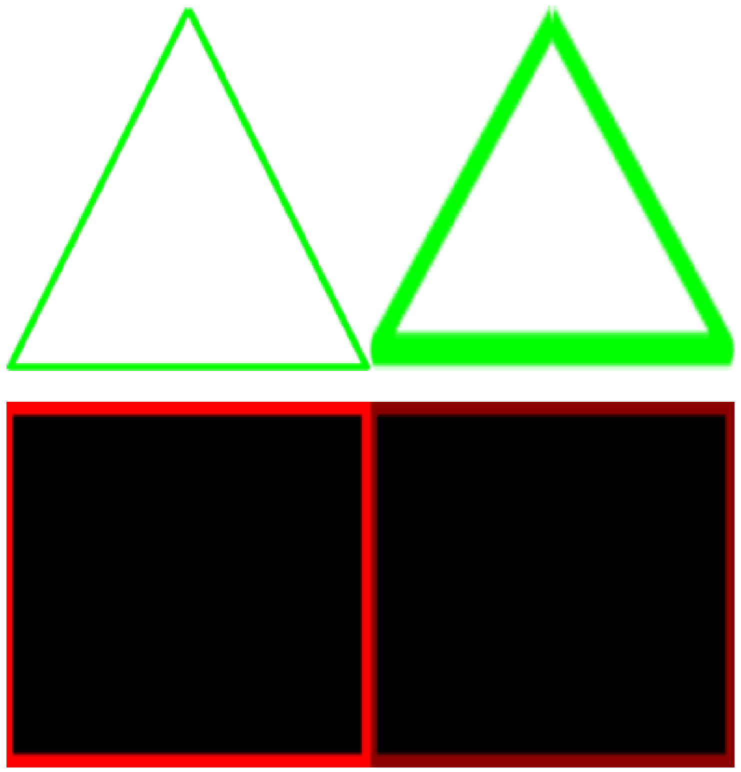
In [ ]:
```
ls2=[]

class2 = ["poly8.png", "poly9.png", "poly10.png", "poly11.png", "poly12.png", "poly13.png", "poly14.png"]

for image in class2:
    img = cv.imread(image)
    img = cv.resize(img, (256, 256))
```

```
    b, g, r = cv.split(img)

    img = cv.merge((r, g, b))
    ls2.append(img)
```

In [ ]:

```
plt.figure(figsize = (7, 7))
plt.imshow(np.concatenate((ls[0], ls[1]), axis = 1))
plt.axis('off')

plt.figure(figsize = (7, 7))
plt.imshow(np.concatenate((ls2[0], ls2[1]), axis = 1))
plt.axis('off')
```

Out[ ]: (-0.5, 511.5, 255.5, -0.5)



Single Perceptron

Defining the necessary functions used

In [ ]:

```
def perceptron_training_alg(X, Y, learning_rate):
  # Adding Bias to the input and Randomly initializing weights
  num_samples, num_features = X.shape

  X = np.hstack((np.ones((num_samples, 1)), X))
  W = np.zeros(num_features + 1).reshape((-1, 1))

  weight_vector = []
  weight_vector.append(W)

  print(W, "\n")

  # Running the algorithm until all the data points are correctly classified
  samples_misclassified = True
  cnt=0
  while samples_misclassified and cnt < 100000:
    # Initially we'll assume that there are no misclassified samples
    samples_misclassified = False
```

```
    for index, x in enumerate(X):
      if cnt >= 100000:
        return W, np.array(weight_vector)

      x = Y[index] * x

      if np.dot(x.reshape((1, -1)), W) <= 0: # if samples misclassified
        samples_misclassified = True
        W = W + (learning_rate * x.reshape((-1, 1)))     # Gradient Descent Step
        weight_vector.append(W)

        print(W, "\n")
        cnt += 1

  return W, np.array(weight_vector)
```

Code

In [ ]:

```
X = []

for img in ls:
    r, g, b = cv.split(img)
    X.append([np.sum(r)/np.sum(r+g+b), np.sum(g)/np.sum(r+g+b)])

for img in ls2:
    r, g, b = cv.split(img)
    X.append([np.sum(r)/np.sum(r+g+b), np.sum(g)/np.sum(r+g+b)])
```

In [ ]:

```
Y=np.concatenate((np.ones(len(ls))*-1,np.ones(len(ls2))))
ans=np.concatenate((np.zeros(len(ls)),np.ones(len(ls2))))
```

In [ ]:

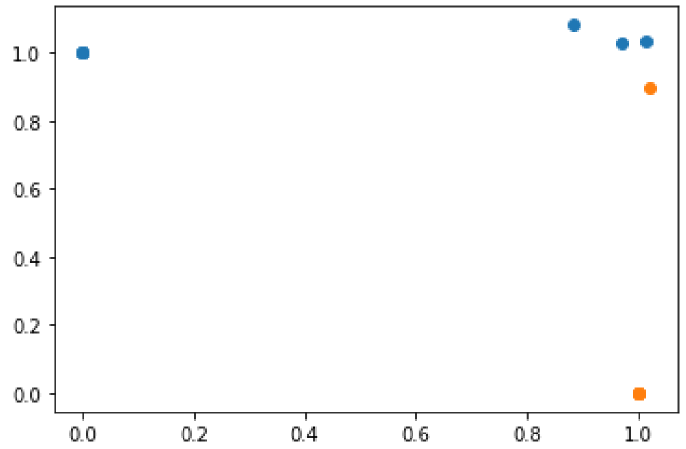
```
X=np.stack(X)
X
```

Out[ ]:

```
array([[0.96962233, 1.02958576],
       [0.88314034, 1.08201861],
       [0.        , 1.        ],
       [0.        , 1.        ],
       [0.        , 1.        ],
       [0.        , 1.        ],
       [1.01528142, 1.03526193],
       [1.        , 0.        ],
       [1.        , 0.        ],
       [1.        , 0.        ],
       [1.        , 0.        ],
       [1.01927362, 0.89945752],
       [1.        , 0.        ],
       [1.        , 0.        ]])
```

In [ ]:

```
plt.scatter(X[:7,0],X[:7,1])
plt.scatter(X[7:,0],X[7:,1])
plt.show()
```



In [ ]:

```
W, weight_vector=perceptron_training_alg(X,Y,0.01)
```

```
[[0.]
 [0.]
 [0.]]

[[-0.01      ]
 [-0.00969622]
 [-0.01029586]]

[[ 0.        ]
 [ 0.00030378]
 [-0.01029586]]

[[ 0.01      ]
 [ 0.01049651]
 [-0.00130128]]

[[ 0.        ]
 [ 0.00080029]
 [-0.01159714]]

[[ 0.01      ]
 [ 0.01099303]
 [-0.00260256]]

[[ 0.        ]
 [ 0.0012968 ]
 [-0.01289842]]

[[ 0.01      ]
 [ 0.01148954]
 [-0.00390385]]

[[ 0.        ]
 [ 0.00179332]
 [-0.0141997 ]]

[[ 0.01      ]
 [ 0.01198605]
 [-0.00520513]]

[[ 0.        ]
 [ 0.00228983]
 [-0.01550099]]

[[ 0.01      ]
 [ 0.01248256]
 [-0.00650641]]

[[ 0.        ]
 [ 0.00278634]
 [-0.01680227]]

[[ 0.01      ]
 [ 0.01297908]
```

```
 [-0.00780769]]

[[ 0.        ]
 [ 0.00328285]
 [-0.01810355]]

[[ 0.01      ]
 [ 0.01347559]
 [-0.00910898]]

[[ 0.        ]
 [ 0.00377937]
 [-0.01940483]]

[[ 0.01      ]
 [ 0.0139721 ]
 [-0.01041026]]

[[ 0.        ]
 [ 0.00427588]
 [-0.02070612]]

[[ 0.01      ]
 [ 0.01446862]
 [-0.01171154]]

[[ 0.        ]
 [ 0.00477239]
 [-0.0220074 ]]

[[ 0.01      ]
 [ 0.01496513]
 [-0.01301282]]

[[ 0.        ]
 [ 0.00526891]
 [-0.02330868]]

[[ 0.01      ]
 [ 0.01546164]
 [-0.01431411]]

[[ 0.        ]
 [ 0.00576542]
 [-0.02460996]]

[[ 0.01      ]
 [ 0.01595816]
 [-0.01561539]]

[[ 0.        ]
 [ 0.00626193]
 [-0.02591125]]

[[ 0.01      ]
 [ 0.01645467]
 [-0.01691667]]

[[ 0.        ]
 [ 0.00675844]
 [-0.02721253]]

[[ 0.01      ]
 [ 0.01695118]
 [-0.01821795]]

[[ 0.        ]
 [ 0.00725496]
 [-0.02851381]]

[[ 0.01      ]
 [ 0.01744769]
 [-0.01951924]]

[[ 0.        ]
```

```
 [ 0.00775147]
 [-0.02981509]]

[[ 0.01      ]
 [ 0.01794421]
 [-0.02082052]]

[[ 0.        ]
 [ 0.00824798]
 [-0.03111638]]

[[ 0.01      ]
 [ 0.01844072]
 [-0.0221218 ]]

[[ 0.        ]
 [ 0.0087445 ]
 [-0.03241766]]

[[ 0.01      ]
 [ 0.01893723]
 [-0.02342308]]

[[ 0.        ]
 [ 0.00924101]
 [-0.03371894]]

[[ 0.01      ]
 [ 0.01943375]
 [-0.02472437]]

[[ 0.        ]
 [ 0.00973752]
 [-0.03502022]]

[[ 0.01      ]
 [ 0.01993026]
 [-0.02602565]]

[[ 0.        ]
 [ 0.01023404]
 [-0.03632151]]

[[ 0.01      ]
 [ 0.02042677]
 [-0.02732693]]

[[ 0.        ]
 [ 0.01073055]
 [-0.03762279]]

[[ 0.01      ]
 [ 0.02092328]
 [-0.02862821]]

[[ 0.        ]
 [ 0.01122706]
 [-0.03892407]]

[[ 0.01     ]
 [ 0.0214198]
 [-0.0299295]]

[[ 0.        ]
 [ 0.01126698]
 [-0.04028211]]

[[ 0.01      ]
 [ 0.02145972]
 [-0.03128754]]
```

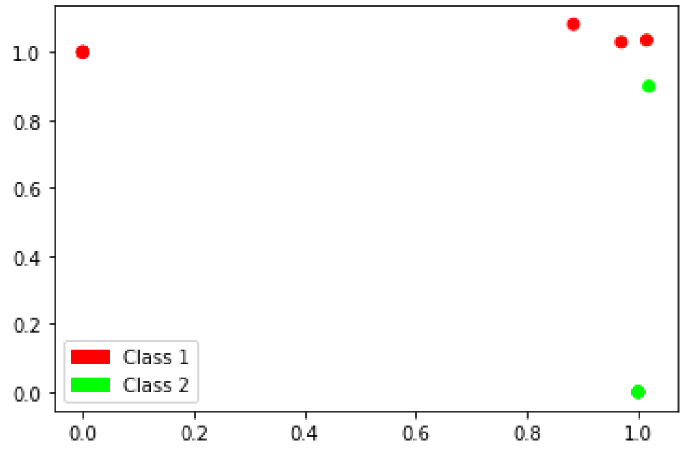
In [ ]:

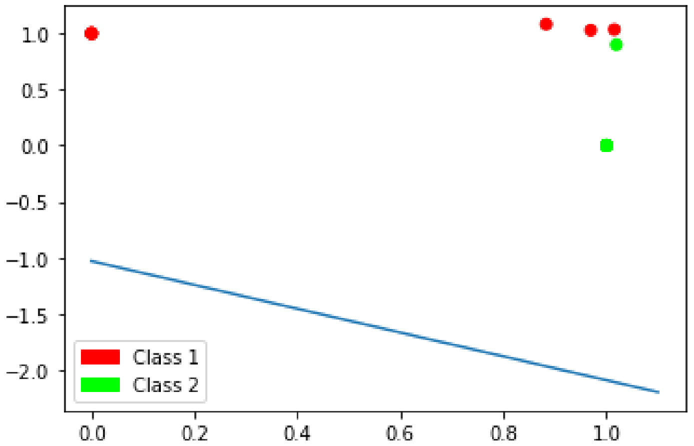
```
W
```

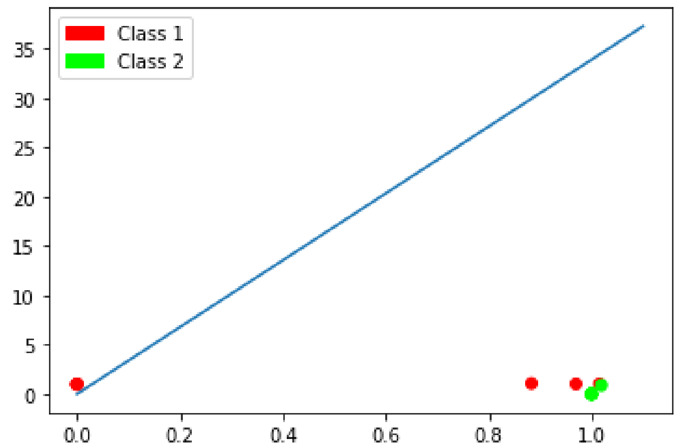
Out[ ]:
```
array([[ 0.01      ],
       [ 0.02145972],
       [-0.03128754]])
```

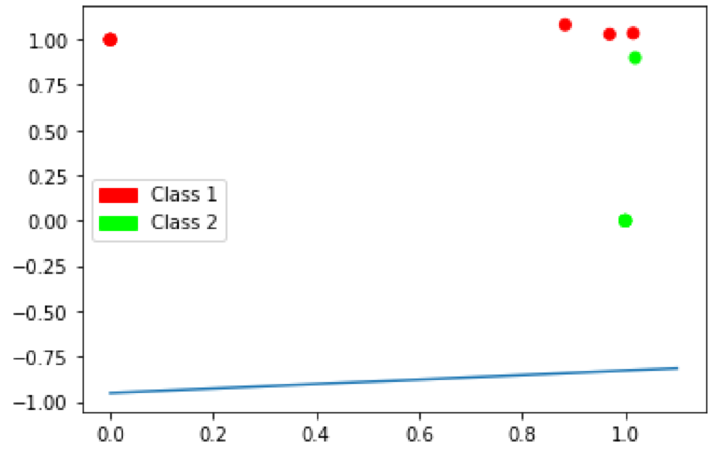
In [ ]:
```
cm  = ListedColormap(['#FF0000', '#00FF00'])
patch0 = mpatches.Patch(color = '#FF0000', label = 'Class 1')
patch1 = mpatches.Patch(color = '#00FF00', label = 'Class 2')

x1 = np.arange(0, 1.2, 0.1)

for w in weight_vector:
  plt.scatter(X[:,0], X[:,1], c = [0] * 7 + [1] * 7, cmap = cm)
  plt.legend(handles = [patch0, patch1])

  slope, c = -w[2] / w[1], -w[0] / w[1]
  plt.plot(x1, slope * x1 + c)
  plt.show()
```

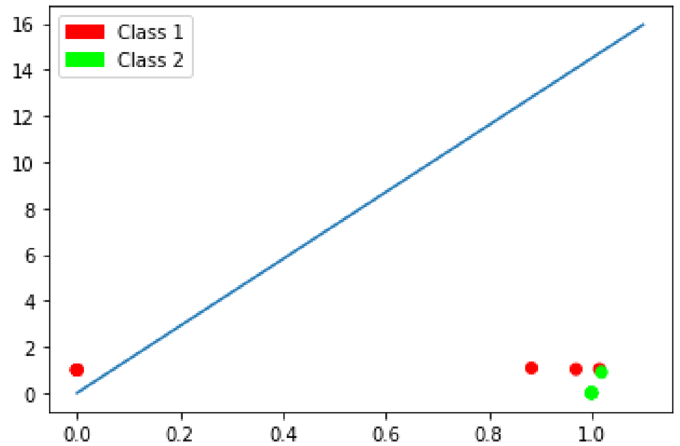
```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:11: RuntimeWarning: invalid value encountered in true_divide
  # This is added back by InteractiveShellApp.init_path()
```

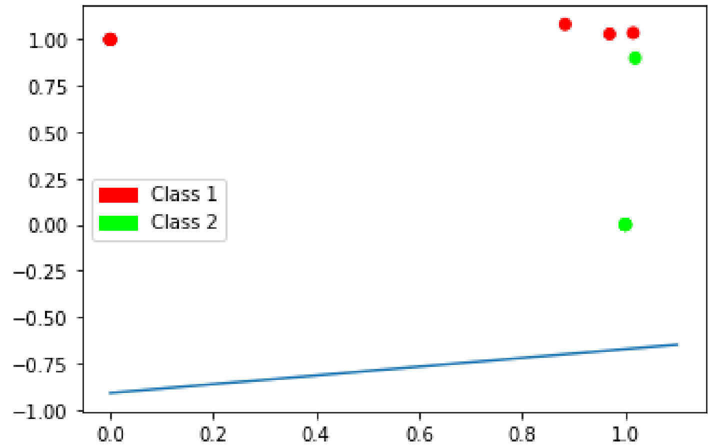


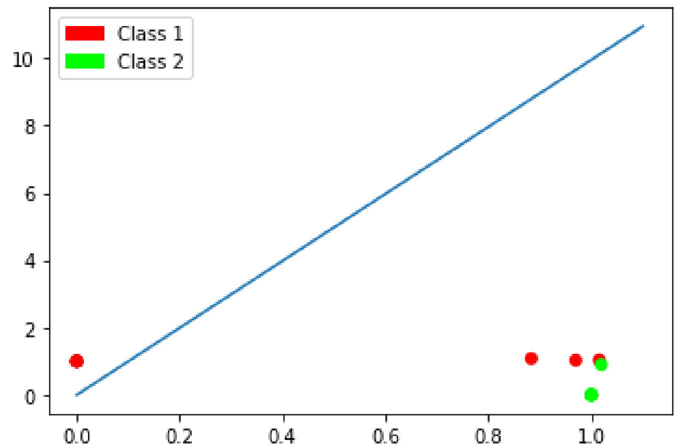


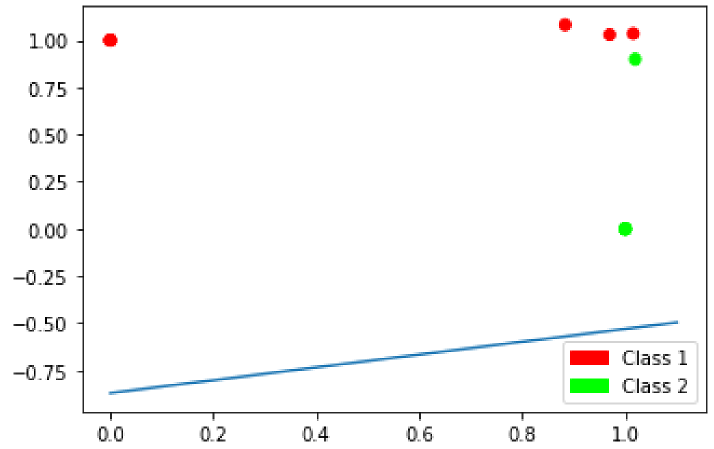


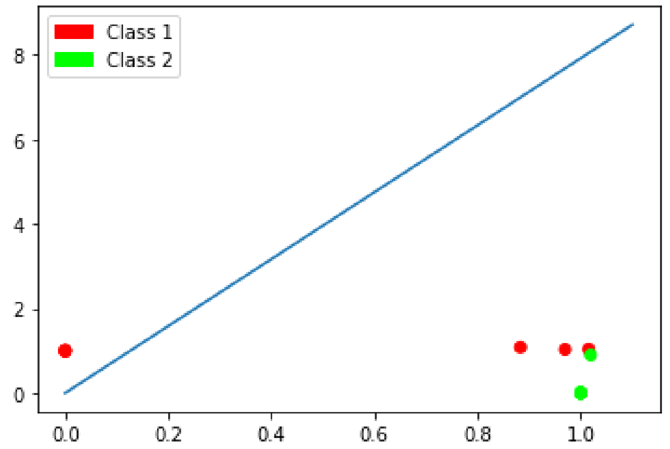

Class 1
Class 2

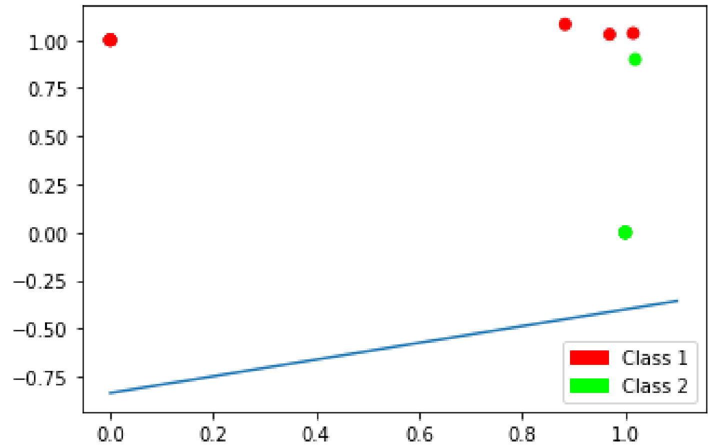

Class 1
Class 2

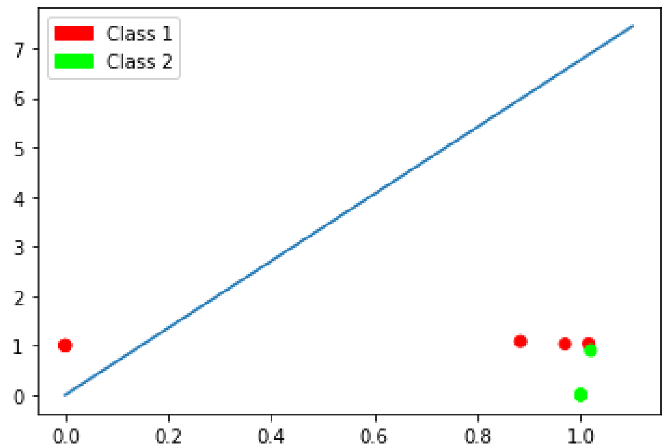

Class 1
Class 2

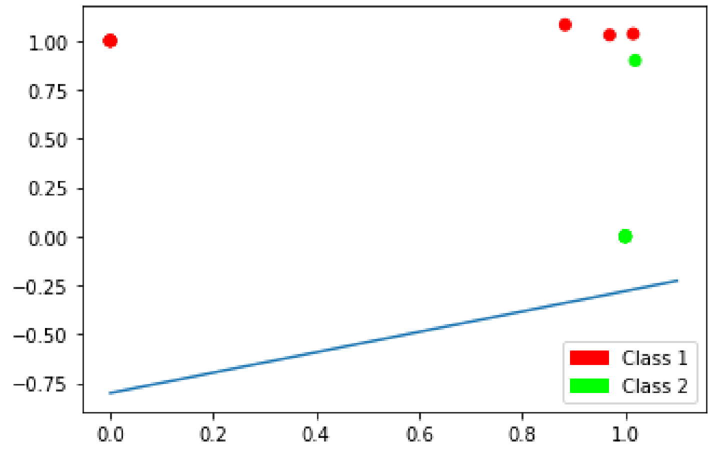

Class 1
Class 2

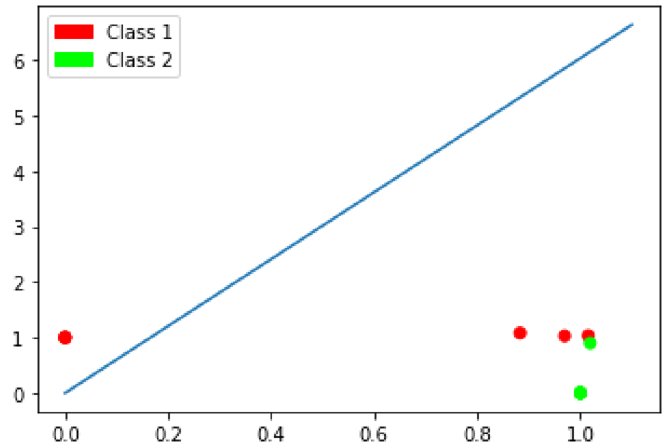

1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
0.0
0.2
0.4
0.6
0.8
1.0
Class 1
Class 2

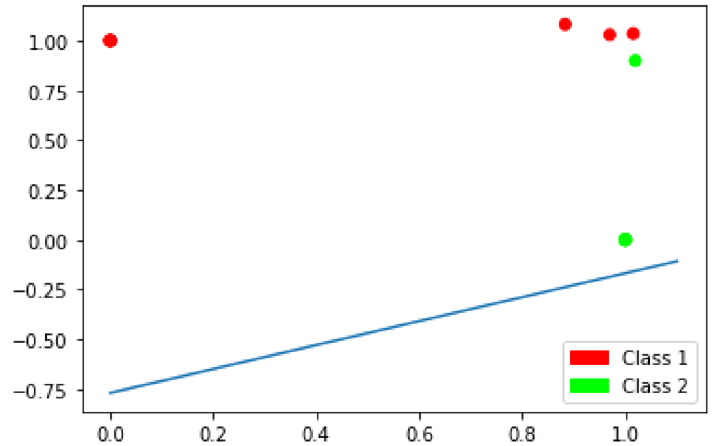

8
6
4
2
0
0.0
0.2
0.4
0.6
0.8
1.0
Class 1
Class 2

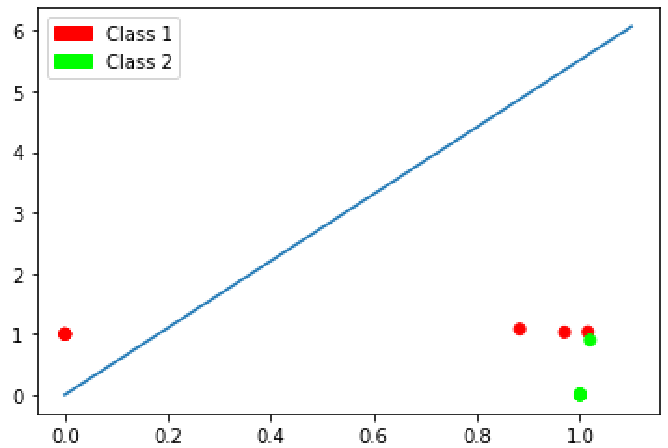

1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
0.0
0.2
0.4
0.6
0.8
1.0
Class 1
Class 2

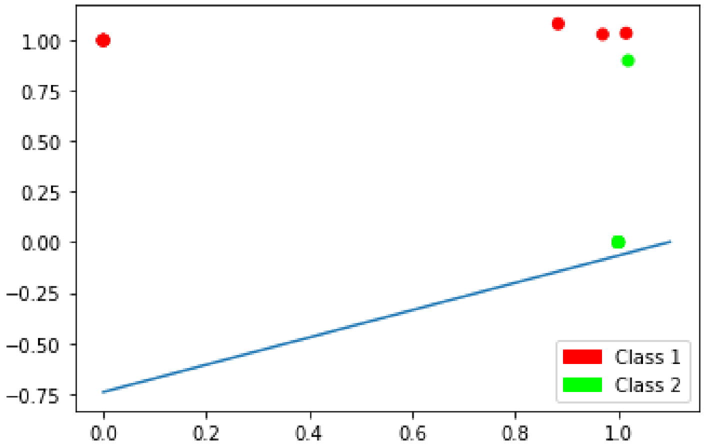

7
6
5
4
3
2
1
0
0.0
0.2
0.4
0.6
0.8
1.0
Class 1
Class 2

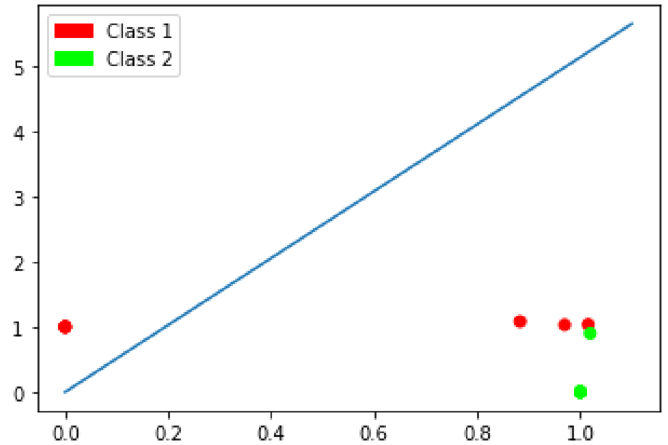

Class 1
Class 2
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
0.0
0.2
0.4
0.6
0.8
1.0

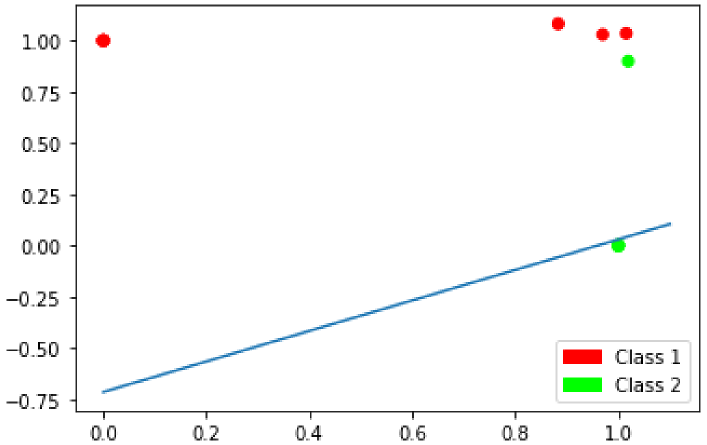

Class 1
Class 2
6
5
4
3
2
1
0
0.0
0.2
0.4
0.6
0.8
1.0

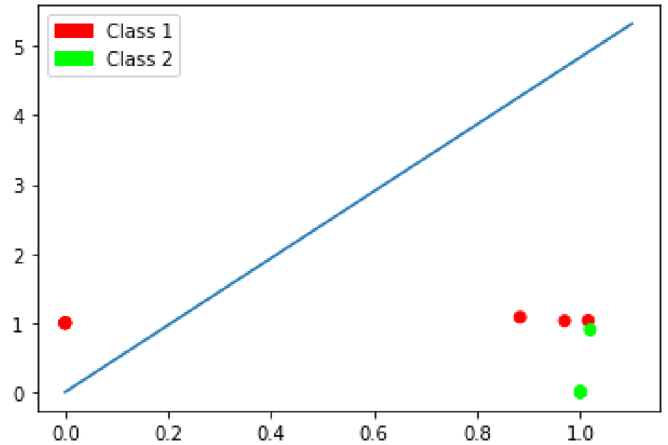

Class 1
Class 2
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
0.0
0.2
0.4
0.6
0.8
1.0

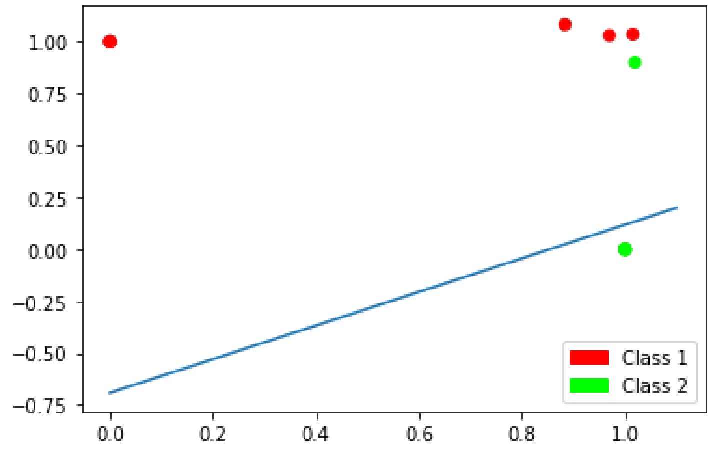

Class 1
Class 2
6
5
4
3
2
1
0
0.0
0.2
0.4
0.6
0.8
1.0

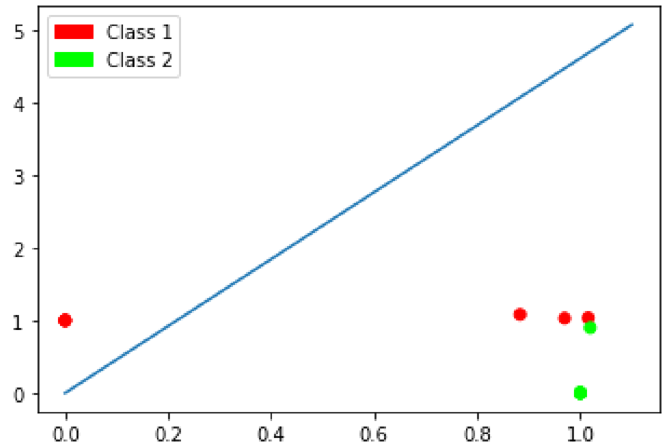

1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
0.0
0.2
0.4
0.6
0.8
1.0
Class 1
Class 2

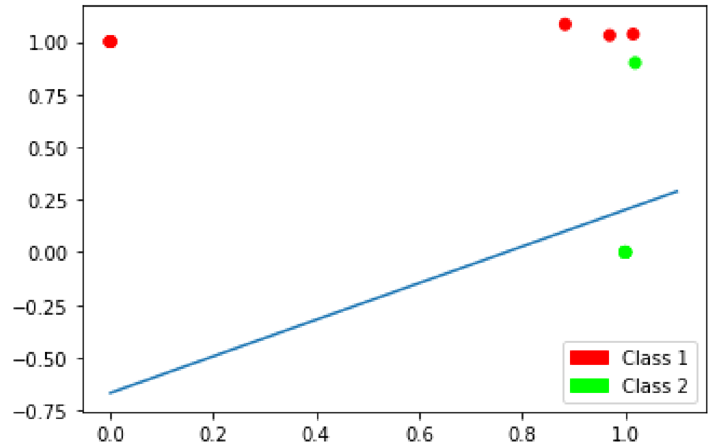

Class 1
Class 2
5
4
3
2
1
0
0.0
0.2
0.4
0.6
0.8
1.0

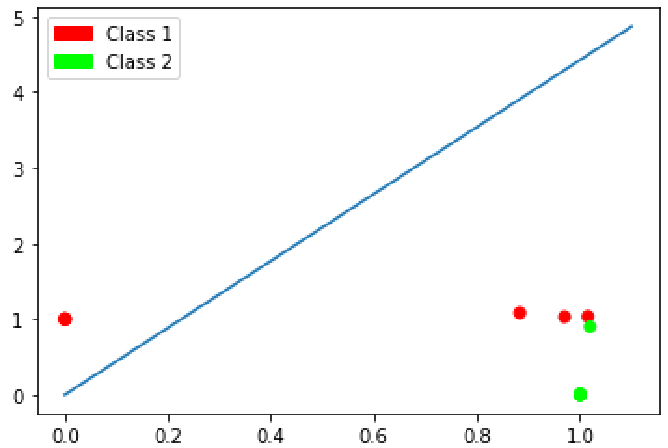

1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
0.0
0.2
0.4
0.6
0.8
1.0
Class 1
Class 2

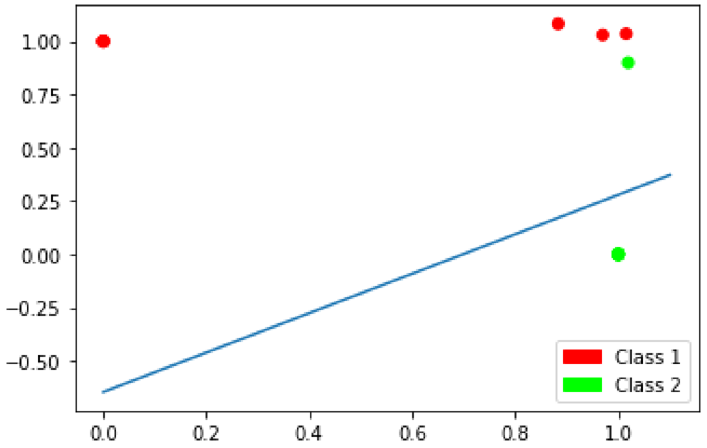

Class 1
Class 2
5
4
3
2
1
0
0.0
0.2
0.4
0.6
0.8
1.0

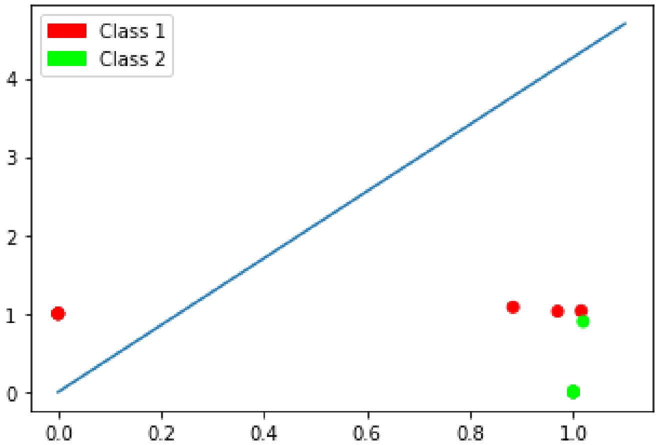

Class 1
Class 2

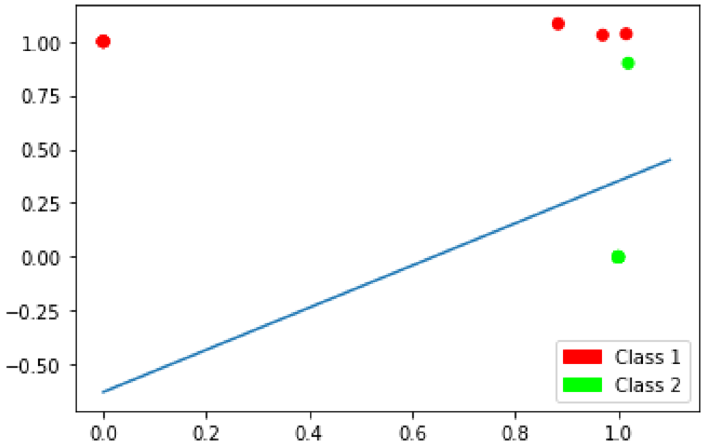

Class 1
Class 2

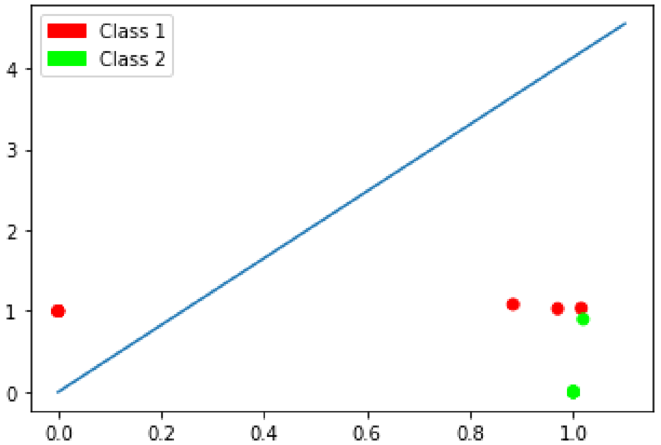

Class 1
Class 2

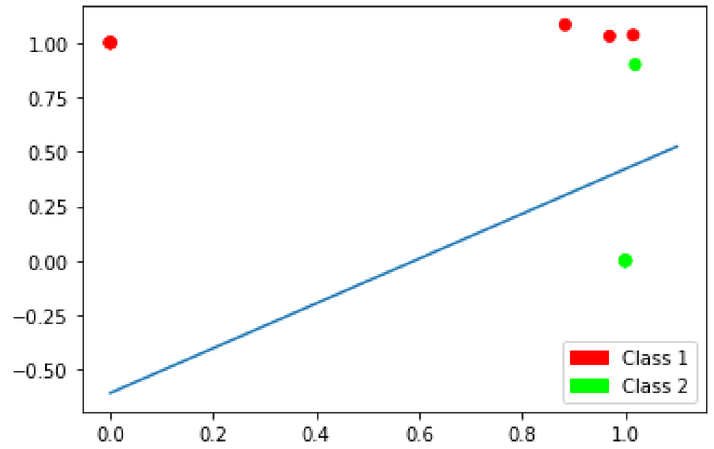

Class 1
Class 2

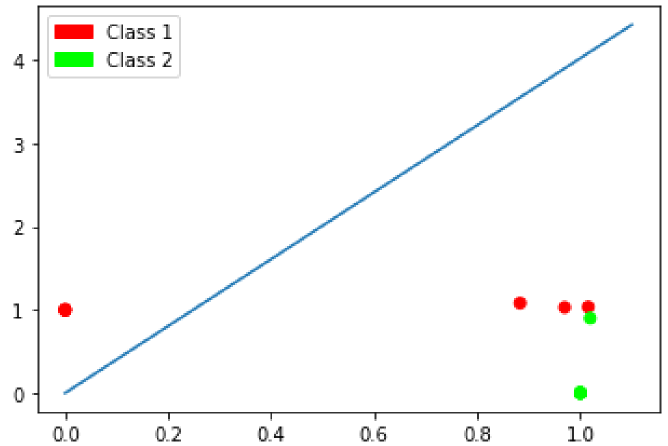

Class 1
Class 2

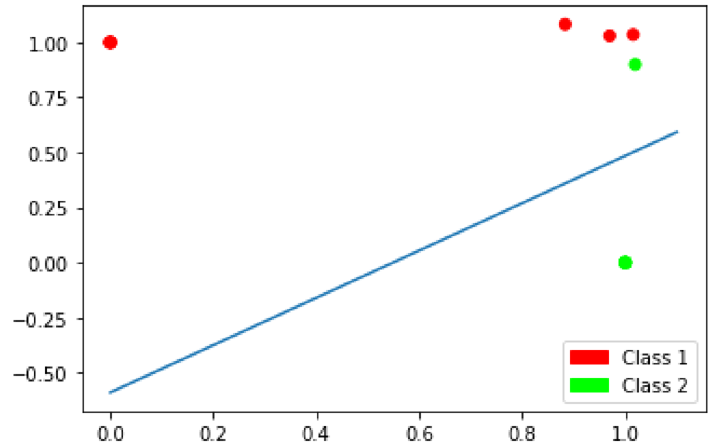

Class 1
Class 2

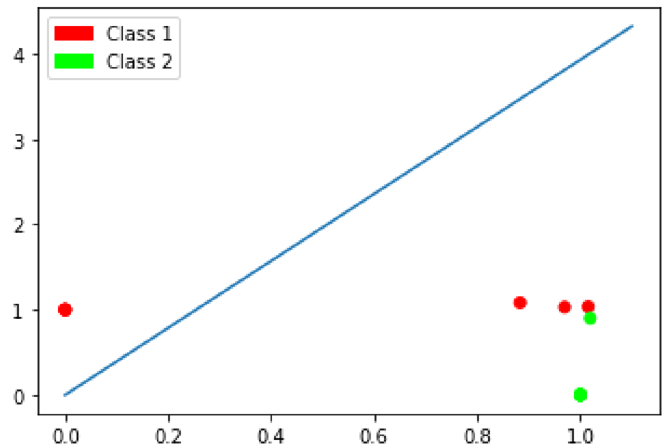

Class 1
Class 2

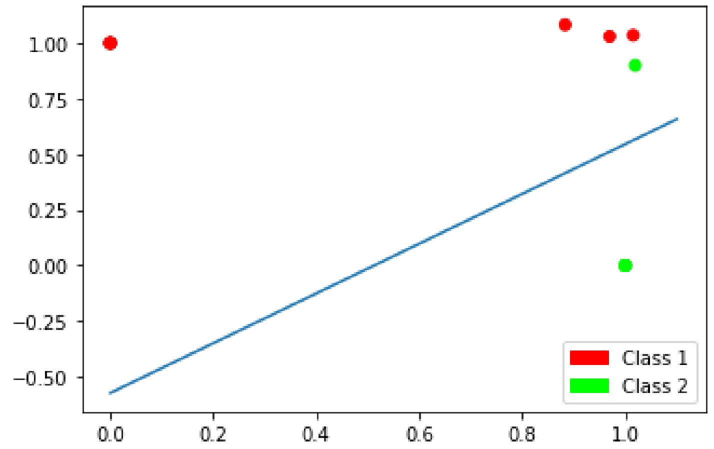

Class 1
Class 2

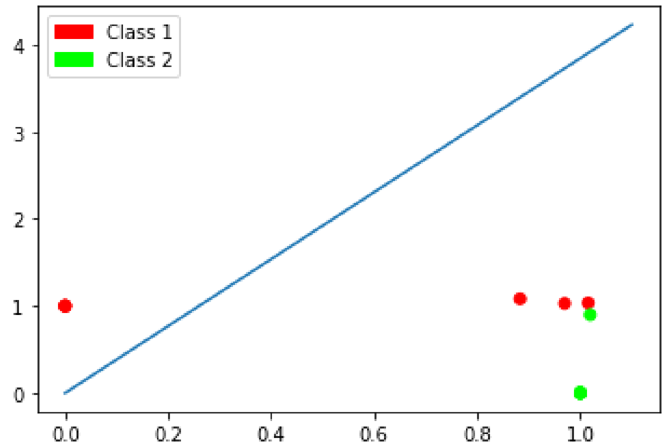

Class 1
Class 2

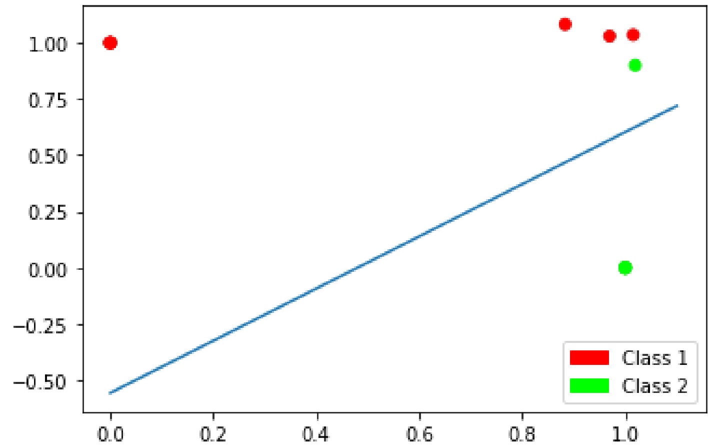

Class 1
Class 2

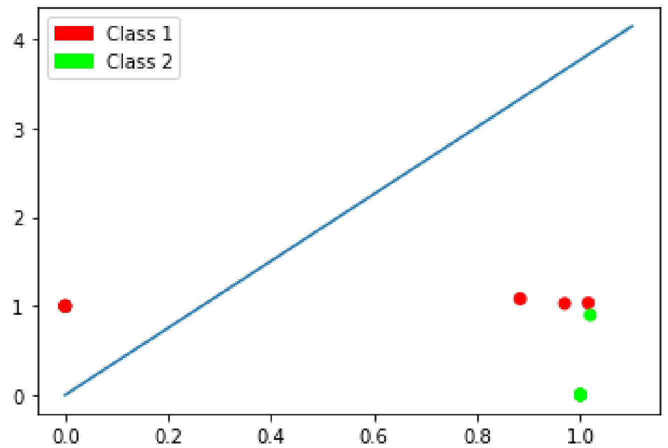

Class 1
Class 2

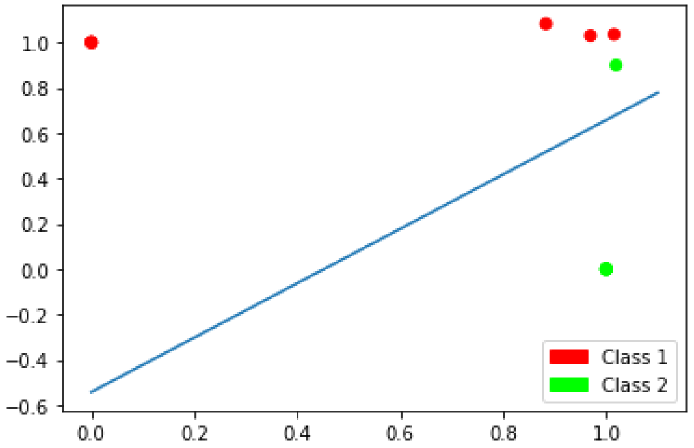

Class 1
Class 2

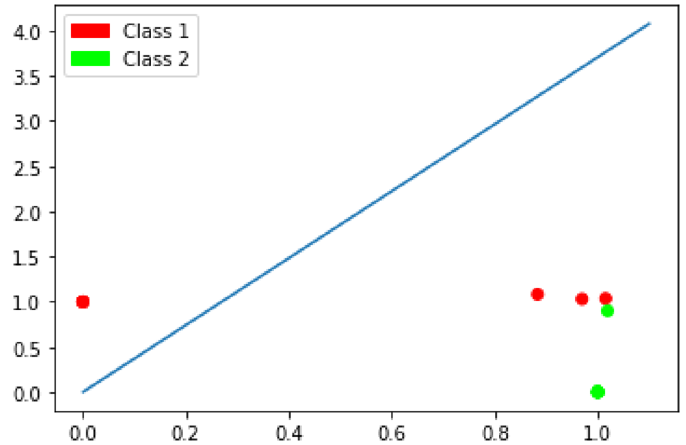

Class 1
Class 2

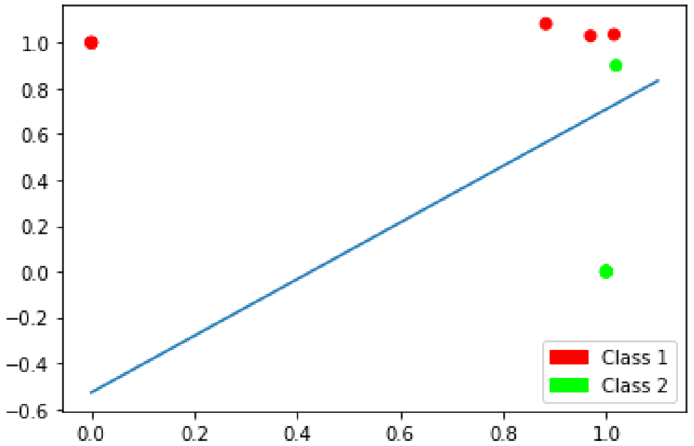

Class 1
Class 2

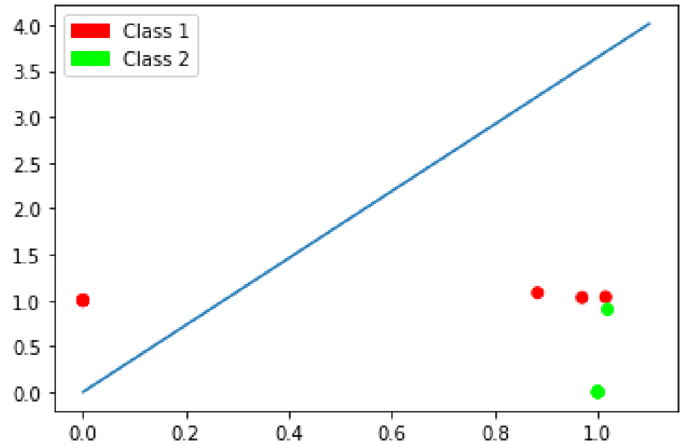

Class 1
Class 2

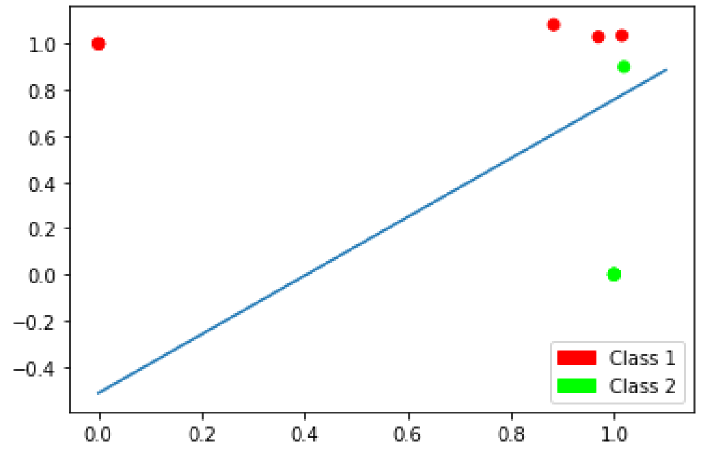

Class 1
Class 2

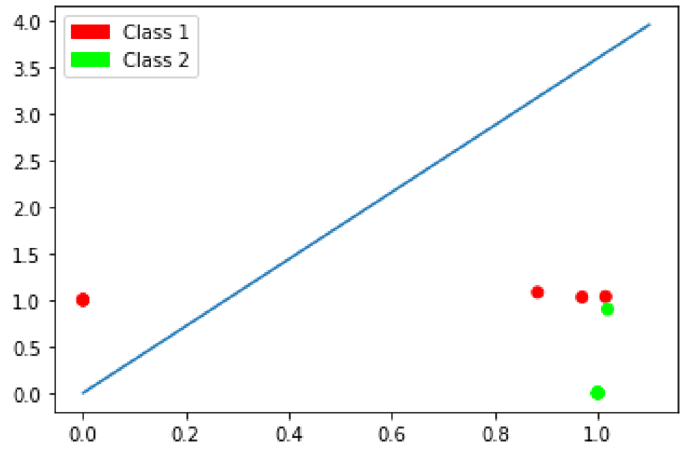

Class 1
Class 2

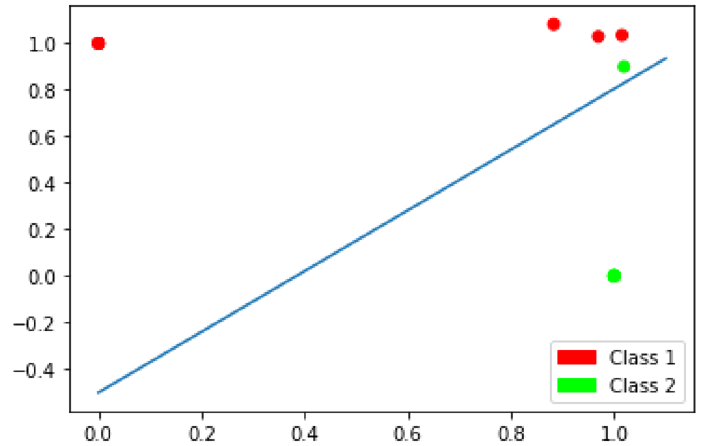

Class 1
Class 2

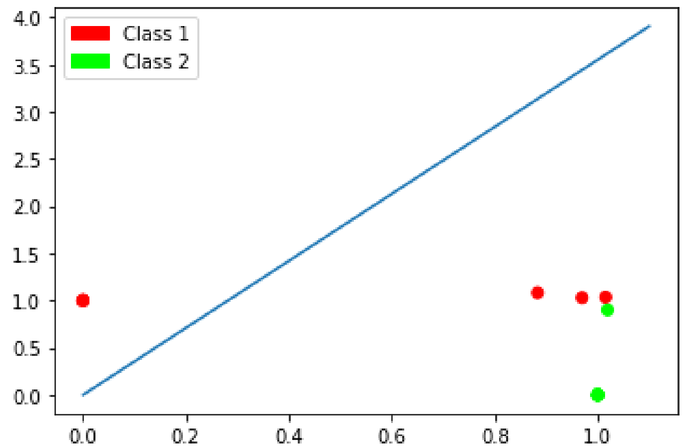

Class 1
Class 2

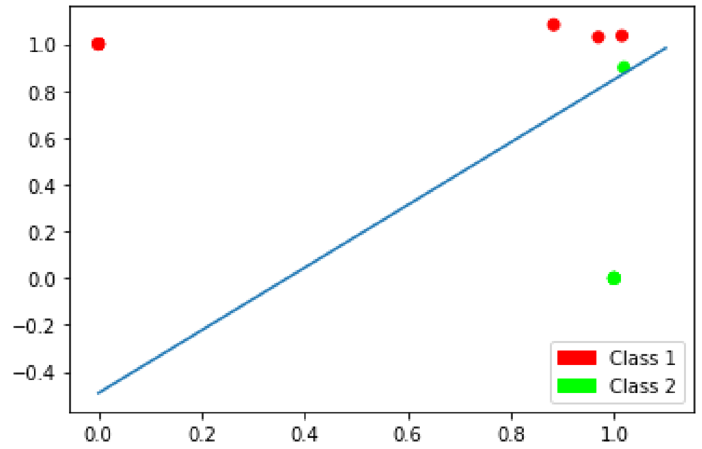

Class 1
Class 2

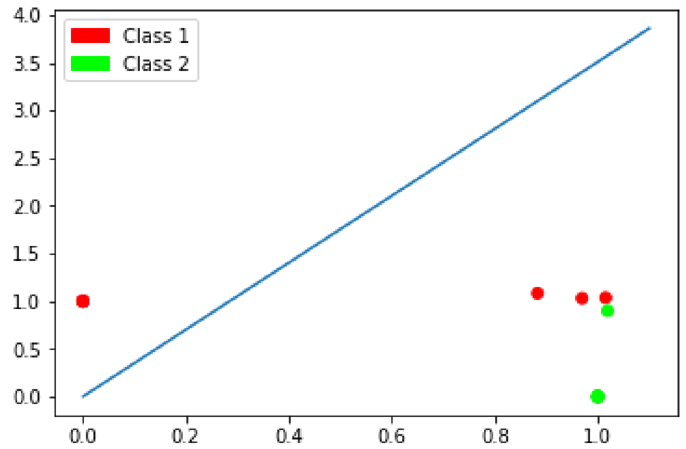

Class 1
Class 2

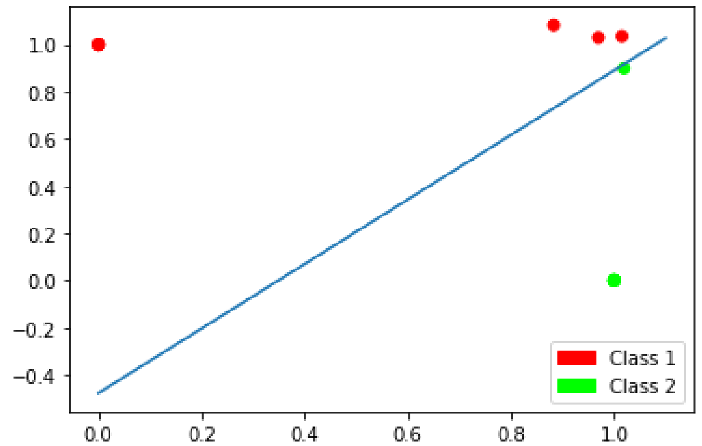

Class 1
Class 2

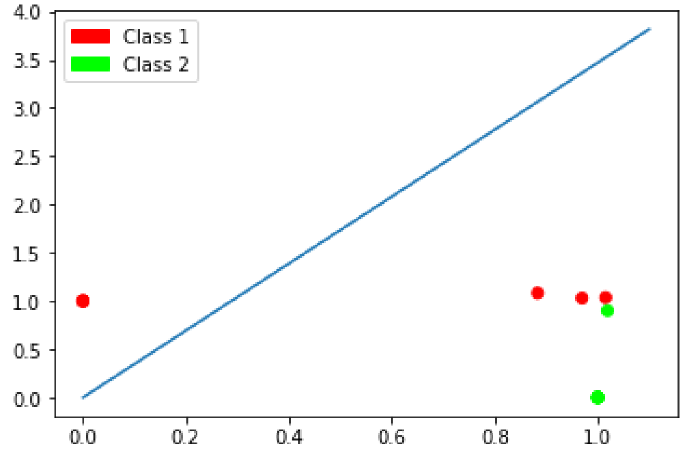

Class 1
Class 2

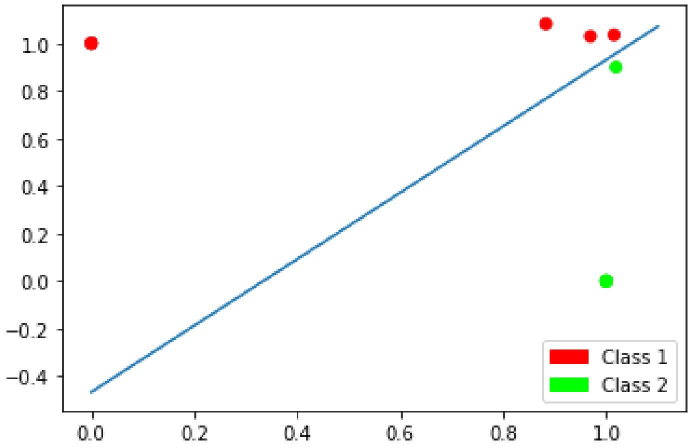

Class 1
Class 2

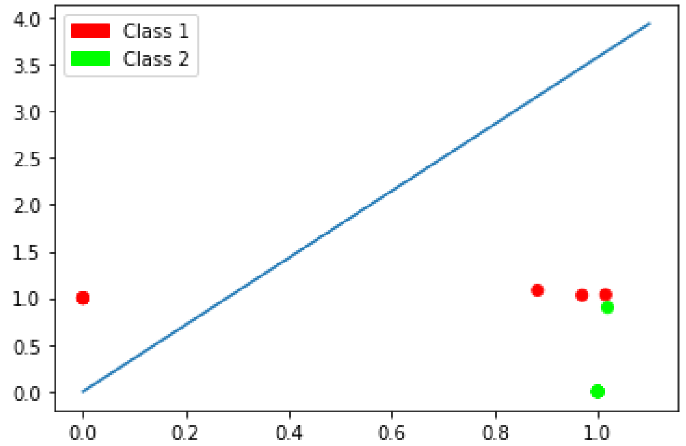

Class 1
Class 2

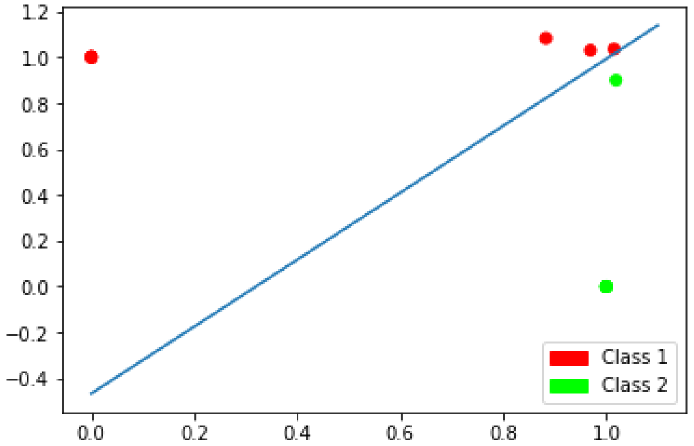

Class 1
Class 2


Class 1
Class 2


Class 1
Class 2







SVM

Defining the necessary functions used

In [ ]:

```
def SVM(C1, C2):
  x1 = np.array(C1.to_numpy())
  x2 = np.array(C2.to_numpy())

  x = np.vstack((x1, x2))
  y = np.hstack((np.ones(len(x1)), np.ones(len(x2)) * -1))

  X = np.dot(x, np.transpose(x))
  Y = np.outer(y, y)

  n = X.shape[0]

  H = matrix(Y * X)
  f = matrix(np.ones(n) * -1)
  A = matrix(np.diag(np.ones(n) * -1))
```

```
    a = matrix(np.zeros(n))
    B = matrix(y, (1, n))
    b = matrix(0.0)

    # solve QP problem
    solution = solvers.qp(H, f, A, a, B, b)
    print('')

    # Lagrange multipliers
    t1 = np.ravel(solution['x'])

    # Support vectors have non zero lagrange multipliers
    for i in range(0, len(t1)):
      if(t1[i] < 1e-4):
        t1[i] = 0.0
      else:
        t1[i] = round(t1[i], 4)

    t1 = np.array(t1)
    print(t1)

    # Weight vector
    w = np.zeros(X.shape[1])
    for i in range(X.shape[1]):
      w[i] = t1[i] * y[i]

    w = np.dot(w, x)

    # Intercept
    b = 0
    for i in range(len(t1)):
      if(t1[i] != 0.0):
        b = round(1 - np.dot(w, np.transpose(x[i])), 4)
        break

    return w, b
```

Code

In [ ]:

```
X = []

for img in ls:
    r, g, b = cv.split(img)
    X.append([np.sum(r)/np.sum(r+g+b), np.sum(g)/np.sum(r+g+b)])

C1 = pd.DataFrame(X)
C1
```

Out[ ]:

	0	1
0	0.969622	1.029586
1	0.883140	1.082019
2	0.000000	1.000000
3	0.000000	1.000000
4	0.000000	1.000000
5	0.000000	1.000000
6	1.015281	1.035262

In [ ]:

```
Y = []

for img in ls2:
```

```
    r, g, b = cv.split(img)
    Y.append([np.sum(r)/np.sum(r+g+b), np.sum(g)/np.sum(r+g+b)])

C2 = pd.DataFrame(Y)
C2
```

Out[ ]:

	0	1
0	1.000000	0.000000
1	1.000000	0.000000
2	1.000000	0.000000
3	1.000000	0.000000
4	1.019274	0.899458
5	1.000000	0.000000
6	1.000000	0.000000

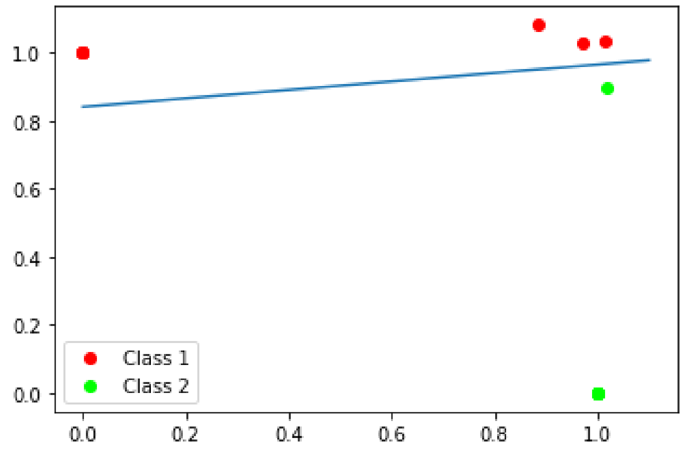
In [ ]:

```
w, b = SVM(C1, C2)


print(w)
print('')
print(b)
```

```
     pcost       dcost       gap    pres   dres
 0: -3.5318e+00 -1.0519e+01  7e+00  8e-16  2e+00
 1: -6.4286e+00 -8.1498e+00  2e+00  8e-17  9e-01
 2: -3.7324e+01 -4.2474e+01  5e+00  4e-15  9e-01
 3: -7.6169e+01 -9.3636e+01  2e+01  4e-15  7e-01
 4: -1.0565e+02 -1.2390e+02  2e+01  2e-14  3e-01
 5: -1.0930e+02 -1.1091e+02  2e+00  2e-14  2e-02
 6: -1.0932e+02 -1.0934e+02  2e-02  2e-14  2e-04
 7: -1.0932e+02 -1.0932e+02  2e-04  1e-14  2e-06
 8: -1.0932e+02 -1.0932e+02  2e-06  3e-14  2e-08
Optimal solution found.

[ 30.3932   0.       0.       0.       0.       0.      78.9254   0.
   0.       0.       0.     109.3187   0.       0.    ]
[-1.82424974 14.67334115]

-12.3386
```

In [ ]:

```
x1 = np.arange(0, 1.2, 0.1)

plt.scatter(C1.iloc[:, [0]], C1.iloc[:, [1]], label = 'Class 1', color = '#FF0000')
plt.scatter(C2.iloc[:, [0]], C2.iloc[:, [1]], label = 'Class 2', color = '#00FF00')
plt.legend()

slope, c = -w[0] / w[1], -b / w[1]
plt.plot(x1, slope * x1 + c)
plt.show()
```


Class 1
Class 2
0.0
0.2
0.4
0.6
0.8
1.0

Q4. From the iris dataset, choose the 'petal length', 'sepal width' for setosa, versicolor and virginica flowers. Learn a decision boundary for the two features using a **single perceptron and SVM**. Assume that all the weights of the perceptron are initialized as 0 with the learning rate of 0.01. Draw the decision boundary.

Importing the necessary libraries

In [ ]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from cvxopt import matrix, solvers

from matplotlib.colors import ListedColormap
import matplotlib.patches as mpatches
```

Iris Dataset

In [ ]:

```
from google.colab import files
uploaded = files.upload()
```

Choose files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving Iris_dataset.csv to Iris_dataset.csv
```

In [ ]:

```
df = pd.read_csv('Iris_dataset.csv')
df
```

Out[ ]:

	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	4.6	3.1	1.5	0.2	Setosa
4	5.0	3.6	1.4	0.2	Setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Virginica
146	6.3	2.5	5.0	1.9	Virginica
147	6.5	3.0	5.2	2.0	Virginica
148	6.2	3.4	5.4	2.3	Virginica
149	5.9	3.0	5.1	1.8	Virginica

150 rows × 5 columns

In [ ]:

```
df = df.drop(['sepal.length', 'petal.width'], axis = 1)
df
```

Out[ ]:

	sepal.width	petal.length	variety
0	3.5	1.4	Setosa
1	3.0	1.4	Setosa

	sepal.width	petal.length	variety
2	3.2	1.3	Setosa
3	3.1	1.5	Setosa
4	3.6	1.4	Setosa
...	...	...	...
145	3.0	5.2	Virginica
146	2.5	5.0	Virginica
147	3.0	5.2	Virginica
148	3.4	5.4	Virginica
149	3.0	5.1	Virginica

150 rows × 3 columns

Single Perceptron

Defining the necessary functions used

In [ ]:

```
def perceptron_training_alg(X, Y, learning_rate):
  # Adding Bias to the input and Randomly initializing weights
  num_samples, num_features = X.shape

  X = np.hstack((np.ones((num_samples, 1)), X))
  W = np.zeros(num_features + 1).reshape((-1, 1))

  weight_vector = []
  weight_vector.append(W)

  print(W, "\n")

  # Running the algorithm until all the data points are correctly classified
  samples_misclassified = True
  cnt=0
  while samples_misclassified and cnt < 100000:
    # Initially we'll assume that there are no misclassified samples
    samples_misclassified = False

    for index, x in enumerate(X):
      if cnt >= 100000:
        return W, np.array(weight_vector)

      x = Y[index] * x

      if np.dot(x.reshape((1, -1)), W) <= 0: # if samples misclassified
        samples_misclassified = True
        W = W + (learning_rate * x.reshape((-1, 1)))     # Gradient Descent Step
        weight_vector.append(W)

        print(W, "\n")
        cnt += 1

  return W, np.array(weight_vector)
```

In [ ]:

```
def sorter(X, Y):
  X_new0 = []
  Y_new0 = []

  X_new1 = []
  Y_new1 = []
```

```
  for x, y in zip(X, Y):
    if y == 0:
      Y_new0.append(y)
      X_new0.append(x)

    else:
      Y_new1.append(y)
      X_new1.append(x)

  X0 = []
  Y0 = []

  X1 = []
  Y1 = []

  if len(X_new0) > 0:
    X0 = np.stack(X_new0)
    Y0 = np.stack(Y_new0)

  elif len(X_new1) > 0:
    X1 = np.stack(X_new1)
    Y1 = np.stack(Y_new1)

  if len(X0) > 0 and len(X1) > 0:
    X = np.concatenate((X0, X1))
    Y = np.concatenate((Y0, Y1))

  elif len(X0) > 0:
    X = X0
    Y = Y0

  else:
    X = X1
    Y = Y1

  return X, Y
```

In [ ]:

```
def solve_perceptron(clas, df):
    df_4 = df[['sepal.width', 'petal.length']]
    X1 = np.array(df_4)

    ls = []
    ls2 = []

    for x in df.itertuples():
        print(x)

        if x[-1] == clas:
            ls.append([1])
            ls2.append([1])

        else:
            ls.append([-1])
            ls2.append([0])

    Y1=np.array(ls)
    ans1=np.array(ls2)

    W, weight_vector = perceptron_training_alg(X1, Y1, 0.01)
    return W, X1, ans1, weight_vector
```

Code

In [ ]:

```
df_1 = df.loc[df.variety == 'Setosa']
print(df_1.head(5))
```

```
print(df_1.tail(5))
df_1.shape
```

```
   sepal.width  petal.length variety
0          3.5           1.4  Setosa
1          3.0           1.4  Setosa
2          3.2           1.3  Setosa
3          3.1           1.5  Setosa
4          3.6           1.4  Setosa
    sepal.width  petal.length variety
45          3.0           1.4  Setosa
46          3.8           1.6  Setosa
47          3.2           1.4  Setosa
48          3.7           1.5  Setosa
49          3.3           1.4  Setosa
```

Out[ ]: (50, 3)

In [ ]:

```
df_2 = df.loc[df.variety == 'Virginica']
print(df_2.head(5))
print(df_2.tail(5))
df_2.shape
```

```
     sepal.width  petal.length    variety
100          3.3           6.0  Virginica
101          2.7           5.1  Virginica
102          3.0           5.9  Virginica
103          2.9           5.6  Virginica
104          3.0           5.8  Virginica
     sepal.width  petal.length    variety
145          3.0           5.2  Virginica
146          2.5           5.0  Virginica
147          3.0           5.2  Virginica
148          3.4           5.4  Virginica
149          3.0           5.1  Virginica
```

Out[ ]: (50, 3)

In [ ]:

```
df_3 = df.loc[df.variety == 'Versicolor']
print(df_3.head(5))
print(df_3.tail(5))
df_3.shape
```

```
    sepal.width  petal.length     variety
50          3.2           4.7  Versicolor
51          3.2           4.5  Versicolor
52          3.1           4.9  Versicolor
53          2.3           4.0  Versicolor
54          2.8           4.6  Versicolor
    sepal.width  petal.length     variety
95          3.0           4.2  Versicolor
96          2.9           4.2  Versicolor
97          2.9           4.3  Versicolor
98          2.5           3.0  Versicolor
99          2.8           4.1  Versicolor
```

Out[ ]: (50, 3)

In [ ]:

```
W1, X1, ans1, weight_vector1 = solve_perceptron('Setosa', df)
```

```
Pandas(Index=0, _1=3.5, _2=1.4, variety='Setosa')
Pandas(Index=1, _1=3.0, _2=1.4, variety='Setosa')
Pandas(Index=2, _1=3.2, _2=1.3, variety='Setosa')
Pandas(Index=3, _1=3.1, _2=1.5, variety='Setosa')
Pandas(Index=4, _1=3.6, _2=1.4, variety='Setosa')
Pandas(Index=5, _1=3.9, _2=1.7, variety='Setosa')
Pandas(Index=6, _1=3.4, _2=1.4, variety='Setosa')
Pandas(Index=7, _1=3.4, _2=1.5, variety='Setosa')
Pandas(Index=8, _1=2.9, _2=1.4, variety='Setosa')
Pandas(Index=9, _1=3.1, _2=1.5, variety='Setosa')
Pandas(Index=10, _1=3.7, _2=1.5, variety='Setosa')
```

```
Pandas(Index=11, _1=3.4, _2=1.6, variety='Setosa')
Pandas(Index=12, _1=3.0, _2=1.4, variety='Setosa')
Pandas(Index=13, _1=3.0, _2=1.1, variety='Setosa')
Pandas(Index=14, _1=4.0, _2=1.2, variety='Setosa')
Pandas(Index=15, _1=4.4, _2=1.5, variety='Setosa')
Pandas(Index=16, _1=3.9, _2=1.3, variety='Setosa')
Pandas(Index=17, _1=3.5, _2=1.4, variety='Setosa')
Pandas(Index=18, _1=3.8, _2=1.7, variety='Setosa')
Pandas(Index=19, _1=3.8, _2=1.5, variety='Setosa')
Pandas(Index=20, _1=3.4, _2=1.7, variety='Setosa')
Pandas(Index=21, _1=3.7, _2=1.5, variety='Setosa')
Pandas(Index=22, _1=3.6, _2=1.0, variety='Setosa')
Pandas(Index=23, _1=3.3, _2=1.7, variety='Setosa')
Pandas(Index=24, _1=3.4, _2=1.9, variety='Setosa')
Pandas(Index=25, _1=3.0, _2=1.6, variety='Setosa')
Pandas(Index=26, _1=3.4, _2=1.6, variety='Setosa')
Pandas(Index=27, _1=3.5, _2=1.5, variety='Setosa')
Pandas(Index=28, _1=3.4, _2=1.4, variety='Setosa')
Pandas(Index=29, _1=3.2, _2=1.6, variety='Setosa')
Pandas(Index=30, _1=3.1, _2=1.6, variety='Setosa')
Pandas(Index=31, _1=3.4, _2=1.5, variety='Setosa')
Pandas(Index=32, _1=4.1, _2=1.5, variety='Setosa')
Pandas(Index=33, _1=4.2, _2=1.4, variety='Setosa')
Pandas(Index=34, _1=3.1, _2=1.5, variety='Setosa')
Pandas(Index=35, _1=3.2, _2=1.2, variety='Setosa')
Pandas(Index=36, _1=3.5, _2=1.3, variety='Setosa')
Pandas(Index=37, _1=3.6, _2=1.4, variety='Setosa')
Pandas(Index=38, _1=3.0, _2=1.3, variety='Setosa')
Pandas(Index=39, _1=3.4, _2=1.5, variety='Setosa')
Pandas(Index=40, _1=3.5, _2=1.3, variety='Setosa')
Pandas(Index=41, _1=2.3, _2=1.3, variety='Setosa')
Pandas(Index=42, _1=3.2, _2=1.3, variety='Setosa')
Pandas(Index=43, _1=3.5, _2=1.6, variety='Setosa')
Pandas(Index=44, _1=3.8, _2=1.9, variety='Setosa')
Pandas(Index=45, _1=3.0, _2=1.4, variety='Setosa')
Pandas(Index=46, _1=3.8, _2=1.6, variety='Setosa')
Pandas(Index=47, _1=3.2, _2=1.4, variety='Setosa')
Pandas(Index=48, _1=3.7, _2=1.5, variety='Setosa')
Pandas(Index=49, _1=3.3, _2=1.4, variety='Setosa')
Pandas(Index=50, _1=3.2, _2=4.7, variety='Versicolor')
Pandas(Index=51, _1=3.2, _2=4.5, variety='Versicolor')
Pandas(Index=52, _1=3.1, _2=4.9, variety='Versicolor')
Pandas(Index=53, _1=2.3, _2=4.0, variety='Versicolor')
Pandas(Index=54, _1=2.8, _2=4.6, variety='Versicolor')
Pandas(Index=55, _1=2.8, _2=4.5, variety='Versicolor')
Pandas(Index=56, _1=3.3, _2=4.7, variety='Versicolor')
Pandas(Index=57, _1=2.4, _2=3.3, variety='Versicolor')
Pandas(Index=58, _1=2.9, _2=4.6, variety='Versicolor')
Pandas(Index=59, _1=2.7, _2=3.9, variety='Versicolor')
Pandas(Index=60, _1=2.0, _2=3.5, variety='Versicolor')
Pandas(Index=61, _1=3.0, _2=4.2, variety='Versicolor')
Pandas(Index=62, _1=2.2, _2=4.0, variety='Versicolor')
Pandas(Index=63, _1=2.9, _2=4.7, variety='Versicolor')
Pandas(Index=64, _1=2.9, _2=3.6, variety='Versicolor')
Pandas(Index=65, _1=3.1, _2=4.4, variety='Versicolor')
Pandas(Index=66, _1=3.0, _2=4.5, variety='Versicolor')
Pandas(Index=67, _1=2.7, _2=4.1, variety='Versicolor')
Pandas(Index=68, _1=2.2, _2=4.5, variety='Versicolor')
Pandas(Index=69, _1=2.5, _2=3.9, variety='Versicolor')
Pandas(Index=70, _1=3.2, _2=4.8, variety='Versicolor')
Pandas(Index=71, _1=2.8, _2=4.0, variety='Versicolor')
Pandas(Index=72, _1=2.5, _2=4.9, variety='Versicolor')
Pandas(Index=73, _1=2.8, _2=4.7, variety='Versicolor')
Pandas(Index=74, _1=2.9, _2=4.3, variety='Versicolor')
Pandas(Index=75, _1=3.0, _2=4.4, variety='Versicolor')
Pandas(Index=76, _1=2.8, _2=4.8, variety='Versicolor')
Pandas(Index=77, _1=3.0, _2=5.0, variety='Versicolor')
Pandas(Index=78, _1=2.9, _2=4.5, variety='Versicolor')
Pandas(Index=79, _1=2.6, _2=3.5, variety='Versicolor')
Pandas(Index=80, _1=2.4, _2=3.8, variety='Versicolor')
Pandas(Index=81, _1=2.4, _2=3.7, variety='Versicolor')
Pandas(Index=82, _1=2.7, _2=3.9, variety='Versicolor')
Pandas(Index=83, _1=2.7, _2=5.1, variety='Versicolor')
Pandas(Index=84, _1=3.0, _2=4.5, variety='Versicolor')
Pandas(Index=85, _1=3.4, _2=4.5, variety='Versicolor')
```

```
Pandas(Index=86, _1=3.1, _2=4.7, variety='Versicolor')
Pandas(Index=87, _1=2.3, _2=4.4, variety='Versicolor')
Pandas(Index=88, _1=3.0, _2=4.1, variety='Versicolor')
Pandas(Index=89, _1=2.5, _2=4.0, variety='Versicolor')
Pandas(Index=90, _1=2.6, _2=4.4, variety='Versicolor')
Pandas(Index=91, _1=3.0, _2=4.6, variety='Versicolor')
Pandas(Index=92, _1=2.6, _2=4.0, variety='Versicolor')
Pandas(Index=93, _1=2.3, _2=3.3, variety='Versicolor')
Pandas(Index=94, _1=2.7, _2=4.2, variety='Versicolor')
Pandas(Index=95, _1=3.0, _2=4.2, variety='Versicolor')
Pandas(Index=96, _1=2.9, _2=4.2, variety='Versicolor')
Pandas(Index=97, _1=2.9, _2=4.3, variety='Versicolor')
Pandas(Index=98, _1=2.5, _2=3.0, variety='Versicolor')
Pandas(Index=99, _1=2.8, _2=4.1, variety='Versicolor')
Pandas(Index=100, _1=3.3, _2=6.0, variety='Virginica')
Pandas(Index=101, _1=2.7, _2=5.1, variety='Virginica')
Pandas(Index=102, _1=3.0, _2=5.9, variety='Virginica')
Pandas(Index=103, _1=2.9, _2=5.6, variety='Virginica')
Pandas(Index=104, _1=3.0, _2=5.8, variety='Virginica')
Pandas(Index=105, _1=3.0, _2=6.6, variety='Virginica')
Pandas(Index=106, _1=2.5, _2=4.5, variety='Virginica')
Pandas(Index=107, _1=2.9, _2=6.3, variety='Virginica')
Pandas(Index=108, _1=2.5, _2=5.8, variety='Virginica')
Pandas(Index=109, _1=3.6, _2=6.1, variety='Virginica')
Pandas(Index=110, _1=3.2, _2=5.1, variety='Virginica')
Pandas(Index=111, _1=2.7, _2=5.3, variety='Virginica')
Pandas(Index=112, _1=3.0, _2=5.5, variety='Virginica')
Pandas(Index=113, _1=2.5, _2=5.0, variety='Virginica')
Pandas(Index=114, _1=2.8, _2=5.1, variety='Virginica')
Pandas(Index=115, _1=3.2, _2=5.3, variety='Virginica')
Pandas(Index=116, _1=3.0, _2=5.5, variety='Virginica')
Pandas(Index=117, _1=3.8, _2=6.7, variety='Virginica')
Pandas(Index=118, _1=2.6, _2=6.9, variety='Virginica')
Pandas(Index=119, _1=2.2, _2=5.0, variety='Virginica')
Pandas(Index=120, _1=3.2, _2=5.7, variety='Virginica')
Pandas(Index=121, _1=2.8, _2=4.9, variety='Virginica')
Pandas(Index=122, _1=2.8, _2=6.7, variety='Virginica')
Pandas(Index=123, _1=2.7, _2=4.9, variety='Virginica')
Pandas(Index=124, _1=3.3, _2=5.7, variety='Virginica')
Pandas(Index=125, _1=3.2, _2=6.0, variety='Virginica')
Pandas(Index=126, _1=2.8, _2=4.8, variety='Virginica')
Pandas(Index=127, _1=3.0, _2=4.9, variety='Virginica')
Pandas(Index=128, _1=2.8, _2=5.6, variety='Virginica')
Pandas(Index=129, _1=3.0, _2=5.8, variety='Virginica')
Pandas(Index=130, _1=2.8, _2=6.1, variety='Virginica')
Pandas(Index=131, _1=3.8, _2=6.4, variety='Virginica')
Pandas(Index=132, _1=2.8, _2=5.6, variety='Virginica')
Pandas(Index=133, _1=2.8, _2=5.1, variety='Virginica')
Pandas(Index=134, _1=2.6, _2=5.6, variety='Virginica')
Pandas(Index=135, _1=3.0, _2=6.1, variety='Virginica')
Pandas(Index=136, _1=3.4, _2=5.6, variety='Virginica')
Pandas(Index=137, _1=3.1, _2=5.5, variety='Virginica')
Pandas(Index=138, _1=3.0, _2=4.8, variety='Virginica')
Pandas(Index=139, _1=3.1, _2=5.4, variety='Virginica')
Pandas(Index=140, _1=3.1, _2=5.6, variety='Virginica')
Pandas(Index=141, _1=3.1, _2=5.1, variety='Virginica')
Pandas(Index=142, _1=2.7, _2=5.1, variety='Virginica')
Pandas(Index=143, _1=3.2, _2=5.9, variety='Virginica')
Pandas(Index=144, _1=3.3, _2=5.7, variety='Virginica')
Pandas(Index=145, _1=3.0, _2=5.2, variety='Virginica')
Pandas(Index=146, _1=2.5, _2=5.0, variety='Virginica')
Pandas(Index=147, _1=3.0, _2=5.2, variety='Virginica')
Pandas(Index=148, _1=3.4, _2=5.4, variety='Virginica')
Pandas(Index=149, _1=3.0, _2=5.1, variety='Virginica')
[[0.]
 [0.]
 [0.]]

[[0.01 ]
 [0.035]
 [0.014]]

[[ 0.   ]
 [ 0.003]
 [-0.033]]
```

```
[[ 0.01 ]
 [ 0.038]
 [-0.019]]

[[ 0.   ]
 [ 0.006]
 [-0.066]]

[[ 0.01 ]
 [ 0.041]
 [-0.052]]
```

In [ ]:
```
W2, X2, ans2, weight_vector2 = solve_perceptron('Virginica', df)
```

```
Streaming output truncated to the last 5000 lines.
[[-3.81 ]
 [-0.865]
 [ 1.331]]

[[-3.8  ]
 [-0.835]
 [ 1.379]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.82 ]
 [-0.898]
 [ 1.283]]

[[-3.83 ]
 [-0.923]
 [ 1.234]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.81 ]
 [-0.871]
 [ 1.33 ]]

[[-3.8  ]
 [-0.841]
 [ 1.378]]

[[-3.81 ]
 [-0.872]
 [ 1.329]]

[[-3.82 ]
 [-0.894]
 [ 1.284]]

[[-3.83 ]
 [-0.919]
 [ 1.235]]

[[-3.82 ]
 [-0.892]
 [ 1.286]]

[[-3.81 ]
 [-0.867]
 [ 1.331]]

[[-3.8  ]
 [-0.837]
 [ 1.379]]

[[-3.81 ]
```

```
 [-0.869]
 [ 1.332]]

[[-3.82 ]
 [-0.9  ]
 [ 1.283]]

[[-3.83 ]
 [-0.925]
 [ 1.234]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.8  ]
 [-0.843]
 [ 1.378]]

[[-3.81 ]
 [-0.874]
 [ 1.329]]

[[-3.82 ]
 [-0.896]
 [ 1.284]]

[[-3.83 ]
 [-0.921]
 [ 1.235]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.81 ]
 [-0.869]
 [ 1.331]]

[[-3.8  ]
 [-0.839]
 [ 1.379]]

[[-3.81]
 [-0.87]
 [ 1.33]]

[[-3.82 ]
 [-0.892]
 [ 1.285]]

[[-3.83 ]
 [-0.917]
 [ 1.236]]

[[-3.82 ]
 [-0.89 ]
 [ 1.287]]

[[-3.81 ]
 [-0.865]
 [ 1.332]]

[[-3.8  ]
 [-0.835]
 [ 1.38 ]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]
```

```
[[-3.82 ]
 [-0.898]
 [ 1.284]]

[[-3.83 ]
 [-0.923]
 [ 1.235]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.8  ]
 [-0.841]
 [ 1.379]]

[[-3.81 ]
 [-0.872]
 [ 1.33 ]]

[[-3.82 ]
 [-0.894]
 [ 1.285]]

[[-3.83 ]
 [-0.919]
 [ 1.236]]

[[-3.82 ]
 [-0.892]
 [ 1.287]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.8  ]
 [-0.837]
 [ 1.38 ]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.82 ]
 [-0.9  ]
 [ 1.284]]

[[-3.83 ]
 [-0.925]
 [ 1.235]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.81 ]
 [-0.873]
 [ 1.331]]

[[-3.8  ]
 [-0.843]
 [ 1.379]]

[[-3.81 ]
 [-0.874]
 [ 1.33 ]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]
```

```
[[-3.83 ]
 [-0.921]
 [ 1.236]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.8  ]
 [-0.839]
 [ 1.38 ]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.82 ]
 [-0.902]
 [ 1.284]]

[[-3.83 ]
 [-0.927]
 [ 1.235]]

[[-3.82 ]
 [-0.9  ]
 [ 1.286]]

[[-3.81 ]
 [-0.875]
 [ 1.331]]

[[-3.8  ]
 [-0.845]
 [ 1.379]]

[[-3.81 ]
 [-0.876]
 [ 1.33 ]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.83 ]
 [-0.923]
 [ 1.236]]

[[-3.82 ]
 [-0.896]
 [ 1.287]]

[[-3.81 ]
 [-0.871]
 [ 1.332]]

[[-3.8  ]
 [-0.841]
 [ 1.38 ]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.83 ]
 [-0.919]
```

```
 [ 1.237]]

[[-3.82 ]
 [-0.892]
 [ 1.288]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.8  ]
 [-0.837]
 [ 1.381]]

[[-3.81 ]
 [-0.869]
 [ 1.334]]

[[-3.82 ]
 [-0.9  ]
 [ 1.285]]

[[-3.83 ]
 [-0.925]
 [ 1.236]]

[[-3.82 ]
 [-0.898]
 [ 1.287]]

[[-3.81 ]
 [-0.873]
 [ 1.332]]

[[-3.8  ]
 [-0.843]
 [ 1.38 ]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.83 ]
 [-0.921]
 [ 1.237]]

[[-3.82 ]
 [-0.894]
 [ 1.288]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.8  ]
 [-0.839]
 [ 1.381]]

[[-3.81 ]
 [-0.871]
 [ 1.334]]

[[-3.82 ]
 [-0.902]
 [ 1.285]]

[[-3.83 ]
 [-0.927]
 [ 1.236]]

[[-3.82 ]
```

```
 [-0.9  ]
 [ 1.287]]

[[-3.81 ]
 [-0.875]
 [ 1.332]]

[[-3.8  ]
 [-0.845]
 [ 1.38 ]]

[[-3.81 ]
 [-0.876]
 [ 1.331]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.83 ]
 [-0.923]
 [ 1.237]]

[[-3.82 ]
 [-0.896]
 [ 1.288]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.8  ]
 [-0.841]
 [ 1.381]]

[[-3.81 ]
 [-0.872]
 [ 1.332]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.84 ]
 [-0.946]
 [ 1.187]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.82 ]
 [-0.894]
 [ 1.283]]

[[-3.81 ]
 [-0.862]
 [ 1.334]]

[[-3.82 ]
 [-0.893]
 [ 1.285]]

[[-3.83 ]
 [-0.918]
 [ 1.236]]

[[-3.82 ]
 [-0.891]
 [ 1.287]]
```

```
[[-3.81 ]
 [-0.866]
 [ 1.332]]

[[-3.8  ]
 [-0.836]
 [ 1.38 ]]

[[-3.81 ]
 [-0.868]
 [ 1.333]]

[[-3.82 ]
 [-0.899]
 [ 1.284]]

[[-3.83 ]
 [-0.924]
 [ 1.235]]

[[-3.82 ]
 [-0.897]
 [ 1.286]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.8  ]
 [-0.842]
 [ 1.379]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.82 ]
 [-0.895]
 [ 1.285]]

[[-3.83 ]
 [-0.92 ]
 [ 1.236]]

[[-3.82 ]
 [-0.893]
 [ 1.287]]

[[-3.81 ]
 [-0.868]
 [ 1.332]]

[[-3.8  ]
 [-0.838]
 [ 1.38 ]]

[[-3.81 ]
 [-0.87 ]
 [ 1.333]]

[[-3.82 ]
 [-0.901]
 [ 1.284]]

[[-3.83 ]
 [-0.926]
 [ 1.235]]

[[-3.82 ]
 [-0.899]
 [ 1.286]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]
```

```
[[-3.8  ]
 [-0.844]
 [ 1.379]]

[[-3.81 ]
 [-0.875]
 [ 1.33 ]]

[[-3.82 ]
 [-0.897]
 [ 1.285]]

[[-3.83 ]
 [-0.922]
 [ 1.236]]

[[-3.82 ]
 [-0.895]
 [ 1.287]]

[[-3.81 ]
 [-0.87 ]
 [ 1.332]]

[[-3.8 ]
 [-0.84]
 [ 1.38]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.82 ]
 [-0.893]
 [ 1.286]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.84 ]
 [-0.945]
 [ 1.186]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.82 ]
 [-0.893]
 [ 1.282]]

[[-3.81 ]
 [-0.861]
 [ 1.333]]

[[-3.82 ]
 [-0.892]
 [ 1.284]]

[[-3.83 ]
 [-0.917]
 [ 1.235]]

[[-3.82 ]
 [-0.89 ]
 [ 1.286]]

[[-3.81 ]
 [-0.865]
 [ 1.331]]

[[-3.8  ]
 [-0.835]
```

```
 [ 1.379]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.82 ]
 [-0.898]
 [ 1.283]]

[[-3.83 ]
 [-0.923]
 [ 1.234]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.81 ]
 [-0.871]
 [ 1.33 ]]

[[-3.8  ]
 [-0.841]
 [ 1.378]]

[[-3.81 ]
 [-0.872]
 [ 1.329]]

[[-3.82 ]
 [-0.894]
 [ 1.284]]

[[-3.83 ]
 [-0.919]
 [ 1.235]]

[[-3.82 ]
 [-0.892]
 [ 1.286]]

[[-3.81 ]
 [-0.867]
 [ 1.331]]

[[-3.8  ]
 [-0.837]
 [ 1.379]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.82 ]
 [-0.9  ]
 [ 1.283]]

[[-3.83 ]
 [-0.925]
 [ 1.234]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.8  ]
 [-0.843]
 [ 1.378]]

[[-3.81 ]
```

```
 [-0.874]
 [ 1.329]]

[[-3.82 ]
 [-0.896]
 [ 1.284]]

[[-3.83 ]
 [-0.921]
 [ 1.235]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.81 ]
 [-0.869]
 [ 1.331]]

[[-3.8  ]
 [-0.839]
 [ 1.379]]

[[-3.81]
 [-0.87]
 [ 1.33]]

[[-3.82 ]
 [-0.892]
 [ 1.285]]

[[-3.83 ]
 [-0.917]
 [ 1.236]]

[[-3.82 ]
 [-0.89 ]
 [ 1.287]]

[[-3.81 ]
 [-0.865]
 [ 1.332]]

[[-3.8  ]
 [-0.835]
 [ 1.38 ]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.82 ]
 [-0.898]
 [ 1.284]]

[[-3.83 ]
 [-0.923]
 [ 1.235]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.8  ]
 [-0.841]
 [ 1.379]]

[[-3.81 ]
 [-0.872]
 [ 1.33 ]]
```

```
[[-3.82 ]
 [-0.894]
 [ 1.285]]

[[-3.83 ]
 [-0.919]
 [ 1.236]]

[[-3.82 ]
 [-0.892]
 [ 1.287]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.8  ]
 [-0.837]
 [ 1.38 ]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.82 ]
 [-0.9  ]
 [ 1.284]]

[[-3.83 ]
 [-0.925]
 [ 1.235]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.81 ]
 [-0.873]
 [ 1.331]]

[[-3.8  ]
 [-0.843]
 [ 1.379]]

[[-3.81 ]
 [-0.874]
 [ 1.33 ]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.83 ]
 [-0.921]
 [ 1.236]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.8  ]
 [-0.839]
 [ 1.38 ]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.82 ]
 [-0.902]
 [ 1.284]]
```

```
[[-3.83 ]
 [-0.927]
 [ 1.235]]

[[-3.82 ]
 [-0.9  ]
 [ 1.286]]

[[-3.81 ]
 [-0.875]
 [ 1.331]]

[[-3.8  ]
 [-0.845]
 [ 1.379]]

[[-3.81 ]
 [-0.876]
 [ 1.33 ]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.83 ]
 [-0.923]
 [ 1.236]]

[[-3.82 ]
 [-0.896]
 [ 1.287]]

[[-3.81 ]
 [-0.871]
 [ 1.332]]

[[-3.8  ]
 [-0.841]
 [ 1.38 ]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.83 ]
 [-0.919]
 [ 1.237]]

[[-3.82 ]
 [-0.892]
 [ 1.288]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.8  ]
 [-0.837]
 [ 1.381]]

[[-3.81 ]
 [-0.869]
 [ 1.334]]

[[-3.82 ]
 [-0.9  ]
 [ 1.285]]

[[-3.83 ]
 [-0.925]
```

```
 [ 1.236]]

[[-3.82 ]
 [-0.898]
 [ 1.287]]

[[-3.81 ]
 [-0.873]
 [ 1.332]]

[[-3.8  ]
 [-0.843]
 [ 1.38 ]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.83 ]
 [-0.921]
 [ 1.237]]

[[-3.82 ]
 [-0.894]
 [ 1.288]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.8  ]
 [-0.839]
 [ 1.381]]

[[-3.81 ]
 [-0.871]
 [ 1.334]]

[[-3.82 ]
 [-0.902]
 [ 1.285]]

[[-3.83 ]
 [-0.927]
 [ 1.236]]

[[-3.82 ]
 [-0.9  ]
 [ 1.287]]

[[-3.81 ]
 [-0.875]
 [ 1.332]]

[[-3.8  ]
 [-0.845]
 [ 1.38 ]]

[[-3.81 ]
 [-0.876]
 [ 1.331]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.83 ]
 [-0.923]
 [ 1.237]]

[[-3.82 ]
```

```
 [-0.896]
 [ 1.288]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.8  ]
 [-0.841]
 [ 1.381]]

[[-3.81 ]
 [-0.872]
 [ 1.332]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.84 ]
 [-0.946]
 [ 1.187]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.82 ]
 [-0.894]
 [ 1.283]]

[[-3.81 ]
 [-0.862]
 [ 1.334]]

[[-3.82 ]
 [-0.893]
 [ 1.285]]

[[-3.83 ]
 [-0.918]
 [ 1.236]]

[[-3.82 ]
 [-0.891]
 [ 1.287]]

[[-3.81 ]
 [-0.866]
 [ 1.332]]

[[-3.8  ]
 [-0.836]
 [ 1.38 ]]

[[-3.81 ]
 [-0.868]
 [ 1.333]]

[[-3.82 ]
 [-0.899]
 [ 1.284]]

[[-3.83 ]
 [-0.924]
 [ 1.235]]

[[-3.82 ]
 [-0.897]
 [ 1.286]]
```

```
[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.8  ]
 [-0.842]
 [ 1.379]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.82 ]
 [-0.895]
 [ 1.285]]

[[-3.83 ]
 [-0.92 ]
 [ 1.236]]

[[-3.82 ]
 [-0.893]
 [ 1.287]]

[[-3.81 ]
 [-0.868]
 [ 1.332]]

[[-3.8  ]
 [-0.838]
 [ 1.38 ]]

[[-3.81 ]
 [-0.87 ]
 [ 1.333]]

[[-3.82 ]
 [-0.901]
 [ 1.284]]

[[-3.83 ]
 [-0.926]
 [ 1.235]]

[[-3.82 ]
 [-0.899]
 [ 1.286]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.8  ]
 [-0.844]
 [ 1.379]]

[[-3.81 ]
 [-0.875]
 [ 1.33 ]]

[[-3.82 ]
 [-0.897]
 [ 1.285]]

[[-3.83 ]
 [-0.922]
 [ 1.236]]

[[-3.82 ]
 [-0.895]
 [ 1.287]]

[[-3.81 ]
 [-0.87 ]
 [ 1.332]]
```

```
[[-3.8 ]
 [-0.84]
 [ 1.38]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.82 ]
 [-0.893]
 [ 1.286]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.84 ]
 [-0.945]
 [ 1.186]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.82 ]
 [-0.893]
 [ 1.282]]

[[-3.81 ]
 [-0.861]
 [ 1.333]]

[[-3.82 ]
 [-0.892]
 [ 1.284]]

[[-3.83 ]
 [-0.917]
 [ 1.235]]

[[-3.82 ]
 [-0.89 ]
 [ 1.286]]

[[-3.81 ]
 [-0.865]
 [ 1.331]]

[[-3.8  ]
 [-0.835]
 [ 1.379]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.82 ]
 [-0.898]
 [ 1.283]]

[[-3.83 ]
 [-0.923]
 [ 1.234]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.81 ]
 [-0.871]
 [ 1.33 ]]

[[-3.8  ]
 [-0.841]
```

```
 [ 1.378]]

[[-3.81 ]
 [-0.872]
 [ 1.329]]

[[-3.82 ]
 [-0.894]
 [ 1.284]]

[[-3.83 ]
 [-0.919]
 [ 1.235]]

[[-3.82 ]
 [-0.892]
 [ 1.286]]

[[-3.81 ]
 [-0.867]
 [ 1.331]]

[[-3.8  ]
 [-0.837]
 [ 1.379]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.82 ]
 [-0.9  ]
 [ 1.283]]

[[-3.83 ]
 [-0.925]
 [ 1.234]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.8  ]
 [-0.843]
 [ 1.378]]

[[-3.81 ]
 [-0.874]
 [ 1.329]]

[[-3.82 ]
 [-0.896]
 [ 1.284]]

[[-3.83 ]
 [-0.921]
 [ 1.235]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.81 ]
 [-0.869]
 [ 1.331]]

[[-3.8  ]
 [-0.839]
 [ 1.379]]

[[-3.81]
```

```
 [-0.87]
 [ 1.33]]

[[-3.82 ]
 [-0.892]
 [ 1.285]]

[[-3.83 ]
 [-0.917]
 [ 1.236]]

[[-3.82 ]
 [-0.89 ]
 [ 1.287]]

[[-3.81 ]
 [-0.865]
 [ 1.332]]

[[-3.8  ]
 [-0.835]
 [ 1.38 ]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.82 ]
 [-0.898]
 [ 1.284]]

[[-3.83 ]
 [-0.923]
 [ 1.235]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.8  ]
 [-0.841]
 [ 1.379]]

[[-3.81 ]
 [-0.872]
 [ 1.33 ]]

[[-3.82 ]
 [-0.894]
 [ 1.285]]

[[-3.83 ]
 [-0.919]
 [ 1.236]]

[[-3.82 ]
 [-0.892]
 [ 1.287]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.8  ]
 [-0.837]
 [ 1.38 ]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]
```

```
[[-3.82 ]
 [-0.9  ]
 [ 1.284]]

[[-3.83 ]
 [-0.925]
 [ 1.235]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.81 ]
 [-0.873]
 [ 1.331]]

[[-3.8  ]
 [-0.843]
 [ 1.379]]

[[-3.81 ]
 [-0.874]
 [ 1.33 ]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.83 ]
 [-0.921]
 [ 1.236]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.8  ]
 [-0.839]
 [ 1.38 ]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.82 ]
 [-0.902]
 [ 1.284]]

[[-3.83 ]
 [-0.927]
 [ 1.235]]

[[-3.82 ]
 [-0.9  ]
 [ 1.286]]

[[-3.81 ]
 [-0.875]
 [ 1.331]]

[[-3.8  ]
 [-0.845]
 [ 1.379]]

[[-3.81 ]
 [-0.876]
 [ 1.33 ]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]
```

```
[[-3.83 ]
 [-0.923]
 [ 1.236]]

[[-3.82 ]
 [-0.896]
 [ 1.287]]

[[-3.81 ]
 [-0.871]
 [ 1.332]]

[[-3.8  ]
 [-0.841]
 [ 1.38 ]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.83 ]
 [-0.919]
 [ 1.237]]

[[-3.82 ]
 [-0.892]
 [ 1.288]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.8  ]
 [-0.837]
 [ 1.381]]

[[-3.81 ]
 [-0.869]
 [ 1.334]]

[[-3.82 ]
 [-0.9  ]
 [ 1.285]]

[[-3.83 ]
 [-0.925]
 [ 1.236]]

[[-3.82 ]
 [-0.898]
 [ 1.287]]

[[-3.81 ]
 [-0.873]
 [ 1.332]]

[[-3.8  ]
 [-0.843]
 [ 1.38 ]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.83 ]
 [-0.921]
```

```
 [ 1.237]]

[[-3.82 ]
 [-0.894]
 [ 1.288]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.8  ]
 [-0.839]
 [ 1.381]]

[[-3.81 ]
 [-0.871]
 [ 1.334]]

[[-3.82 ]
 [-0.902]
 [ 1.285]]

[[-3.83 ]
 [-0.927]
 [ 1.236]]

[[-3.82 ]
 [-0.9  ]
 [ 1.287]]

[[-3.81 ]
 [-0.875]
 [ 1.332]]

[[-3.8  ]
 [-0.845]
 [ 1.38 ]]

[[-3.81 ]
 [-0.876]
 [ 1.331]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.83 ]
 [-0.923]
 [ 1.237]]

[[-3.82 ]
 [-0.896]
 [ 1.288]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.8  ]
 [-0.841]
 [ 1.381]]

[[-3.81 ]
 [-0.872]
 [ 1.332]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.84 ]
```

```
 [-0.946]
 [ 1.187]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.82 ]
 [-0.894]
 [ 1.283]]

[[-3.81 ]
 [-0.862]
 [ 1.334]]

[[-3.82 ]
 [-0.893]
 [ 1.285]]

[[-3.83 ]
 [-0.918]
 [ 1.236]]

[[-3.82 ]
 [-0.891]
 [ 1.287]]

[[-3.81 ]
 [-0.866]
 [ 1.332]]

[[-3.8  ]
 [-0.836]
 [ 1.38 ]]

[[-3.81 ]
 [-0.868]
 [ 1.333]]

[[-3.82 ]
 [-0.899]
 [ 1.284]]

[[-3.83 ]
 [-0.924]
 [ 1.235]]

[[-3.82 ]
 [-0.897]
 [ 1.286]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.8  ]
 [-0.842]
 [ 1.379]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.82 ]
 [-0.895]
 [ 1.285]]

[[-3.83 ]
 [-0.92 ]
 [ 1.236]]

[[-3.82 ]
 [-0.893]
 [ 1.287]]
```

```
[[-3.81 ]
 [-0.868]
 [ 1.332]]

[[-3.8  ]
 [-0.838]
 [ 1.38 ]]

[[-3.81 ]
 [-0.87 ]
 [ 1.333]]

[[-3.82 ]
 [-0.901]
 [ 1.284]]

[[-3.83 ]
 [-0.926]
 [ 1.235]]

[[-3.82 ]
 [-0.899]
 [ 1.286]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.8  ]
 [-0.844]
 [ 1.379]]

[[-3.81 ]
 [-0.875]
 [ 1.33 ]]

[[-3.82 ]
 [-0.897]
 [ 1.285]]

[[-3.83 ]
 [-0.922]
 [ 1.236]]

[[-3.82 ]
 [-0.895]
 [ 1.287]]

[[-3.81 ]
 [-0.87 ]
 [ 1.332]]

[[-3.8 ]
 [-0.84]
 [ 1.38]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.82 ]
 [-0.893]
 [ 1.286]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.84 ]
 [-0.945]
 [ 1.186]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]
```

```
[[-3.82 ]
 [-0.893]
 [ 1.282]]

[[-3.81 ]
 [-0.861]
 [ 1.333]]

[[-3.82 ]
 [-0.892]
 [ 1.284]]

[[-3.83 ]
 [-0.917]
 [ 1.235]]

[[-3.82 ]
 [-0.89 ]
 [ 1.286]]

[[-3.81 ]
 [-0.865]
 [ 1.331]]

[[-3.8  ]
 [-0.835]
 [ 1.379]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.82 ]
 [-0.898]
 [ 1.283]]

[[-3.83 ]
 [-0.923]
 [ 1.234]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.81 ]
 [-0.871]
 [ 1.33 ]]

[[-3.8  ]
 [-0.841]
 [ 1.378]]

[[-3.81 ]
 [-0.872]
 [ 1.329]]

[[-3.82 ]
 [-0.894]
 [ 1.284]]

[[-3.83 ]
 [-0.919]
 [ 1.235]]

[[-3.82 ]
 [-0.892]
 [ 1.286]]

[[-3.81 ]
 [-0.867]
 [ 1.331]]

[[-3.8  ]
 [-0.837]
```

```
 [ 1.379]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.82 ]
 [-0.9  ]
 [ 1.283]]

[[-3.83 ]
 [-0.925]
 [ 1.234]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.8  ]
 [-0.843]
 [ 1.378]]

[[-3.81 ]
 [-0.874]
 [ 1.329]]

[[-3.82 ]
 [-0.896]
 [ 1.284]]

[[-3.83 ]
 [-0.921]
 [ 1.235]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.81 ]
 [-0.869]
 [ 1.331]]

[[-3.8  ]
 [-0.839]
 [ 1.379]]

[[-3.81]
 [-0.87]
 [ 1.33]]

[[-3.82 ]
 [-0.892]
 [ 1.285]]

[[-3.83 ]
 [-0.917]
 [ 1.236]]

[[-3.82 ]
 [-0.89 ]
 [ 1.287]]

[[-3.81 ]
 [-0.865]
 [ 1.332]]

[[-3.8  ]
 [-0.835]
 [ 1.38 ]]

[[-3.81 ]
```

```
 [-0.867]
 [ 1.333]]

[[-3.82 ]
 [-0.898]
 [ 1.284]]

[[-3.83 ]
 [-0.923]
 [ 1.235]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.8  ]
 [-0.841]
 [ 1.379]]

[[-3.81 ]
 [-0.872]
 [ 1.33 ]]

[[-3.82 ]
 [-0.894]
 [ 1.285]]

[[-3.83 ]
 [-0.919]
 [ 1.236]]

[[-3.82 ]
 [-0.892]
 [ 1.287]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.8  ]
 [-0.837]
 [ 1.38 ]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.82 ]
 [-0.9  ]
 [ 1.284]]

[[-3.83 ]
 [-0.925]
 [ 1.235]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.81 ]
 [-0.873]
 [ 1.331]]

[[-3.8  ]
 [-0.843]
 [ 1.379]]

[[-3.81 ]
 [-0.874]
 [ 1.33 ]]
```

```
[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.83 ]
 [-0.921]
 [ 1.236]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.8  ]
 [-0.839]
 [ 1.38 ]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.82 ]
 [-0.902]
 [ 1.284]]

[[-3.83 ]
 [-0.927]
 [ 1.235]]

[[-3.82 ]
 [-0.9  ]
 [ 1.286]]

[[-3.81 ]
 [-0.875]
 [ 1.331]]

[[-3.8  ]
 [-0.845]
 [ 1.379]]

[[-3.81 ]
 [-0.876]
 [ 1.33 ]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.83 ]
 [-0.923]
 [ 1.236]]

[[-3.82 ]
 [-0.896]
 [ 1.287]]

[[-3.81 ]
 [-0.871]
 [ 1.332]]

[[-3.8  ]
 [-0.841]
 [ 1.38 ]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]
```

```
[[-3.83 ]
 [-0.919]
 [ 1.237]]

[[-3.82 ]
 [-0.892]
 [ 1.288]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.8  ]
 [-0.837]
 [ 1.381]]

[[-3.81 ]
 [-0.869]
 [ 1.334]]

[[-3.82 ]
 [-0.9  ]
 [ 1.285]]

[[-3.83 ]
 [-0.925]
 [ 1.236]]

[[-3.82 ]
 [-0.898]
 [ 1.287]]

[[-3.81 ]
 [-0.873]
 [ 1.332]]

[[-3.8  ]
 [-0.843]
 [ 1.38 ]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.83 ]
 [-0.921]
 [ 1.237]]

[[-3.82 ]
 [-0.894]
 [ 1.288]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.8  ]
 [-0.839]
 [ 1.381]]

[[-3.81 ]
 [-0.871]
 [ 1.334]]

[[-3.82 ]
 [-0.902]
 [ 1.285]]

[[-3.83 ]
 [-0.927]
```

```
 [ 1.236]]

[[-3.82 ]
 [-0.9  ]
 [ 1.287]]

[[-3.81 ]
 [-0.875]
 [ 1.332]]

[[-3.8  ]
 [-0.845]
 [ 1.38 ]]

[[-3.81 ]
 [-0.876]
 [ 1.331]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.83 ]
 [-0.923]
 [ 1.237]]

[[-3.82 ]
 [-0.896]
 [ 1.288]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.8  ]
 [-0.841]
 [ 1.381]]

[[-3.81 ]
 [-0.872]
 [ 1.332]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.84 ]
 [-0.946]
 [ 1.187]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.82 ]
 [-0.894]
 [ 1.283]]

[[-3.81 ]
 [-0.862]
 [ 1.334]]

[[-3.82 ]
 [-0.893]
 [ 1.285]]

[[-3.83 ]
 [-0.918]
 [ 1.236]]

[[-3.82 ]
```

```
 [-0.891]
 [ 1.287]]

[[-3.81 ]
 [-0.866]
 [ 1.332]]

[[-3.8  ]
 [-0.836]
 [ 1.38 ]]

[[-3.81 ]
 [-0.868]
 [ 1.333]]

[[-3.82 ]
 [-0.899]
 [ 1.284]]

[[-3.83 ]
 [-0.924]
 [ 1.235]]

[[-3.82 ]
 [-0.897]
 [ 1.286]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.8  ]
 [-0.842]
 [ 1.379]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.82 ]
 [-0.895]
 [ 1.285]]

[[-3.83 ]
 [-0.92 ]
 [ 1.236]]

[[-3.82 ]
 [-0.893]
 [ 1.287]]

[[-3.81 ]
 [-0.868]
 [ 1.332]]

[[-3.8  ]
 [-0.838]
 [ 1.38 ]]

[[-3.81 ]
 [-0.87 ]
 [ 1.333]]

[[-3.82 ]
 [-0.901]
 [ 1.284]]

[[-3.83 ]
 [-0.926]
 [ 1.235]]

[[-3.82 ]
 [-0.899]
 [ 1.286]]
```

```
[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.8  ]
 [-0.844]
 [ 1.379]]

[[-3.81 ]
 [-0.875]
 [ 1.33 ]]

[[-3.82 ]
 [-0.897]
 [ 1.285]]

[[-3.83 ]
 [-0.922]
 [ 1.236]]

[[-3.82 ]
 [-0.895]
 [ 1.287]]

[[-3.81 ]
 [-0.87 ]
 [ 1.332]]

[[-3.8 ]
 [-0.84]
 [ 1.38]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.82 ]
 [-0.893]
 [ 1.286]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.84 ]
 [-0.945]
 [ 1.186]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.82 ]
 [-0.893]
 [ 1.282]]

[[-3.81 ]
 [-0.861]
 [ 1.333]]

[[-3.82 ]
 [-0.892]
 [ 1.284]]

[[-3.83 ]
 [-0.917]
 [ 1.235]]

[[-3.82 ]
 [-0.89 ]
 [ 1.286]]

[[-3.81 ]
 [-0.865]
 [ 1.331]]
```

```
[[-3.8  ]
 [-0.835]
 [ 1.379]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.82 ]
 [-0.898]
 [ 1.283]]

[[-3.83 ]
 [-0.923]
 [ 1.234]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.81 ]
 [-0.871]
 [ 1.33 ]]

[[-3.8  ]
 [-0.841]
 [ 1.378]]

[[-3.81 ]
 [-0.872]
 [ 1.329]]

[[-3.82 ]
 [-0.894]
 [ 1.284]]

[[-3.83 ]
 [-0.919]
 [ 1.235]]

[[-3.82 ]
 [-0.892]
 [ 1.286]]

[[-3.81 ]
 [-0.867]
 [ 1.331]]

[[-3.8  ]
 [-0.837]
 [ 1.379]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.82 ]
 [-0.9  ]
 [ 1.283]]

[[-3.83 ]
 [-0.925]
 [ 1.234]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.8  ]
 [-0.843]
```

```
 [ 1.378]]

[[-3.81 ]
 [-0.874]
 [ 1.329]]

[[-3.82 ]
 [-0.896]
 [ 1.284]]

[[-3.83 ]
 [-0.921]
 [ 1.235]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.81 ]
 [-0.869]
 [ 1.331]]

[[-3.8  ]
 [-0.839]
 [ 1.379]]

[[-3.81]
 [-0.87]
 [ 1.33]]

[[-3.82 ]
 [-0.892]
 [ 1.285]]

[[-3.83 ]
 [-0.917]
 [ 1.236]]

[[-3.82 ]
 [-0.89 ]
 [ 1.287]]

[[-3.81 ]
 [-0.865]
 [ 1.332]]

[[-3.8  ]
 [-0.835]
 [ 1.38 ]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.82 ]
 [-0.898]
 [ 1.284]]

[[-3.83 ]
 [-0.923]
 [ 1.235]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.8  ]
 [-0.841]
 [ 1.379]]

[[-3.81 ]
```

```
 [-0.872]
 [ 1.33 ]]

[[-3.82 ]
 [-0.894]
 [ 1.285]]

[[-3.83 ]
 [-0.919]
 [ 1.236]]

[[-3.82 ]
 [-0.892]
 [ 1.287]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.8  ]
 [-0.837]
 [ 1.38 ]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.82 ]
 [-0.9  ]
 [ 1.284]]

[[-3.83 ]
 [-0.925]
 [ 1.235]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.81 ]
 [-0.873]
 [ 1.331]]

[[-3.8  ]
 [-0.843]
 [ 1.379]]

[[-3.81 ]
 [-0.874]
 [ 1.33 ]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.83 ]
 [-0.921]
 [ 1.236]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.8  ]
 [-0.839]
 [ 1.38 ]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]
```

```
[[-3.82 ]
 [-0.902]
 [ 1.284]]

[[-3.83 ]
 [-0.927]
 [ 1.235]]

[[-3.82 ]
 [-0.9  ]
 [ 1.286]]

[[-3.81 ]
 [-0.875]
 [ 1.331]]

[[-3.8  ]
 [-0.845]
 [ 1.379]]

[[-3.81 ]
 [-0.876]
 [ 1.33 ]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.83 ]
 [-0.923]
 [ 1.236]]

[[-3.82 ]
 [-0.896]
 [ 1.287]]

[[-3.81 ]
 [-0.871]
 [ 1.332]]

[[-3.8  ]
 [-0.841]
 [ 1.38 ]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.83 ]
 [-0.919]
 [ 1.237]]

[[-3.82 ]
 [-0.892]
 [ 1.288]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.8  ]
 [-0.837]
 [ 1.381]]

[[-3.81 ]
 [-0.869]
 [ 1.334]]

[[-3.82 ]
 [-0.9  ]
 [ 1.285]]
```

```
[[-3.83 ]
 [-0.925]
 [ 1.236]]

[[-3.82 ]
 [-0.898]
 [ 1.287]]

[[-3.81 ]
 [-0.873]
 [ 1.332]]

[[-3.8  ]
 [-0.843]
 [ 1.38 ]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.83 ]
 [-0.921]
 [ 1.237]]

[[-3.82 ]
 [-0.894]
 [ 1.288]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.8  ]
 [-0.839]
 [ 1.381]]

[[-3.81 ]
 [-0.871]
 [ 1.334]]

[[-3.82 ]
 [-0.902]
 [ 1.285]]

[[-3.83 ]
 [-0.927]
 [ 1.236]]

[[-3.82 ]
 [-0.9  ]
 [ 1.287]]

[[-3.81 ]
 [-0.875]
 [ 1.332]]

[[-3.8  ]
 [-0.845]
 [ 1.38 ]]

[[-3.81 ]
 [-0.876]
 [ 1.331]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.83 ]
 [-0.923]
```

```
 [ 1.237]]

[[-3.82 ]
 [-0.896]
 [ 1.288]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.8  ]
 [-0.841]
 [ 1.381]]

[[-3.81 ]
 [-0.872]
 [ 1.332]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.84 ]
 [-0.946]
 [ 1.187]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.82 ]
 [-0.894]
 [ 1.283]]

[[-3.81 ]
 [-0.862]
 [ 1.334]]

[[-3.82 ]
 [-0.893]
 [ 1.285]]

[[-3.83 ]
 [-0.918]
 [ 1.236]]

[[-3.82 ]
 [-0.891]
 [ 1.287]]

[[-3.81 ]
 [-0.866]
 [ 1.332]]

[[-3.8  ]
 [-0.836]
 [ 1.38 ]]

[[-3.81 ]
 [-0.868]
 [ 1.333]]

[[-3.82 ]
 [-0.899]
 [ 1.284]]

[[-3.83 ]
 [-0.924]
 [ 1.235]]

[[-3.82 ]
```

```
 [-0.897]
 [ 1.286]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.8  ]
 [-0.842]
 [ 1.379]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.82 ]
 [-0.895]
 [ 1.285]]

[[-3.83 ]
 [-0.92 ]
 [ 1.236]]

[[-3.82 ]
 [-0.893]
 [ 1.287]]

[[-3.81 ]
 [-0.868]
 [ 1.332]]

[[-3.8  ]
 [-0.838]
 [ 1.38 ]]

[[-3.81 ]
 [-0.87 ]
 [ 1.333]]

[[-3.82 ]
 [-0.901]
 [ 1.284]]

[[-3.83 ]
 [-0.926]
 [ 1.235]]

[[-3.82 ]
 [-0.899]
 [ 1.286]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.8  ]
 [-0.844]
 [ 1.379]]

[[-3.81 ]
 [-0.875]
 [ 1.33 ]]

[[-3.82 ]
 [-0.897]
 [ 1.285]]

[[-3.83 ]
 [-0.922]
 [ 1.236]]

[[-3.82 ]
 [-0.895]
 [ 1.287]]
```

```
[[-3.81 ]
 [-0.87 ]
 [ 1.332]]

[[-3.8 ]
 [-0.84]
 [ 1.38]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.82 ]
 [-0.893]
 [ 1.286]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.84 ]
 [-0.945]
 [ 1.186]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.82 ]
 [-0.893]
 [ 1.282]]

[[-3.81 ]
 [-0.861]
 [ 1.333]]

[[-3.82 ]
 [-0.892]
 [ 1.284]]

[[-3.83 ]
 [-0.917]
 [ 1.235]]

[[-3.82 ]
 [-0.89 ]
 [ 1.286]]

[[-3.81 ]
 [-0.865]
 [ 1.331]]

[[-3.8  ]
 [-0.835]
 [ 1.379]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.82 ]
 [-0.898]
 [ 1.283]]

[[-3.83 ]
 [-0.923]
 [ 1.234]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.81 ]
 [-0.871]
 [ 1.33 ]]
```

```
[[-3.8  ]
 [-0.841]
 [ 1.378]]

[[-3.81 ]
 [-0.872]
 [ 1.329]]

[[-3.82 ]
 [-0.894]
 [ 1.284]]

[[-3.83 ]
 [-0.919]
 [ 1.235]]

[[-3.82 ]
 [-0.892]
 [ 1.286]]

[[-3.81 ]
 [-0.867]
 [ 1.331]]

[[-3.8  ]
 [-0.837]
 [ 1.379]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.82 ]
 [-0.9  ]
 [ 1.283]]

[[-3.83 ]
 [-0.925]
 [ 1.234]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.8  ]
 [-0.843]
 [ 1.378]]

[[-3.81 ]
 [-0.874]
 [ 1.329]]

[[-3.82 ]
 [-0.896]
 [ 1.284]]

[[-3.83 ]
 [-0.921]
 [ 1.235]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.81 ]
 [-0.869]
 [ 1.331]]

[[-3.8  ]
 [-0.839]
```

```
 [ 1.379]]

[[-3.81]
 [-0.87]
 [ 1.33]]

[[-3.82 ]
 [-0.892]
 [ 1.285]]

[[-3.83 ]
 [-0.917]
 [ 1.236]]

[[-3.82 ]
 [-0.89 ]
 [ 1.287]]

[[-3.81 ]
 [-0.865]
 [ 1.332]]

[[-3.8  ]
 [-0.835]
 [ 1.38 ]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.82 ]
 [-0.898]
 [ 1.284]]

[[-3.83 ]
 [-0.923]
 [ 1.235]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.8  ]
 [-0.841]
 [ 1.379]]

[[-3.81 ]
 [-0.872]
 [ 1.33 ]]

[[-3.82 ]
 [-0.894]
 [ 1.285]]

[[-3.83 ]
 [-0.919]
 [ 1.236]]

[[-3.82 ]
 [-0.892]
 [ 1.287]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.8  ]
 [-0.837]
 [ 1.38 ]]

[[-3.81 ]
```

```
 [-0.869]
 [ 1.333]]

[[-3.82 ]
 [-0.9  ]
 [ 1.284]]

[[-3.83 ]
 [-0.925]
 [ 1.235]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.81 ]
 [-0.873]
 [ 1.331]]

[[-3.8  ]
 [-0.843]
 [ 1.379]]

[[-3.81 ]
 [-0.874]
 [ 1.33 ]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.83 ]
 [-0.921]
 [ 1.236]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.8  ]
 [-0.839]
 [ 1.38 ]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.82 ]
 [-0.902]
 [ 1.284]]

[[-3.83 ]
 [-0.927]
 [ 1.235]]

[[-3.82 ]
 [-0.9  ]
 [ 1.286]]

[[-3.81 ]
 [-0.875]
 [ 1.331]]

[[-3.8  ]
 [-0.845]
 [ 1.379]]

[[-3.81 ]
 [-0.876]
 [ 1.33 ]]
```

```
[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.83 ]
 [-0.923]
 [ 1.236]]

[[-3.82 ]
 [-0.896]
 [ 1.287]]

[[-3.81 ]
 [-0.871]
 [ 1.332]]

[[-3.8  ]
 [-0.841]
 [ 1.38 ]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.83 ]
 [-0.919]
 [ 1.237]]

[[-3.82 ]
 [-0.892]
 [ 1.288]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.8  ]
 [-0.837]
 [ 1.381]]

[[-3.81 ]
 [-0.869]
 [ 1.334]]

[[-3.82 ]
 [-0.9  ]
 [ 1.285]]

[[-3.83 ]
 [-0.925]
 [ 1.236]]

[[-3.82 ]
 [-0.898]
 [ 1.287]]

[[-3.81 ]
 [-0.873]
 [ 1.332]]

[[-3.8  ]
 [-0.843]
 [ 1.38 ]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]
```

```
[[-3.83 ]
 [-0.921]
 [ 1.237]]

[[-3.82 ]
 [-0.894]
 [ 1.288]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.8  ]
 [-0.839]
 [ 1.381]]

[[-3.81 ]
 [-0.871]
 [ 1.334]]

[[-3.82 ]
 [-0.902]
 [ 1.285]]

[[-3.83 ]
 [-0.927]
 [ 1.236]]

[[-3.82 ]
 [-0.9  ]
 [ 1.287]]

[[-3.81 ]
 [-0.875]
 [ 1.332]]

[[-3.8  ]
 [-0.845]
 [ 1.38 ]]

[[-3.81 ]
 [-0.876]
 [ 1.331]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.83 ]
 [-0.923]
 [ 1.237]]

[[-3.82 ]
 [-0.896]
 [ 1.288]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.8  ]
 [-0.841]
 [ 1.381]]

[[-3.81 ]
 [-0.872]
 [ 1.332]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.83 ]
 [-0.919]
```

```
 [ 1.238]]

[[-3.84 ]
 [-0.946]
 [ 1.187]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.82 ]
 [-0.894]
 [ 1.283]]

[[-3.81 ]
 [-0.862]
 [ 1.334]]

[[-3.82 ]
 [-0.893]
 [ 1.285]]

[[-3.83 ]
 [-0.918]
 [ 1.236]]

[[-3.82 ]
 [-0.891]
 [ 1.287]]

[[-3.81 ]
 [-0.866]
 [ 1.332]]

[[-3.8  ]
 [-0.836]
 [ 1.38 ]]

[[-3.81 ]
 [-0.868]
 [ 1.333]]

[[-3.82 ]
 [-0.899]
 [ 1.284]]

[[-3.83 ]
 [-0.924]
 [ 1.235]]

[[-3.82 ]
 [-0.897]
 [ 1.286]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.8  ]
 [-0.842]
 [ 1.379]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.82 ]
 [-0.895]
 [ 1.285]]

[[-3.83 ]
 [-0.92 ]
 [ 1.236]]

[[-3.82 ]
```

```
 [-0.893]
 [ 1.287]]

[[-3.81 ]
 [-0.868]
 [ 1.332]]

[[-3.8  ]
 [-0.838]
 [ 1.38 ]]

[[-3.81 ]
 [-0.87 ]
 [ 1.333]]

[[-3.82 ]
 [-0.901]
 [ 1.284]]

[[-3.83 ]
 [-0.926]
 [ 1.235]]

[[-3.82 ]
 [-0.899]
 [ 1.286]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.8  ]
 [-0.844]
 [ 1.379]]

[[-3.81 ]
 [-0.875]
 [ 1.33 ]]

[[-3.82 ]
 [-0.897]
 [ 1.285]]

[[-3.83 ]
 [-0.922]
 [ 1.236]]

[[-3.82 ]
 [-0.895]
 [ 1.287]]

[[-3.81 ]
 [-0.87 ]
 [ 1.332]]

[[-3.8 ]
 [-0.84]
 [ 1.38]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.82 ]
 [-0.893]
 [ 1.286]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.84 ]
 [-0.945]
 [ 1.186]]
```

```
[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.82 ]
 [-0.893]
 [ 1.282]]

[[-3.81 ]
 [-0.861]
 [ 1.333]]

[[-3.82 ]
 [-0.892]
 [ 1.284]]

[[-3.83 ]
 [-0.917]
 [ 1.235]]

[[-3.82 ]
 [-0.89 ]
 [ 1.286]]

[[-3.81 ]
 [-0.865]
 [ 1.331]]

[[-3.8  ]
 [-0.835]
 [ 1.379]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.82 ]
 [-0.898]
 [ 1.283]]

[[-3.83 ]
 [-0.923]
 [ 1.234]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.81 ]
 [-0.871]
 [ 1.33 ]]

[[-3.8  ]
 [-0.841]
 [ 1.378]]

[[-3.81 ]
 [-0.872]
 [ 1.329]]

[[-3.82 ]
 [-0.894]
 [ 1.284]]

[[-3.83 ]
 [-0.919]
 [ 1.235]]

[[-3.82 ]
 [-0.892]
 [ 1.286]]

[[-3.81 ]
 [-0.867]
 [ 1.331]]
```

```
[[-3.8  ]
 [-0.837]
 [ 1.379]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.82 ]
 [-0.9  ]
 [ 1.283]]

[[-3.83 ]
 [-0.925]
 [ 1.234]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.8  ]
 [-0.843]
 [ 1.378]]

[[-3.81 ]
 [-0.874]
 [ 1.329]]

[[-3.82 ]
 [-0.896]
 [ 1.284]]

[[-3.83 ]
 [-0.921]
 [ 1.235]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.81 ]
 [-0.869]
 [ 1.331]]

[[-3.8  ]
 [-0.839]
 [ 1.379]]

[[-3.81]
 [-0.87]
 [ 1.33]]

[[-3.82 ]
 [-0.892]
 [ 1.285]]

[[-3.83 ]
 [-0.917]
 [ 1.236]]

[[-3.82 ]
 [-0.89 ]
 [ 1.287]]

[[-3.81 ]
 [-0.865]
 [ 1.332]]

[[-3.8  ]
 [-0.835]
```

```
 [ 1.38 ]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.82 ]
 [-0.898]
 [ 1.284]]

[[-3.83 ]
 [-0.923]
 [ 1.235]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.8  ]
 [-0.841]
 [ 1.379]]

[[-3.81 ]
 [-0.872]
 [ 1.33 ]]

[[-3.82 ]
 [-0.894]
 [ 1.285]]

[[-3.83 ]
 [-0.919]
 [ 1.236]]

[[-3.82 ]
 [-0.892]
 [ 1.287]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.8  ]
 [-0.837]
 [ 1.38 ]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.82 ]
 [-0.9  ]
 [ 1.284]]

[[-3.83 ]
 [-0.925]
 [ 1.235]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.81 ]
 [-0.873]
 [ 1.331]]

[[-3.8  ]
 [-0.843]
 [ 1.379]]

[[-3.81 ]
```

```
 [-0.874]
 [ 1.33 ]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.83 ]
 [-0.921]
 [ 1.236]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.8  ]
 [-0.839]
 [ 1.38 ]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.82 ]
 [-0.902]
 [ 1.284]]

[[-3.83 ]
 [-0.927]
 [ 1.235]]

[[-3.82 ]
 [-0.9  ]
 [ 1.286]]

[[-3.81 ]
 [-0.875]
 [ 1.331]]

[[-3.8  ]
 [-0.845]
 [ 1.379]]

[[-3.81 ]
 [-0.876]
 [ 1.33 ]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.83 ]
 [-0.923]
 [ 1.236]]

[[-3.82 ]
 [-0.896]
 [ 1.287]]

[[-3.81 ]
 [-0.871]
 [ 1.332]]

[[-3.8  ]
 [-0.841]
 [ 1.38 ]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]
```

```
[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.83 ]
 [-0.919]
 [ 1.237]]

[[-3.82 ]
 [-0.892]
 [ 1.288]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.8  ]
 [-0.837]
 [ 1.381]]

[[-3.81 ]
 [-0.869]
 [ 1.334]]

[[-3.82 ]
 [-0.9  ]
 [ 1.285]]

[[-3.83 ]
 [-0.925]
 [ 1.236]]

[[-3.82 ]
 [-0.898]
 [ 1.287]]

[[-3.81 ]
 [-0.873]
 [ 1.332]]

[[-3.8  ]
 [-0.843]
 [ 1.38 ]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.83 ]
 [-0.921]
 [ 1.237]]

[[-3.82 ]
 [-0.894]
 [ 1.288]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.8  ]
 [-0.839]
 [ 1.381]]

[[-3.81 ]
 [-0.871]
 [ 1.334]]

[[-3.82 ]
 [-0.902]
 [ 1.285]]
```

```
[[-3.83 ]
 [-0.927]
 [ 1.236]]

[[-3.82 ]
 [-0.9  ]
 [ 1.287]]

[[-3.81 ]
 [-0.875]
 [ 1.332]]

[[-3.8  ]
 [-0.845]
 [ 1.38 ]]

[[-3.81 ]
 [-0.876]
 [ 1.331]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.83 ]
 [-0.923]
 [ 1.237]]

[[-3.82 ]
 [-0.896]
 [ 1.288]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.8  ]
 [-0.841]
 [ 1.381]]

[[-3.81 ]
 [-0.872]
 [ 1.332]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.84 ]
 [-0.946]
 [ 1.187]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.82 ]
 [-0.894]
 [ 1.283]]

[[-3.81 ]
 [-0.862]
 [ 1.334]]

[[-3.82 ]
 [-0.893]
 [ 1.285]]

[[-3.83 ]
 [-0.918]
```

```
 [ 1.236]]

[[-3.82 ]
 [-0.891]
 [ 1.287]]

[[-3.81 ]
 [-0.866]
 [ 1.332]]

[[-3.8  ]
 [-0.836]
 [ 1.38 ]]

[[-3.81 ]
 [-0.868]
 [ 1.333]]

[[-3.82 ]
 [-0.899]
 [ 1.284]]

[[-3.83 ]
 [-0.924]
 [ 1.235]]

[[-3.82 ]
 [-0.897]
 [ 1.286]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.8  ]
 [-0.842]
 [ 1.379]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.82 ]
 [-0.895]
 [ 1.285]]

[[-3.83 ]
 [-0.92 ]
 [ 1.236]]

[[-3.82 ]
 [-0.893]
 [ 1.287]]

[[-3.81 ]
 [-0.868]
 [ 1.332]]

[[-3.8  ]
 [-0.838]
 [ 1.38 ]]

[[-3.81 ]
 [-0.87 ]
 [ 1.333]]

[[-3.82 ]
 [-0.901]
 [ 1.284]]

[[-3.83 ]
 [-0.926]
 [ 1.235]]

[[-3.82 ]
```

```
 [-0.899]
 [ 1.286]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.8  ]
 [-0.844]
 [ 1.379]]

[[-3.81 ]
 [-0.875]
 [ 1.33 ]]

[[-3.82 ]
 [-0.897]
 [ 1.285]]

[[-3.83 ]
 [-0.922]
 [ 1.236]]

[[-3.82 ]
 [-0.895]
 [ 1.287]]

[[-3.81 ]
 [-0.87 ]
 [ 1.332]]

[[-3.8 ]
 [-0.84]
 [ 1.38]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.82 ]
 [-0.893]
 [ 1.286]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.84 ]
 [-0.945]
 [ 1.186]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.82 ]
 [-0.893]
 [ 1.282]]

[[-3.81 ]
 [-0.861]
 [ 1.333]]

[[-3.82 ]
 [-0.892]
 [ 1.284]]

[[-3.83 ]
 [-0.917]
 [ 1.235]]

[[-3.82 ]
 [-0.89 ]
 [ 1.286]]
```

```
[[-3.81 ]
 [-0.865]
 [ 1.331]]

[[-3.8  ]
 [-0.835]
 [ 1.379]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.82 ]
 [-0.898]
 [ 1.283]]

[[-3.83 ]
 [-0.923]
 [ 1.234]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.81 ]
 [-0.871]
 [ 1.33 ]]

[[-3.8  ]
 [-0.841]
 [ 1.378]]

[[-3.81 ]
 [-0.872]
 [ 1.329]]

[[-3.82 ]
 [-0.894]
 [ 1.284]]

[[-3.83 ]
 [-0.919]
 [ 1.235]]

[[-3.82 ]
 [-0.892]
 [ 1.286]]

[[-3.81 ]
 [-0.867]
 [ 1.331]]

[[-3.8  ]
 [-0.837]
 [ 1.379]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.82 ]
 [-0.9  ]
 [ 1.283]]

[[-3.83 ]
 [-0.925]
 [ 1.234]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]
```

```
[[-3.8  ]
 [-0.843]
 [ 1.378]]

[[-3.81 ]
 [-0.874]
 [ 1.329]]

[[-3.82 ]
 [-0.896]
 [ 1.284]]

[[-3.83 ]
 [-0.921]
 [ 1.235]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.81 ]
 [-0.869]
 [ 1.331]]

[[-3.8  ]
 [-0.839]
 [ 1.379]]

[[-3.81]
 [-0.87]
 [ 1.33]]

[[-3.82 ]
 [-0.892]
 [ 1.285]]

[[-3.83 ]
 [-0.917]
 [ 1.236]]

[[-3.82 ]
 [-0.89 ]
 [ 1.287]]

[[-3.81 ]
 [-0.865]
 [ 1.332]]

[[-3.8  ]
 [-0.835]
 [ 1.38 ]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.82 ]
 [-0.898]
 [ 1.284]]

[[-3.83 ]
 [-0.923]
 [ 1.235]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.8  ]
 [-0.841]
```

```
 [ 1.379]]

[[-3.81 ]
 [-0.872]
 [ 1.33 ]]

[[-3.82 ]
 [-0.894]
 [ 1.285]]

[[-3.83 ]
 [-0.919]
 [ 1.236]]

[[-3.82 ]
 [-0.892]
 [ 1.287]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.8  ]
 [-0.837]
 [ 1.38 ]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.82 ]
 [-0.9  ]
 [ 1.284]]

[[-3.83 ]
 [-0.925]
 [ 1.235]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.81 ]
 [-0.873]
 [ 1.331]]

[[-3.8  ]
 [-0.843]
 [ 1.379]]

[[-3.81 ]
 [-0.874]
 [ 1.33 ]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.83 ]
 [-0.921]
 [ 1.236]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.8  ]
 [-0.839]
 [ 1.38 ]]

[[-3.81 ]
```

```
 [-0.871]
 [ 1.333]]

[[-3.82 ]
 [-0.902]
 [ 1.284]]

[[-3.83 ]
 [-0.927]
 [ 1.235]]

[[-3.82 ]
 [-0.9  ]
 [ 1.286]]

[[-3.81 ]
 [-0.875]
 [ 1.331]]

[[-3.8  ]
 [-0.845]
 [ 1.379]]

[[-3.81 ]
 [-0.876]
 [ 1.33 ]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.83 ]
 [-0.923]
 [ 1.236]]

[[-3.82 ]
 [-0.896]
 [ 1.287]]

[[-3.81 ]
 [-0.871]
 [ 1.332]]

[[-3.8  ]
 [-0.841]
 [ 1.38 ]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.83 ]
 [-0.919]
 [ 1.237]]

[[-3.82 ]
 [-0.892]
 [ 1.288]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.8  ]
 [-0.837]
 [ 1.381]]

[[-3.81 ]
 [-0.869]
 [ 1.334]]
```

```
[[-3.82 ]
 [-0.9  ]
 [ 1.285]]

[[-3.83 ]
 [-0.925]
 [ 1.236]]

[[-3.82 ]
 [-0.898]
 [ 1.287]]

[[-3.81 ]
 [-0.873]
 [ 1.332]]

[[-3.8  ]
 [-0.843]
 [ 1.38 ]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.83 ]
 [-0.921]
 [ 1.237]]

[[-3.82 ]
 [-0.894]
 [ 1.288]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.8  ]
 [-0.839]
 [ 1.381]]

[[-3.81 ]
 [-0.871]
 [ 1.334]]

[[-3.82 ]
 [-0.902]
 [ 1.285]]

[[-3.83 ]
 [-0.927]
 [ 1.236]]

[[-3.82 ]
 [-0.9  ]
 [ 1.287]]

[[-3.81 ]
 [-0.875]
 [ 1.332]]

[[-3.8  ]
 [-0.845]
 [ 1.38 ]]

[[-3.81 ]
 [-0.876]
 [ 1.331]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]
```

```
[[-3.83 ]
 [-0.923]
 [ 1.237]]

[[-3.82 ]
 [-0.896]
 [ 1.288]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.8  ]
 [-0.841]
 [ 1.381]]

[[-3.81 ]
 [-0.872]
 [ 1.332]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.84 ]
 [-0.946]
 [ 1.187]]

[[-3.83 ]
 [-0.919]
 [ 1.238]]

[[-3.82 ]
 [-0.894]
 [ 1.283]]

[[-3.81 ]
 [-0.862]
 [ 1.334]]

[[-3.82 ]
 [-0.893]
 [ 1.285]]

[[-3.83 ]
 [-0.918]
 [ 1.236]]

[[-3.82 ]
 [-0.891]
 [ 1.287]]

[[-3.81 ]
 [-0.866]
 [ 1.332]]

[[-3.8  ]
 [-0.836]
 [ 1.38 ]]

[[-3.81 ]
 [-0.868]
 [ 1.333]]

[[-3.82 ]
 [-0.899]
 [ 1.284]]

[[-3.83 ]
 [-0.924]
```

```
 [ 1.235]]

[[-3.82 ]
 [-0.897]
 [ 1.286]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.8  ]
 [-0.842]
 [ 1.379]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.82 ]
 [-0.895]
 [ 1.285]]

[[-3.83 ]
 [-0.92 ]
 [ 1.236]]

[[-3.82 ]
 [-0.893]
 [ 1.287]]

[[-3.81 ]
 [-0.868]
 [ 1.332]]

[[-3.8  ]
 [-0.838]
 [ 1.38 ]]

[[-3.81 ]
 [-0.87 ]
 [ 1.333]]

[[-3.82 ]
 [-0.901]
 [ 1.284]]

[[-3.83 ]
 [-0.926]
 [ 1.235]]

[[-3.82 ]
 [-0.899]
 [ 1.286]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]

[[-3.8  ]
 [-0.844]
 [ 1.379]]

[[-3.81 ]
 [-0.875]
 [ 1.33 ]]

[[-3.82 ]
 [-0.897]
 [ 1.285]]

[[-3.83 ]
 [-0.922]
 [ 1.236]]

[[-3.82 ]
```

```
 [-0.895]
 [ 1.287]]

[[-3.81 ]
 [-0.87 ]
 [ 1.332]]

[[-3.8 ]
 [-0.84]
 [ 1.38]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.82 ]
 [-0.893]
 [ 1.286]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.84 ]
 [-0.945]
 [ 1.186]]

[[-3.83 ]
 [-0.918]
 [ 1.237]]

[[-3.82 ]
 [-0.893]
 [ 1.282]]

[[-3.81 ]
 [-0.861]
 [ 1.333]]

[[-3.82 ]
 [-0.892]
 [ 1.284]]

[[-3.83 ]
 [-0.917]
 [ 1.235]]

[[-3.82 ]
 [-0.89 ]
 [ 1.286]]

[[-3.81 ]
 [-0.865]
 [ 1.331]]

[[-3.8  ]
 [-0.835]
 [ 1.379]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.82 ]
 [-0.898]
 [ 1.283]]

[[-3.83 ]
 [-0.923]
 [ 1.234]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]
```

```
[[-3.81 ]
 [-0.871]
 [ 1.33 ]]

[[-3.8  ]
 [-0.841]
 [ 1.378]]

[[-3.81 ]
 [-0.872]
 [ 1.329]]

[[-3.82 ]
 [-0.894]
 [ 1.284]]

[[-3.83 ]
 [-0.919]
 [ 1.235]]

[[-3.82 ]
 [-0.892]
 [ 1.286]]

[[-3.81 ]
 [-0.867]
 [ 1.331]]

[[-3.8  ]
 [-0.837]
 [ 1.379]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.82 ]
 [-0.9  ]
 [ 1.283]]

[[-3.83 ]
 [-0.925]
 [ 1.234]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.81 ]
 [-0.873]
 [ 1.33 ]]

[[-3.8  ]
 [-0.843]
 [ 1.378]]

[[-3.81 ]
 [-0.874]
 [ 1.329]]

[[-3.82 ]
 [-0.896]
 [ 1.284]]

[[-3.83 ]
 [-0.921]
 [ 1.235]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.81 ]
 [-0.869]
 [ 1.331]]
```

```
[[-3.8  ]
 [-0.839]
 [ 1.379]]

[[-3.81]
 [-0.87]
 [ 1.33]]

[[-3.82 ]
 [-0.892]
 [ 1.285]]

[[-3.83 ]
 [-0.917]
 [ 1.236]]

[[-3.82 ]
 [-0.89 ]
 [ 1.287]]

[[-3.81 ]
 [-0.865]
 [ 1.332]]

[[-3.8  ]
 [-0.835]
 [ 1.38 ]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.82 ]
 [-0.898]
 [ 1.284]]

[[-3.83 ]
 [-0.923]
 [ 1.235]]

[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.81 ]
 [-0.871]
 [ 1.331]]

[[-3.8  ]
 [-0.841]
 [ 1.379]]

[[-3.81 ]
 [-0.872]
 [ 1.33 ]]

[[-3.82 ]
 [-0.894]
 [ 1.285]]

[[-3.83 ]
 [-0.919]
 [ 1.236]]

[[-3.82 ]
 [-0.892]
 [ 1.287]]

[[-3.81 ]
 [-0.867]
 [ 1.332]]

[[-3.8  ]
 [-0.837]
```

```
 [ 1.38 ]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.82 ]
 [-0.9  ]
 [ 1.284]]

[[-3.83 ]
 [-0.925]
 [ 1.235]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.81 ]
 [-0.873]
 [ 1.331]]

[[-3.8  ]
 [-0.843]
 [ 1.379]]

[[-3.81 ]
 [-0.874]
 [ 1.33 ]]

[[-3.82 ]
 [-0.896]
 [ 1.285]]

[[-3.83 ]
 [-0.921]
 [ 1.236]]

[[-3.82 ]
 [-0.894]
 [ 1.287]]

[[-3.81 ]
 [-0.869]
 [ 1.332]]

[[-3.8  ]
 [-0.839]
 [ 1.38 ]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.82 ]
 [-0.902]
 [ 1.284]]

[[-3.83 ]
 [-0.927]
 [ 1.235]]

[[-3.82 ]
 [-0.9  ]
 [ 1.286]]

[[-3.81 ]
 [-0.875]
 [ 1.331]]

[[-3.8  ]
 [-0.845]
 [ 1.379]]

[[-3.81 ]
```

```
 [-0.876]
 [ 1.33 ]]

[[-3.82 ]
 [-0.898]
 [ 1.285]]

[[-3.83 ]
 [-0.923]
 [ 1.236]]

[[-3.82 ]
 [-0.896]
 [ 1.287]]

[[-3.81 ]
 [-0.871]
 [ 1.332]]

[[-3.8  ]
 [-0.841]
 [ 1.38 ]]

[[-3.81 ]
 [-0.872]
 [ 1.331]]

[[-3.82 ]
 [-0.894]
 [ 1.286]]

[[-3.83 ]
 [-0.919]
 [ 1.237]]

[[-3.82 ]
 [-0.892]
 [ 1.288]]

[[-3.81 ]
 [-0.867]
 [ 1.333]]

[[-3.8  ]
 [-0.837]
 [ 1.381]]

[[-3.81 ]
 [-0.869]
 [ 1.334]]

[[-3.82 ]
 [-0.9  ]
 [ 1.285]]

[[-3.83 ]
 [-0.925]
 [ 1.236]]

[[-3.82 ]
 [-0.898]
 [ 1.287]]

[[-3.81 ]
 [-0.873]
 [ 1.332]]

[[-3.8  ]
 [-0.843]
 [ 1.38 ]]

[[-3.81 ]
 [-0.874]
 [ 1.331]]
```

```
[[-3.82 ]
 [-0.896]
 [ 1.286]]

[[-3.83 ]
 [-0.921]
 [ 1.237]]

[[-3.82 ]
 [-0.894]
 [ 1.288]]

[[-3.81 ]
 [-0.869]
 [ 1.333]]

[[-3.8  ]
 [-0.839]
 [ 1.381]]

[[-3.81 ]
 [-0.871]
 [ 1.334]]

[[-3.82 ]
 [-0.902]
 [ 1.285]]

[[-3.83 ]
 [-0.927]
 [ 1.236]]

[[-3.82 ]
 [-0.9  ]
 [ 1.287]]

[[-3.81 ]
 [-0.875]
 [ 1.332]]

[[-3.8  ]
 [-0.845]
 [ 1.38 ]]

[[-3.81 ]
 [-0.876]
 [ 1.331]]

[[-3.82 ]
 [-0.898]
 [ 1.286]]

[[-3.83 ]
 [-0.923]
 [ 1.237]]

[[-3.82 ]
 [-0.896]
 [ 1.288]]

[[-3.81 ]
 [-0.871]
 [ 1.333]]

[[-3.8  ]
 [-0.841]
 [ 1.381]]
```

In [ ]:

```
W3, X3, ans3, weight_vector3 = solve_perceptron('Versicolor', df)
```

Streaming output truncated to the last 5000 lines.

```
[[0.09 ]
 [0.022]
```

```
 [0.028]]

[[ 0.08 ]
 [-0.011]
 [-0.032]]

[[ 0.07 ]
 [-0.041]
 [-0.046]]

[[ 0.08 ]
 [-0.009]
 [ 0.001]]

[[ 0.07 ]
 [-0.042]
 [-0.059]]

[[ 0.08 ]
 [-0.01 ]
 [-0.012]]

[[0.09 ]
 [0.022]
 [0.033]]

[[ 0.08 ]
 [-0.011]
 [-0.027]]

[[ 0.07 ]
 [-0.046]
 [-0.041]]

[[ 0.08 ]
 [-0.014]
 [ 0.006]]

[[ 0.07 ]
 [-0.047]
 [-0.054]]

[[ 0.08 ]
 [-0.015]
 [-0.007]]

[[0.09 ]
 [0.016]
 [0.042]]

[[ 0.08 ]
 [-0.017]
 [-0.018]]

[[ 0.07 ]
 [-0.047]
 [-0.032]]

[[ 0.08 ]
 [-0.015]
 [ 0.015]]

[[ 0.07 ]
 [-0.048]
 [-0.045]]

[[ 0.08 ]
 [-0.016]
 [ 0.002]]

[[ 0.07 ]
 [-0.049]
 [-0.058]]

[[ 0.08 ]
```

```
 [-0.017]
 [-0.011]]

[[0.09 ]
 [0.015]
 [0.034]]

[[ 0.08 ]
 [-0.018]
 [-0.026]]

[[ 0.07 ]
 [-0.041]
 [-0.039]]

[[ 0.08 ]
 [-0.009]
 [ 0.008]]

[[ 0.07 ]
 [-0.042]
 [-0.052]]

[[ 0.08 ]
 [-0.01 ]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.065]]

[[ 0.08 ]
 [-0.011]
 [-0.018]]

[[0.09 ]
 [0.021]
 [0.027]]

[[ 0.08 ]
 [-0.012]
 [-0.033]]

[[ 0.07 ]
 [-0.042]
 [-0.044]]

[[ 0.08 ]
 [-0.01 ]
 [ 0.003]]

[[ 0.07 ]
 [-0.043]
 [-0.057]]

[[ 0.08 ]
 [-0.011]
 [-0.01 ]]

[[0.09 ]
 [0.021]
 [0.035]]

[[ 0.08 ]
 [-0.012]
 [-0.025]]

[[ 0.07 ]
 [-0.047]
 [-0.039]]

[[ 0.08 ]
 [-0.015]
 [ 0.008]]
```

```
[[ 0.07 ]
 [-0.048]
 [-0.052]]

[[ 0.08 ]
 [-0.016]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.056]]

[[ 0.08 ]
 [-0.011]
 [-0.009]]

[[ 0.07 ]
 [-0.038]
 [-0.06 ]]

[[ 0.08 ]
 [-0.006]
 [-0.013]]

[[0.09 ]
 [0.025]
 [0.036]]

[[ 0.08 ]
 [-0.008]
 [-0.024]]

[[ 0.07 ]
 [-0.043]
 [-0.038]]

[[ 0.08 ]
 [-0.011]
 [ 0.009]]

[[ 0.07 ]
 [-0.044]
 [-0.051]]

[[ 0.08 ]
 [-0.012]
 [-0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.064]]

[[ 0.08 ]
 [-0.013]
 [-0.017]]

[[0.09 ]
 [0.019]
 [0.028]]

[[ 0.08 ]
 [-0.014]
 [-0.032]]

[[ 0.07 ]
 [-0.044]
 [-0.043]]

[[ 0.08 ]
 [-0.012]
 [ 0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.056]]
```

```
[[ 0.08 ]
 [-0.013]
 [-0.009]]

[[0.09 ]
 [0.019]
 [0.036]]

[[ 0.08 ]
 [-0.014]
 [-0.024]]

[[ 0.07 ]
 [-0.044]
 [-0.038]]

[[ 0.08 ]
 [-0.012]
 [ 0.009]]

[[ 0.07 ]
 [-0.045]
 [-0.051]]

[[ 0.08 ]
 [-0.013]
 [-0.004]]

[[ 0.07 ]
 [-0.046]
 [-0.064]]

[[ 0.08 ]
 [-0.014]
 [-0.017]]

[[0.09 ]
 [0.018]
 [0.028]]

[[ 0.08 ]
 [-0.015]
 [-0.032]]

[[ 0.07 ]
 [-0.038]
 [-0.045]]

[[ 0.08 ]
 [-0.006]
 [ 0.002]]

[[ 0.07 ]
 [-0.039]
 [-0.058]]

[[ 0.08 ]
 [-0.007]
 [-0.011]]

[[ 0.07 ]
 [-0.034]
 [-0.062]]

[[ 0.08 ]
 [-0.002]
 [-0.015]]

[[0.09 ]
 [0.028]
 [0.035]]

[[ 0.08 ]
 [-0.005]
```

```
 [-0.025]]

[[ 0.07 ]
 [-0.04 ]
 [-0.039]]

[[ 0.08 ]
 [-0.008]
 [ 0.008]]

[[ 0.07 ]
 [-0.041]
 [-0.052]]

[[ 0.08 ]
 [-0.009]
 [-0.005]]

[[ 0.07 ]
 [-0.042]
 [-0.065]]

[[ 0.08 ]
 [-0.01 ]
 [-0.018]]

[[0.09 ]
 [0.022]
 [0.027]]

[[ 0.08 ]
 [-0.011]
 [-0.033]]

[[ 0.07 ]
 [-0.041]
 [-0.047]]

[[ 8.00000000e-02]
 [-9.00000000e-03]
 [ 8.70137296e-14]]

[[ 0.07 ]
 [-0.042]
 [-0.06 ]]

[[ 0.08 ]
 [-0.01 ]
 [-0.013]]

[[0.09 ]
 [0.022]
 [0.032]]

[[ 0.08 ]
 [-0.011]
 [-0.028]]

[[ 0.07 ]
 [-0.046]
 [-0.042]]

[[ 0.08 ]
 [-0.014]
 [ 0.005]]

[[ 0.07 ]
 [-0.047]
 [-0.055]]

[[ 0.08 ]
 [-0.015]
 [-0.008]]

[[0.09 ]
```

```
 [0.017]
 [0.037]]

[[ 0.08 ]
 [-0.016]
 [-0.023]]

[[ 0.07 ]
 [-0.045]
 [-0.037]]

[[ 0.08 ]
 [-0.013]
 [ 0.01 ]]

[[ 0.07 ]
 [-0.046]
 [-0.05 ]]

[[ 0.08 ]
 [-0.014]
 [-0.003]]

[[ 0.07 ]
 [-0.047]
 [-0.063]]

[[ 0.08 ]
 [-0.015]
 [-0.016]]

[[0.09 ]
 [0.017]
 [0.029]]

[[ 0.08 ]
 [-0.016]
 [-0.031]]

[[ 0.07 ]
 [-0.039]
 [-0.044]]

[[ 0.08 ]
 [-0.007]
 [ 0.003]]

[[ 0.07 ]
 [-0.04 ]
 [-0.057]]

[[ 0.08 ]
 [-0.008]
 [-0.01 ]]

[[ 0.07 ]
 [-0.035]
 [-0.061]]

[[ 0.08 ]
 [-0.003]
 [-0.014]]

[[ 0.07 ]
 [-0.03 ]
 [-0.065]]

[[ 0.08 ]
 [ 0.002]
 [-0.018]]

[[0.09 ]
 [0.033]
 [0.031]]
```

```
[[ 8.0000000e-02]
 [-3.2765457e-14]
 [-2.9000000e-02]]

[[ 0.07 ]
 [-0.035]
 [-0.043]]

[[ 0.08 ]
 [-0.003]
 [ 0.004]]

[[ 0.07 ]
 [-0.036]
 [-0.056]]

[[ 0.08 ]
 [-0.004]
 [-0.009]]

[[ 0.07 ]
 [-0.037]
 [-0.069]]

[[ 0.08 ]
 [-0.005]
 [-0.022]]

[[0.09 ]
 [0.027]
 [0.023]]

[[ 0.08 ]
 [-0.006]
 [-0.037]]

[[ 0.07 ]
 [-0.041]
 [-0.051]]

[[ 0.08 ]
 [-0.009]
 [-0.004]]

[[ 0.07 ]
 [-0.042]
 [-0.064]]

[[ 0.08 ]
 [-0.01 ]
 [-0.017]]

[[0.09 ]
 [0.022]
 [0.028]]

[[ 0.08 ]
 [-0.011]
 [-0.032]]

[[ 0.07 ]
 [-0.041]
 [-0.046]]

[[ 0.08 ]
 [-0.009]
 [ 0.001]]

[[ 0.07 ]
 [-0.042]
 [-0.059]]

[[ 0.08 ]
 [-0.01 ]
 [-0.012]]
```

```
[[0.09 ]
 [0.022]
 [0.033]]

[[ 0.08 ]
 [-0.011]
 [-0.027]]

[[ 0.07 ]
 [-0.046]
 [-0.041]]

[[ 0.08 ]
 [-0.014]
 [ 0.006]]

[[ 0.07 ]
 [-0.047]
 [-0.054]]

[[ 0.08 ]
 [-0.015]
 [-0.007]]

[[0.09 ]
 [0.016]
 [0.042]]

[[ 0.08 ]
 [-0.017]
 [-0.018]]

[[ 0.07 ]
 [-0.047]
 [-0.032]]

[[ 0.08 ]
 [-0.015]
 [ 0.015]]

[[ 0.07 ]
 [-0.048]
 [-0.045]]

[[ 0.08 ]
 [-0.016]
 [ 0.002]]

[[ 0.07 ]
 [-0.049]
 [-0.058]]

[[ 0.08 ]
 [-0.017]
 [-0.011]]

[[0.09 ]
 [0.015]
 [0.034]]

[[ 0.08 ]
 [-0.018]
 [-0.026]]

[[ 0.07 ]
 [-0.041]
 [-0.039]]

[[ 0.08 ]
 [-0.009]
 [ 0.008]]

[[ 0.07 ]
 [-0.042]
```

```
 [-0.052]]

[[ 0.08 ]
 [-0.01 ]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.065]]

[[ 0.08 ]
 [-0.011]
 [-0.018]]

[[0.09 ]
 [0.021]
 [0.027]]

[[ 0.08 ]
 [-0.012]
 [-0.033]]

[[ 0.07 ]
 [-0.042]
 [-0.044]]

[[ 0.08 ]
 [-0.01 ]
 [ 0.003]]

[[ 0.07 ]
 [-0.043]
 [-0.057]]

[[ 0.08 ]
 [-0.011]
 [-0.01 ]]

[[0.09 ]
 [0.021]
 [0.035]]

[[ 0.08 ]
 [-0.012]
 [-0.025]]

[[ 0.07 ]
 [-0.047]
 [-0.039]]

[[ 0.08 ]
 [-0.015]
 [ 0.008]]

[[ 0.07 ]
 [-0.048]
 [-0.052]]

[[ 0.08 ]
 [-0.016]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.056]]

[[ 0.08 ]
 [-0.011]
 [-0.009]]

[[ 0.07 ]
 [-0.038]
 [-0.06 ]]

[[ 0.08 ]
```

```
 [-0.006]
 [-0.013]]

[[0.09 ]
 [0.025]
 [0.036]]

[[ 0.08 ]
 [-0.008]
 [-0.024]]

[[ 0.07 ]
 [-0.043]
 [-0.038]]

[[ 0.08 ]
 [-0.011]
 [ 0.009]]

[[ 0.07 ]
 [-0.044]
 [-0.051]]

[[ 0.08 ]
 [-0.012]
 [-0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.064]]

[[ 0.08 ]
 [-0.013]
 [-0.017]]

[[0.09 ]
 [0.019]
 [0.028]]

[[ 0.08 ]
 [-0.014]
 [-0.032]]

[[ 0.07 ]
 [-0.044]
 [-0.043]]

[[ 0.08 ]
 [-0.012]
 [ 0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.056]]

[[ 0.08 ]
 [-0.013]
 [-0.009]]

[[0.09 ]
 [0.019]
 [0.036]]

[[ 0.08 ]
 [-0.014]
 [-0.024]]

[[ 0.07 ]
 [-0.044]
 [-0.038]]

[[ 0.08 ]
 [-0.012]
 [ 0.009]]
```

```
[[ 0.07 ]
 [-0.045]
 [-0.051]]

[[ 0.08 ]
 [-0.013]
 [-0.004]]

[[ 0.07 ]
 [-0.046]
 [-0.064]]

[[ 0.08 ]
 [-0.014]
 [-0.017]]

[[0.09 ]
 [0.018]
 [0.028]]

[[ 0.08 ]
 [-0.015]
 [-0.032]]

[[ 0.07 ]
 [-0.038]
 [-0.045]]

[[ 0.08 ]
 [-0.006]
 [ 0.002]]

[[ 0.07 ]
 [-0.039]
 [-0.058]]

[[ 0.08 ]
 [-0.007]
 [-0.011]]

[[ 0.07 ]
 [-0.034]
 [-0.062]]

[[ 0.08 ]
 [-0.002]
 [-0.015]]

[[0.09 ]
 [0.028]
 [0.035]]

[[ 0.08 ]
 [-0.005]
 [-0.025]]

[[ 0.07 ]
 [-0.04 ]
 [-0.039]]

[[ 0.08 ]
 [-0.008]
 [ 0.008]]

[[ 0.07 ]
 [-0.041]
 [-0.052]]

[[ 0.08 ]
 [-0.009]
 [-0.005]]

[[ 0.07 ]
 [-0.042]
 [-0.065]]
```

```
[[ 0.08 ]
 [-0.01 ]
 [-0.018]]

[[0.09 ]
 [0.022]
 [0.027]]

[[ 0.08 ]
 [-0.011]
 [-0.033]]

[[ 0.07 ]
 [-0.041]
 [-0.047]]

[[ 8.00000000e-02]
 [-9.00000000e-03]
 [ 8.71247519e-14]]

[[ 0.07 ]
 [-0.042]
 [-0.06 ]]

[[ 0.08 ]
 [-0.01 ]
 [-0.013]]

[[0.09 ]
 [0.022]
 [0.032]]

[[ 0.08 ]
 [-0.011]
 [-0.028]]

[[ 0.07 ]
 [-0.046]
 [-0.042]]

[[ 0.08 ]
 [-0.014]
 [ 0.005]]

[[ 0.07 ]
 [-0.047]
 [-0.055]]

[[ 0.08 ]
 [-0.015]
 [-0.008]]

[[0.09 ]
 [0.017]
 [0.037]]

[[ 0.08 ]
 [-0.016]
 [-0.023]]

[[ 0.07 ]
 [-0.045]
 [-0.037]]

[[ 0.08 ]
 [-0.013]
 [ 0.01 ]]

[[ 0.07 ]
 [-0.046]
 [-0.05 ]]

[[ 0.08 ]
 [-0.014]
```

```
 [-0.003]]

[[ 0.07 ]
 [-0.047]
 [-0.063]]

[[ 0.08 ]
 [-0.015]
 [-0.016]]

[[0.09 ]
 [0.017]
 [0.029]]

[[ 0.08 ]
 [-0.016]
 [-0.031]]

[[ 0.07 ]
 [-0.039]
 [-0.044]]

[[ 0.08 ]
 [-0.007]
 [ 0.003]]

[[ 0.07 ]
 [-0.04 ]
 [-0.057]]

[[ 0.08 ]
 [-0.008]
 [-0.01 ]]

[[ 0.07 ]
 [-0.035]
 [-0.061]]

[[ 0.08 ]
 [-0.003]
 [-0.014]]

[[ 0.07 ]
 [-0.03 ]
 [-0.065]]

[[ 0.08 ]
 [ 0.002]
 [-0.018]]

[[0.09 ]
 [0.033]
 [0.031]]

[[ 8.00000000e-02]
 [-3.28070904e-14]
 [-2.90000000e-02]]

[[ 0.07 ]
 [-0.035]
 [-0.043]]

[[ 0.08 ]
 [-0.003]
 [ 0.004]]

[[ 0.07 ]
 [-0.036]
 [-0.056]]

[[ 0.08 ]
 [-0.004]
 [-0.009]]

[[ 0.07 ]
```

```
 [-0.037]
 [-0.069]]

[[ 0.08 ]
 [-0.005]
 [-0.022]]

[[0.09 ]
 [0.027]
 [0.023]]

[[ 0.08 ]
 [-0.006]
 [-0.037]]

[[ 0.07 ]
 [-0.041]
 [-0.051]]

[[ 0.08 ]
 [-0.009]
 [-0.004]]

[[ 0.07 ]
 [-0.042]
 [-0.064]]

[[ 0.08 ]
 [-0.01 ]
 [-0.017]]

[[0.09 ]
 [0.022]
 [0.028]]

[[ 0.08 ]
 [-0.011]
 [-0.032]]

[[ 0.07 ]
 [-0.041]
 [-0.046]]

[[ 0.08 ]
 [-0.009]
 [ 0.001]]

[[ 0.07 ]
 [-0.042]
 [-0.059]]

[[ 0.08 ]
 [-0.01 ]
 [-0.012]]

[[0.09 ]
 [0.022]
 [0.033]]

[[ 0.08 ]
 [-0.011]
 [-0.027]]

[[ 0.07 ]
 [-0.046]
 [-0.041]]

[[ 0.08 ]
 [-0.014]
 [ 0.006]]

[[ 0.07 ]
 [-0.047]
 [-0.054]]
```

```
[[ 0.08 ]
 [-0.015]
 [-0.007]]

[[0.09 ]
 [0.016]
 [0.042]]

[[ 0.08 ]
 [-0.017]
 [-0.018]]

[[ 0.07 ]
 [-0.047]
 [-0.032]]

[[ 0.08 ]
 [-0.015]
 [ 0.015]]

[[ 0.07 ]
 [-0.048]
 [-0.045]]

[[ 0.08 ]
 [-0.016]
 [ 0.002]]

[[ 0.07 ]
 [-0.049]
 [-0.058]]

[[ 0.08 ]
 [-0.017]
 [-0.011]]

[[0.09 ]
 [0.015]
 [0.034]]

[[ 0.08 ]
 [-0.018]
 [-0.026]]

[[ 0.07 ]
 [-0.041]
 [-0.039]]

[[ 0.08 ]
 [-0.009]
 [ 0.008]]

[[ 0.07 ]
 [-0.042]
 [-0.052]]

[[ 0.08 ]
 [-0.01 ]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.065]]

[[ 0.08 ]
 [-0.011]
 [-0.018]]

[[0.09 ]
 [0.021]
 [0.027]]

[[ 0.08 ]
 [-0.012]
 [-0.033]]
```

```
[[ 0.07 ]
 [-0.042]
 [-0.044]]

[[ 0.08 ]
 [-0.01 ]
 [ 0.003]]

[[ 0.07 ]
 [-0.043]
 [-0.057]]

[[ 0.08 ]
 [-0.011]
 [-0.01 ]]

[[0.09 ]
 [0.021]
 [0.035]]

[[ 0.08 ]
 [-0.012]
 [-0.025]]

[[ 0.07 ]
 [-0.047]
 [-0.039]]

[[ 0.08 ]
 [-0.015]
 [ 0.008]]

[[ 0.07 ]
 [-0.048]
 [-0.052]]

[[ 0.08 ]
 [-0.016]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.056]]

[[ 0.08 ]
 [-0.011]
 [-0.009]]

[[ 0.07 ]
 [-0.038]
 [-0.06 ]]

[[ 0.08 ]
 [-0.006]
 [-0.013]]

[[0.09 ]
 [0.025]
 [0.036]]

[[ 0.08 ]
 [-0.008]
 [-0.024]]

[[ 0.07 ]
 [-0.043]
 [-0.038]]

[[ 0.08 ]
 [-0.011]
 [ 0.009]]

[[ 0.07 ]
 [-0.044]
```

```
 [-0.051]]

[[ 0.08 ]
 [-0.012]
 [-0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.064]]

[[ 0.08 ]
 [-0.013]
 [-0.017]]

[[0.09 ]
 [0.019]
 [0.028]]

[[ 0.08 ]
 [-0.014]
 [-0.032]]

[[ 0.07 ]
 [-0.044]
 [-0.043]]

[[ 0.08 ]
 [-0.012]
 [ 0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.056]]

[[ 0.08 ]
 [-0.013]
 [-0.009]]

[[0.09 ]
 [0.019]
 [0.036]]

[[ 0.08 ]
 [-0.014]
 [-0.024]]

[[ 0.07 ]
 [-0.044]
 [-0.038]]

[[ 0.08 ]
 [-0.012]
 [ 0.009]]

[[ 0.07 ]
 [-0.045]
 [-0.051]]

[[ 0.08 ]
 [-0.013]
 [-0.004]]

[[ 0.07 ]
 [-0.046]
 [-0.064]]

[[ 0.08 ]
 [-0.014]
 [-0.017]]

[[0.09 ]
 [0.018]
 [0.028]]

[[ 0.08 ]
```

```
 [-0.015]
 [-0.032]]

[[ 0.07 ]
 [-0.038]
 [-0.045]]

[[ 0.08 ]
 [-0.006]
 [ 0.002]]

[[ 0.07 ]
 [-0.039]
 [-0.058]]

[[ 0.08 ]
 [-0.007]
 [-0.011]]

[[ 0.07 ]
 [-0.034]
 [-0.062]]

[[ 0.08 ]
 [-0.002]
 [-0.015]]

[[0.09 ]
 [0.028]
 [0.035]]

[[ 0.08 ]
 [-0.005]
 [-0.025]]

[[ 0.07 ]
 [-0.04 ]
 [-0.039]]

[[ 0.08 ]
 [-0.008]
 [ 0.008]]

[[ 0.07 ]
 [-0.041]
 [-0.052]]

[[ 0.08 ]
 [-0.009]
 [-0.005]]

[[ 0.07 ]
 [-0.042]
 [-0.065]]

[[ 0.08 ]
 [-0.01 ]
 [-0.018]]

[[0.09 ]
 [0.022]
 [0.027]]

[[ 0.08 ]
 [-0.011]
 [-0.033]]

[[ 0.07 ]
 [-0.041]
 [-0.047]]

[[ 8.00000000e-02]
 [-9.00000000e-03]
 [ 8.72357742e-14]]
```

```
[[ 0.07 ]
 [-0.042]
 [-0.06 ]]

[[ 0.08 ]
 [-0.01 ]
 [-0.013]]

[[0.09 ]
 [0.022]
 [0.032]]

[[ 0.08 ]
 [-0.011]
 [-0.028]]

[[ 0.07 ]
 [-0.046]
 [-0.042]]

[[ 0.08 ]
 [-0.014]
 [ 0.005]]

[[ 0.07 ]
 [-0.047]
 [-0.055]]

[[ 0.08 ]
 [-0.015]
 [-0.008]]

[[0.09 ]
 [0.017]
 [0.037]]

[[ 0.08 ]
 [-0.016]
 [-0.023]]

[[ 0.07 ]
 [-0.045]
 [-0.037]]

[[ 0.08 ]
 [-0.013]
 [ 0.01 ]]

[[ 0.07 ]
 [-0.046]
 [-0.05 ]]

[[ 0.08 ]
 [-0.014]
 [-0.003]]

[[ 0.07 ]
 [-0.047]
 [-0.063]]

[[ 0.08 ]
 [-0.015]
 [-0.016]]

[[0.09 ]
 [0.017]
 [0.029]]

[[ 0.08 ]
 [-0.016]
 [-0.031]]

[[ 0.07 ]
 [-0.039]
 [-0.044]]
```

```
[[ 0.08 ]
 [-0.007]
 [ 0.003]]

[[ 0.07 ]
 [-0.04 ]
 [-0.057]]

[[ 0.08 ]
 [-0.008]
 [-0.01 ]]

[[ 0.07 ]
 [-0.035]
 [-0.061]]

[[ 0.08 ]
 [-0.003]
 [-0.014]]

[[ 0.07 ]
 [-0.03 ]
 [-0.065]]

[[ 0.08 ]
 [ 0.002]
 [-0.018]]

[[0.09 ]
 [0.033]
 [0.031]]

[[ 8.00000000e-02]
 [-3.28487237e-14]
 [-2.90000000e-02]]

[[ 0.07 ]
 [-0.035]
 [-0.043]]

[[ 0.08 ]
 [-0.003]
 [ 0.004]]

[[ 0.07 ]
 [-0.036]
 [-0.056]]

[[ 0.08 ]
 [-0.004]
 [-0.009]]

[[ 0.07 ]
 [-0.037]
 [-0.069]]

[[ 0.08 ]
 [-0.005]
 [-0.022]]

[[0.09 ]
 [0.027]
 [0.023]]

[[ 0.08 ]
 [-0.006]
 [-0.037]]

[[ 0.07 ]
 [-0.041]
 [-0.051]]

[[ 0.08 ]
 [-0.009]
```

```
 [-0.004]]

[[ 0.07 ]
 [-0.042]
 [-0.064]]

[[ 0.08 ]
 [-0.01 ]
 [-0.017]]

[[0.09 ]
 [0.022]
 [0.028]]

[[ 0.08 ]
 [-0.011]
 [-0.032]]

[[ 0.07 ]
 [-0.041]
 [-0.046]]

[[ 0.08 ]
 [-0.009]
 [ 0.001]]

[[ 0.07 ]
 [-0.042]
 [-0.059]]

[[ 0.08 ]
 [-0.01 ]
 [-0.012]]

[[0.09 ]
 [0.022]
 [0.033]]

[[ 0.08 ]
 [-0.011]
 [-0.027]]

[[ 0.07 ]
 [-0.046]
 [-0.041]]

[[ 0.08 ]
 [-0.014]
 [ 0.006]]

[[ 0.07 ]
 [-0.047]
 [-0.054]]

[[ 0.08 ]
 [-0.015]
 [-0.007]]

[[0.09 ]
 [0.016]
 [0.042]]

[[ 0.08 ]
 [-0.017]
 [-0.018]]

[[ 0.07 ]
 [-0.047]
 [-0.032]]

[[ 0.08 ]
 [-0.015]
 [ 0.015]]

[[ 0.07 ]
```

```
 [-0.048]
 [-0.045]]

[[ 0.08 ]
 [-0.016]
 [ 0.002]]

[[ 0.07 ]
 [-0.049]
 [-0.058]]

[[ 0.08 ]
 [-0.017]
 [-0.011]]

[[0.09 ]
 [0.015]
 [0.034]]

[[ 0.08 ]
 [-0.018]
 [-0.026]]

[[ 0.07 ]
 [-0.041]
 [-0.039]]

[[ 0.08 ]
 [-0.009]
 [ 0.008]]

[[ 0.07 ]
 [-0.042]
 [-0.052]]

[[ 0.08 ]
 [-0.01 ]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.065]]

[[ 0.08 ]
 [-0.011]
 [-0.018]]

[[0.09 ]
 [0.021]
 [0.027]]

[[ 0.08 ]
 [-0.012]
 [-0.033]]

[[ 0.07 ]
 [-0.042]
 [-0.044]]

[[ 0.08 ]
 [-0.01 ]
 [ 0.003]]

[[ 0.07 ]
 [-0.043]
 [-0.057]]

[[ 0.08 ]
 [-0.011]
 [-0.01 ]]

[[0.09 ]
 [0.021]
 [0.035]]
```

```
[[ 0.08 ]
 [-0.012]
 [-0.025]]

[[ 0.07 ]
 [-0.047]
 [-0.039]]

[[ 0.08 ]
 [-0.015]
 [ 0.008]]

[[ 0.07 ]
 [-0.048]
 [-0.052]]

[[ 0.08 ]
 [-0.016]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.056]]

[[ 0.08 ]
 [-0.011]
 [-0.009]]

[[ 0.07 ]
 [-0.038]
 [-0.06 ]]

[[ 0.08 ]
 [-0.006]
 [-0.013]]

[[0.09 ]
 [0.025]
 [0.036]]

[[ 0.08 ]
 [-0.008]
 [-0.024]]

[[ 0.07 ]
 [-0.043]
 [-0.038]]

[[ 0.08 ]
 [-0.011]
 [ 0.009]]

[[ 0.07 ]
 [-0.044]
 [-0.051]]

[[ 0.08 ]
 [-0.012]
 [-0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.064]]

[[ 0.08 ]
 [-0.013]
 [-0.017]]

[[0.09 ]
 [0.019]
 [0.028]]

[[ 0.08 ]
 [-0.014]
 [-0.032]]
```

```
[[ 0.07 ]
 [-0.044]
 [-0.043]]

[[ 0.08 ]
 [-0.012]
 [ 0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.056]]

[[ 0.08 ]
 [-0.013]
 [-0.009]]

[[0.09 ]
 [0.019]
 [0.036]]

[[ 0.08 ]
 [-0.014]
 [-0.024]]

[[ 0.07 ]
 [-0.044]
 [-0.038]]

[[ 0.08 ]
 [-0.012]
 [ 0.009]]

[[ 0.07 ]
 [-0.045]
 [-0.051]]

[[ 0.08 ]
 [-0.013]
 [-0.004]]

[[ 0.07 ]
 [-0.046]
 [-0.064]]

[[ 0.08 ]
 [-0.014]
 [-0.017]]

[[0.09 ]
 [0.018]
 [0.028]]

[[ 0.08 ]
 [-0.015]
 [-0.032]]

[[ 0.07 ]
 [-0.038]
 [-0.045]]

[[ 0.08 ]
 [-0.006]
 [ 0.002]]

[[ 0.07 ]
 [-0.039]
 [-0.058]]

[[ 0.08 ]
 [-0.007]
 [-0.011]]

[[ 0.07 ]
 [-0.034]
```

```
 [-0.062]]

[[ 0.08 ]
 [-0.002]
 [-0.015]]

[[0.09 ]
 [0.028]
 [0.035]]

[[ 0.08 ]
 [-0.005]
 [-0.025]]

[[ 0.07 ]
 [-0.04 ]
 [-0.039]]

[[ 0.08 ]
 [-0.008]
 [ 0.008]]

[[ 0.07 ]
 [-0.041]
 [-0.052]]

[[ 0.08 ]
 [-0.009]
 [-0.005]]

[[ 0.07 ]
 [-0.042]
 [-0.065]]

[[ 0.08 ]
 [-0.01 ]
 [-0.018]]

[[0.09 ]
 [0.022]
 [0.027]]

[[ 0.08 ]
 [-0.011]
 [-0.033]]

[[ 0.07 ]
 [-0.041]
 [-0.047]]

[[ 8.00000000e-02]
 [-9.00000000e-03]
 [ 8.73467965e-14]]

[[ 0.07 ]
 [-0.042]
 [-0.06 ]]

[[ 0.08 ]
 [-0.01 ]
 [-0.013]]

[[0.09 ]
 [0.022]
 [0.032]]

[[ 0.08 ]
 [-0.011]
 [-0.028]]

[[ 0.07 ]
 [-0.046]
 [-0.042]]

[[ 0.08 ]
```

```
 [-0.014]
 [ 0.005]]

[[ 0.07 ]
 [-0.047]
 [-0.055]]

[[ 0.08 ]
 [-0.015]
 [-0.008]]

[[0.09 ]
 [0.017]
 [0.037]]

[[ 0.08 ]
 [-0.016]
 [-0.023]]

[[ 0.07 ]
 [-0.045]
 [-0.037]]

[[ 0.08 ]
 [-0.013]
 [ 0.01 ]]

[[ 0.07 ]
 [-0.046]
 [-0.05 ]]

[[ 0.08 ]
 [-0.014]
 [-0.003]]

[[ 0.07 ]
 [-0.047]
 [-0.063]]

[[ 0.08 ]
 [-0.015]
 [-0.016]]

[[0.09 ]
 [0.017]
 [0.029]]

[[ 0.08 ]
 [-0.016]
 [-0.031]]

[[ 0.07 ]
 [-0.039]
 [-0.044]]

[[ 0.08 ]
 [-0.007]
 [ 0.003]]

[[ 0.07 ]
 [-0.04 ]
 [-0.057]]

[[ 0.08 ]
 [-0.008]
 [-0.01 ]]

[[ 0.07 ]
 [-0.035]
 [-0.061]]

[[ 0.08 ]
 [-0.003]
 [-0.014]]
```

```
[[ 0.07 ]
 [-0.03 ]
 [-0.065]]

[[ 0.08 ]
 [ 0.002]
 [-0.018]]

[[0.09 ]
 [0.033]
 [0.031]]

[[ 8.00000000e-02]
 [-3.28903571e-14]
 [-2.90000000e-02]]

[[ 0.07 ]
 [-0.035]
 [-0.043]]

[[ 0.08 ]
 [-0.003]
 [ 0.004]]

[[ 0.07 ]
 [-0.036]
 [-0.056]]

[[ 0.08 ]
 [-0.004]
 [-0.009]]

[[ 0.07 ]
 [-0.037]
 [-0.069]]

[[ 0.08 ]
 [-0.005]
 [-0.022]]

[[0.09 ]
 [0.027]
 [0.023]]

[[ 0.08 ]
 [-0.006]
 [-0.037]]

[[ 0.07 ]
 [-0.041]
 [-0.051]]

[[ 0.08 ]
 [-0.009]
 [-0.004]]

[[ 0.07 ]
 [-0.042]
 [-0.064]]

[[ 0.08 ]
 [-0.01 ]
 [-0.017]]

[[0.09 ]
 [0.022]
 [0.028]]

[[ 0.08 ]
 [-0.011]
 [-0.032]]

[[ 0.07 ]
 [-0.041]
 [-0.046]]
```

```
[[ 0.08 ]
 [-0.009]
 [ 0.001]]

[[ 0.07 ]
 [-0.042]
 [-0.059]]

[[ 0.08 ]
 [-0.01 ]
 [-0.012]]

[[0.09 ]
 [0.022]
 [0.033]]

[[ 0.08 ]
 [-0.011]
 [-0.027]]

[[ 0.07 ]
 [-0.046]
 [-0.041]]

[[ 0.08 ]
 [-0.014]
 [ 0.006]]

[[ 0.07 ]
 [-0.047]
 [-0.054]]

[[ 0.08 ]
 [-0.015]
 [-0.007]]

[[0.09 ]
 [0.016]
 [0.042]]

[[ 0.08 ]
 [-0.017]
 [-0.018]]

[[ 0.07 ]
 [-0.047]
 [-0.032]]

[[ 0.08 ]
 [-0.015]
 [ 0.015]]

[[ 0.07 ]
 [-0.048]
 [-0.045]]

[[ 0.08 ]
 [-0.016]
 [ 0.002]]

[[ 0.07 ]
 [-0.049]
 [-0.058]]

[[ 0.08 ]
 [-0.017]
 [-0.011]]

[[0.09 ]
 [0.015]
 [0.034]]

[[ 0.08 ]
 [-0.018]
```

```
 [-0.026]]

[[ 0.07 ]
 [-0.041]
 [-0.039]]

[[ 0.08 ]
 [-0.009]
 [ 0.008]]

[[ 0.07 ]
 [-0.042]
 [-0.052]]

[[ 0.08 ]
 [-0.01 ]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.065]]

[[ 0.08 ]
 [-0.011]
 [-0.018]]

[[0.09 ]
 [0.021]
 [0.027]]

[[ 0.08 ]
 [-0.012]
 [-0.033]]

[[ 0.07 ]
 [-0.042]
 [-0.044]]

[[ 0.08 ]
 [-0.01 ]
 [ 0.003]]

[[ 0.07 ]
 [-0.043]
 [-0.057]]

[[ 0.08 ]
 [-0.011]
 [-0.01 ]]

[[0.09 ]
 [0.021]
 [0.035]]

[[ 0.08 ]
 [-0.012]
 [-0.025]]

[[ 0.07 ]
 [-0.047]
 [-0.039]]

[[ 0.08 ]
 [-0.015]
 [ 0.008]]

[[ 0.07 ]
 [-0.048]
 [-0.052]]

[[ 0.08 ]
 [-0.016]
 [-0.005]]

[[ 0.07 ]
```

```
 [-0.043]
 [-0.056]]

[[ 0.08 ]
 [-0.011]
 [-0.009]]

[[ 0.07 ]
 [-0.038]
 [-0.06 ]]

[[ 0.08 ]
 [-0.006]
 [-0.013]]

[[0.09 ]
 [0.025]
 [0.036]]

[[ 0.08 ]
 [-0.008]
 [-0.024]]

[[ 0.07 ]
 [-0.043]
 [-0.038]]

[[ 0.08 ]
 [-0.011]
 [ 0.009]]

[[ 0.07 ]
 [-0.044]
 [-0.051]]

[[ 0.08 ]
 [-0.012]
 [-0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.064]]

[[ 0.08 ]
 [-0.013]
 [-0.017]]

[[0.09 ]
 [0.019]
 [0.028]]

[[ 0.08 ]
 [-0.014]
 [-0.032]]

[[ 0.07 ]
 [-0.044]
 [-0.043]]

[[ 0.08 ]
 [-0.012]
 [ 0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.056]]

[[ 0.08 ]
 [-0.013]
 [-0.009]]

[[0.09 ]
 [0.019]
 [0.036]]
```

```
[[ 0.08 ]
 [-0.014]
 [-0.024]]

[[ 0.07 ]
 [-0.044]
 [-0.038]]

[[ 0.08 ]
 [-0.012]
 [ 0.009]]

[[ 0.07 ]
 [-0.045]
 [-0.051]]

[[ 0.08 ]
 [-0.013]
 [-0.004]]

[[ 0.07 ]
 [-0.046]
 [-0.064]]

[[ 0.08 ]
 [-0.014]
 [-0.017]]

[[0.09 ]
 [0.018]
 [0.028]]

[[ 0.08 ]
 [-0.015]
 [-0.032]]

[[ 0.07 ]
 [-0.038]
 [-0.045]]

[[ 0.08 ]
 [-0.006]
 [ 0.002]]

[[ 0.07 ]
 [-0.039]
 [-0.058]]

[[ 0.08 ]
 [-0.007]
 [-0.011]]

[[ 0.07 ]
 [-0.034]
 [-0.062]]

[[ 0.08 ]
 [-0.002]
 [-0.015]]

[[0.09 ]
 [0.028]
 [0.035]]

[[ 0.08 ]
 [-0.005]
 [-0.025]]

[[ 0.07 ]
 [-0.04 ]
 [-0.039]]

[[ 0.08 ]
 [-0.008]
 [ 0.008]]
```

```
[[ 0.07 ]
 [-0.041]
 [-0.052]]

[[ 0.08 ]
 [-0.009]
 [-0.005]]

[[ 0.07 ]
 [-0.042]
 [-0.065]]

[[ 0.08 ]
 [-0.01 ]
 [-0.018]]

[[0.09 ]
 [0.022]
 [0.027]]

[[ 0.08 ]
 [-0.011]
 [-0.033]]

[[ 0.07 ]
 [-0.041]
 [-0.047]]

[[ 8.00000000e-02]
 [-9.00000000e-03]
 [ 8.74578188e-14]]

[[ 0.07 ]
 [-0.042]
 [-0.06 ]]

[[ 0.08 ]
 [-0.01 ]
 [-0.013]]

[[0.09 ]
 [0.022]
 [0.032]]

[[ 0.08 ]
 [-0.011]
 [-0.028]]

[[ 0.07 ]
 [-0.046]
 [-0.042]]

[[ 0.08 ]
 [-0.014]
 [ 0.005]]

[[ 0.07 ]
 [-0.047]
 [-0.055]]

[[ 0.08 ]
 [-0.015]
 [-0.008]]

[[0.09 ]
 [0.017]
 [0.037]]

[[ 0.08 ]
 [-0.016]
 [-0.023]]

[[ 0.07 ]
 [-0.045]
```

```
  [-0.037]]

 [[ 0.08 ]
  [-0.013]
  [ 0.01 ]]

 [[ 0.07 ]
  [-0.046]
  [-0.05 ]]

 [[ 0.08 ]
  [-0.014]
  [-0.003]]

 [[ 0.07 ]
  [-0.047]
  [-0.063]]

 [[ 0.08 ]
  [-0.015]
  [-0.016]]

 [[0.09 ]
  [0.017]
  [0.029]]

 [[ 0.08 ]
  [-0.016]
  [-0.031]]

 [[ 0.07 ]
  [-0.039]
  [-0.044]]

 [[ 0.08 ]
  [-0.007]
  [ 0.003]]

 [[ 0.07 ]
  [-0.04 ]
  [-0.057]]

 [[ 0.08 ]
  [-0.008]
  [-0.01 ]]

 [[ 0.07 ]
  [-0.035]
  [-0.061]]

 [[ 0.08 ]
  [-0.003]
  [-0.014]]

 [[ 0.07 ]
  [-0.03 ]
  [-0.065]]

 [[ 0.08 ]
  [ 0.002]
  [-0.018]]

 [[0.09 ]
  [0.033]
  [0.031]]

 [[ 8.00000000e-02]
  [-3.29319905e-14]
  [-2.90000000e-02]]

 [[ 0.07 ]
  [-0.035]
  [-0.043]]

 [[ 0.08 ]
```

```
 [-0.003]
 [ 0.004]]

[[ 0.07 ]
 [-0.036]
 [-0.056]]

[[ 0.08 ]
 [-0.004]
 [-0.009]]

[[ 0.07 ]
 [-0.037]
 [-0.069]]

[[ 0.08 ]
 [-0.005]
 [-0.022]]

[[0.09 ]
 [0.027]
 [0.023]]

[[ 0.08 ]
 [-0.006]
 [-0.037]]

[[ 0.07 ]
 [-0.041]
 [-0.051]]

[[ 0.08 ]
 [-0.009]
 [-0.004]]

[[ 0.07 ]
 [-0.042]
 [-0.064]]

[[ 0.08 ]
 [-0.01 ]
 [-0.017]]

[[0.09 ]
 [0.022]
 [0.028]]

[[ 0.08 ]
 [-0.011]
 [-0.032]]

[[ 0.07 ]
 [-0.041]
 [-0.046]]

[[ 0.08 ]
 [-0.009]
 [ 0.001]]

[[ 0.07 ]
 [-0.042]
 [-0.059]]

[[ 0.08 ]
 [-0.01 ]
 [-0.012]]

[[0.09 ]
 [0.022]
 [0.033]]

[[ 0.08 ]
 [-0.011]
 [-0.027]]
```

```
[[ 0.07 ]
 [-0.046]
 [-0.041]]

[[ 0.08 ]
 [-0.014]
 [ 0.006]]

[[ 0.07 ]
 [-0.047]
 [-0.054]]

[[ 0.08 ]
 [-0.015]
 [-0.007]]

[[0.09 ]
 [0.016]
 [0.042]]

[[ 0.08 ]
 [-0.017]
 [-0.018]]

[[ 0.07 ]
 [-0.047]
 [-0.032]]

[[ 0.08 ]
 [-0.015]
 [ 0.015]]

[[ 0.07 ]
 [-0.048]
 [-0.045]]

[[ 0.08 ]
 [-0.016]
 [ 0.002]]

[[ 0.07 ]
 [-0.049]
 [-0.058]]

[[ 0.08 ]
 [-0.017]
 [-0.011]]

[[0.09 ]
 [0.015]
 [0.034]]

[[ 0.08 ]
 [-0.018]
 [-0.026]]

[[ 0.07 ]
 [-0.041]
 [-0.039]]

[[ 0.08 ]
 [-0.009]
 [ 0.008]]

[[ 0.07 ]
 [-0.042]
 [-0.052]]

[[ 0.08 ]
 [-0.01 ]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.065]]
```

```
[[ 0.08 ]
 [-0.011]
 [-0.018]]

[[0.09 ]
 [0.021]
 [0.027]]

[[ 0.08 ]
 [-0.012]
 [-0.033]]

[[ 0.07 ]
 [-0.042]
 [-0.044]]

[[ 0.08 ]
 [-0.01 ]
 [ 0.003]]

[[ 0.07 ]
 [-0.043]
 [-0.057]]

[[ 0.08 ]
 [-0.011]
 [-0.01 ]]

[[0.09 ]
 [0.021]
 [0.035]]

[[ 0.08 ]
 [-0.012]
 [-0.025]]

[[ 0.07 ]
 [-0.047]
 [-0.039]]

[[ 0.08 ]
 [-0.015]
 [ 0.008]]

[[ 0.07 ]
 [-0.048]
 [-0.052]]

[[ 0.08 ]
 [-0.016]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.056]]

[[ 0.08 ]
 [-0.011]
 [-0.009]]

[[ 0.07 ]
 [-0.038]
 [-0.06 ]]

[[ 0.08 ]
 [-0.006]
 [-0.013]]

[[0.09 ]
 [0.025]
 [0.036]]

[[ 0.08 ]
 [-0.008]
```

```
 [-0.024]]

[[ 0.07 ]
 [-0.043]
 [-0.038]]

[[ 0.08 ]
 [-0.011]
 [ 0.009]]

[[ 0.07 ]
 [-0.044]
 [-0.051]]

[[ 0.08 ]
 [-0.012]
 [-0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.064]]

[[ 0.08 ]
 [-0.013]
 [-0.017]]

[[0.09 ]
 [0.019]
 [0.028]]

[[ 0.08 ]
 [-0.014]
 [-0.032]]

[[ 0.07 ]
 [-0.044]
 [-0.043]]

[[ 0.08 ]
 [-0.012]
 [ 0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.056]]

[[ 0.08 ]
 [-0.013]
 [-0.009]]

[[0.09 ]
 [0.019]
 [0.036]]

[[ 0.08 ]
 [-0.014]
 [-0.024]]

[[ 0.07 ]
 [-0.044]
 [-0.038]]

[[ 0.08 ]
 [-0.012]
 [ 0.009]]

[[ 0.07 ]
 [-0.045]
 [-0.051]]

[[ 0.08 ]
 [-0.013]
 [-0.004]]

[[ 0.07 ]
```

```
 [-0.046]
 [-0.064]]

[[ 0.08 ]
 [-0.014]
 [-0.017]]

[[0.09 ]
 [0.018]
 [0.028]]

[[ 0.08 ]
 [-0.015]
 [-0.032]]

[[ 0.07 ]
 [-0.038]
 [-0.045]]

[[ 0.08 ]
 [-0.006]
 [ 0.002]]

[[ 0.07 ]
 [-0.039]
 [-0.058]]

[[ 0.08 ]
 [-0.007]
 [-0.011]]

[[ 0.07 ]
 [-0.034]
 [-0.062]]

[[ 0.08 ]
 [-0.002]
 [-0.015]]

[[0.09 ]
 [0.028]
 [0.035]]

[[ 0.08 ]
 [-0.005]
 [-0.025]]

[[ 0.07 ]
 [-0.04 ]
 [-0.039]]

[[ 0.08 ]
 [-0.008]
 [ 0.008]]

[[ 0.07 ]
 [-0.041]
 [-0.052]]

[[ 0.08 ]
 [-0.009]
 [-0.005]]

[[ 0.07 ]
 [-0.042]
 [-0.065]]

[[ 0.08 ]
 [-0.01 ]
 [-0.018]]

[[0.09 ]
 [0.022]
 [0.027]]
```

```
[[ 0.08 ]
 [-0.011]
 [-0.033]]

[[ 0.07 ]
 [-0.041]
 [-0.047]]

[[ 8.00000000e-02]
 [-9.00000000e-03]
 [ 8.75688411e-14]]

[[ 0.07 ]
 [-0.042]
 [-0.06 ]]

[[ 0.08 ]
 [-0.01 ]
 [-0.013]]

[[0.09 ]
 [0.022]
 [0.032]]

[[ 0.08 ]
 [-0.011]
 [-0.028]]

[[ 0.07 ]
 [-0.046]
 [-0.042]]

[[ 0.08 ]
 [-0.014]
 [ 0.005]]

[[ 0.07 ]
 [-0.047]
 [-0.055]]

[[ 0.08 ]
 [-0.015]
 [-0.008]]

[[0.09 ]
 [0.017]
 [0.037]]

[[ 0.08 ]
 [-0.016]
 [-0.023]]

[[ 0.07 ]
 [-0.045]
 [-0.037]]

[[ 0.08 ]
 [-0.013]
 [ 0.01 ]]

[[ 0.07 ]
 [-0.046]
 [-0.05 ]]

[[ 0.08 ]
 [-0.014]
 [-0.003]]

[[ 0.07 ]
 [-0.047]
 [-0.063]]

[[ 0.08 ]
 [-0.015]
 [-0.016]]
```

```
[[0.09 ]
 [0.017]
 [0.029]]

[[ 0.08 ]
 [-0.016]
 [-0.031]]

[[ 0.07 ]
 [-0.039]
 [-0.044]]

[[ 0.08 ]
 [-0.007]
 [ 0.003]]

[[ 0.07 ]
 [-0.04 ]
 [-0.057]]

[[ 0.08 ]
 [-0.008]
 [-0.01 ]]

[[ 0.07 ]
 [-0.035]
 [-0.061]]

[[ 0.08 ]
 [-0.003]
 [-0.014]]

[[ 0.07 ]
 [-0.03 ]
 [-0.065]]

[[ 0.08 ]
 [ 0.002]
 [-0.018]]

[[0.09 ]
 [0.033]
 [0.031]]

[[ 8.00000000e-02]
 [-3.29736238e-14]
 [-2.90000000e-02]]

[[ 0.07 ]
 [-0.035]
 [-0.043]]

[[ 0.08 ]
 [-0.003]
 [ 0.004]]

[[ 0.07 ]
 [-0.036]
 [-0.056]]

[[ 0.08 ]
 [-0.004]
 [-0.009]]

[[ 0.07 ]
 [-0.037]
 [-0.069]]

[[ 0.08 ]
 [-0.005]
 [-0.022]]

[[0.09 ]
 [0.027]
```

```
 [0.023]]

[[ 0.08 ]
 [-0.006]
 [-0.037]]

[[ 0.07 ]
 [-0.041]
 [-0.051]]

[[ 0.08 ]
 [-0.009]
 [-0.004]]

[[ 0.07 ]
 [-0.042]
 [-0.064]]

[[ 0.08 ]
 [-0.01 ]
 [-0.017]]

[[0.09 ]
 [0.022]
 [0.028]]

[[ 0.08 ]
 [-0.011]
 [-0.032]]

[[ 0.07 ]
 [-0.041]
 [-0.046]]

[[ 0.08 ]
 [-0.009]
 [ 0.001]]

[[ 0.07 ]
 [-0.042]
 [-0.059]]

[[ 0.08 ]
 [-0.01 ]
 [-0.012]]

[[0.09 ]
 [0.022]
 [0.033]]

[[ 0.08 ]
 [-0.011]
 [-0.027]]

[[ 0.07 ]
 [-0.046]
 [-0.041]]

[[ 0.08 ]
 [-0.014]
 [ 0.006]]

[[ 0.07 ]
 [-0.047]
 [-0.054]]

[[ 0.08 ]
 [-0.015]
 [-0.007]]

[[0.09 ]
 [0.016]
 [0.042]]

[[ 0.08 ]
```

```
 [-0.017]
 [-0.018]]

[[ 0.07 ]
 [-0.047]
 [-0.032]]

[[ 0.08 ]
 [-0.015]
 [ 0.015]]

[[ 0.07 ]
 [-0.048]
 [-0.045]]

[[ 0.08 ]
 [-0.016]
 [ 0.002]]

[[ 0.07 ]
 [-0.049]
 [-0.058]]

[[ 0.08 ]
 [-0.017]
 [-0.011]]

[[0.09 ]
 [0.015]
 [0.034]]

[[ 0.08 ]
 [-0.018]
 [-0.026]]

[[ 0.07 ]
 [-0.041]
 [-0.039]]

[[ 0.08 ]
 [-0.009]
 [ 0.008]]

[[ 0.07 ]
 [-0.042]
 [-0.052]]

[[ 0.08 ]
 [-0.01 ]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.065]]

[[ 0.08 ]
 [-0.011]
 [-0.018]]

[[0.09 ]
 [0.021]
 [0.027]]

[[ 0.08 ]
 [-0.012]
 [-0.033]]

[[ 0.07 ]
 [-0.042]
 [-0.044]]

[[ 0.08 ]
 [-0.01 ]
 [ 0.003]]
```

```
[[ 0.07 ]
 [-0.043]
 [-0.057]]

[[ 0.08 ]
 [-0.011]
 [-0.01 ]]

[[0.09 ]
 [0.021]
 [0.035]]

[[ 0.08 ]
 [-0.012]
 [-0.025]]

[[ 0.07 ]
 [-0.047]
 [-0.039]]

[[ 0.08 ]
 [-0.015]
 [ 0.008]]

[[ 0.07 ]
 [-0.048]
 [-0.052]]

[[ 0.08 ]
 [-0.016]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.056]]

[[ 0.08 ]
 [-0.011]
 [-0.009]]

[[ 0.07 ]
 [-0.038]
 [-0.06 ]]

[[ 0.08 ]
 [-0.006]
 [-0.013]]

[[0.09 ]
 [0.025]
 [0.036]]

[[ 0.08 ]
 [-0.008]
 [-0.024]]

[[ 0.07 ]
 [-0.043]
 [-0.038]]

[[ 0.08 ]
 [-0.011]
 [ 0.009]]

[[ 0.07 ]
 [-0.044]
 [-0.051]]

[[ 0.08 ]
 [-0.012]
 [-0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.064]]
```

```
[[ 0.08 ]
 [-0.013]
 [-0.017]]

[[0.09 ]
 [0.019]
 [0.028]]

[[ 0.08 ]
 [-0.014]
 [-0.032]]

[[ 0.07 ]
 [-0.044]
 [-0.043]]

[[ 0.08 ]
 [-0.012]
 [ 0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.056]]

[[ 0.08 ]
 [-0.013]
 [-0.009]]

[[0.09 ]
 [0.019]
 [0.036]]

[[ 0.08 ]
 [-0.014]
 [-0.024]]

[[ 0.07 ]
 [-0.044]
 [-0.038]]

[[ 0.08 ]
 [-0.012]
 [ 0.009]]

[[ 0.07 ]
 [-0.045]
 [-0.051]]

[[ 0.08 ]
 [-0.013]
 [-0.004]]

[[ 0.07 ]
 [-0.046]
 [-0.064]]

[[ 0.08 ]
 [-0.014]
 [-0.017]]

[[0.09 ]
 [0.018]
 [0.028]]

[[ 0.08 ]
 [-0.015]
 [-0.032]]

[[ 0.07 ]
 [-0.038]
 [-0.045]]

[[ 0.08 ]
 [-0.006]
```

```
 [ 0.002]]

[[ 0.07 ]
 [-0.039]
 [-0.058]]

[[ 0.08 ]
 [-0.007]
 [-0.011]]

[[ 0.07 ]
 [-0.034]
 [-0.062]]

[[ 0.08 ]
 [-0.002]
 [-0.015]]

[[0.09 ]
 [0.028]
 [0.035]]

[[ 0.08 ]
 [-0.005]
 [-0.025]]

[[ 0.07 ]
 [-0.04 ]
 [-0.039]]

[[ 0.08 ]
 [-0.008]
 [ 0.008]]

[[ 0.07 ]
 [-0.041]
 [-0.052]]

[[ 0.08 ]
 [-0.009]
 [-0.005]]

[[ 0.07 ]
 [-0.042]
 [-0.065]]

[[ 0.08 ]
 [-0.01 ]
 [-0.018]]

[[0.09 ]
 [0.022]
 [0.027]]

[[ 0.08 ]
 [-0.011]
 [-0.033]]

[[ 0.07 ]
 [-0.041]
 [-0.047]]

[[ 8.00000000e-02]
 [-9.00000000e-03]
 [ 8.76798634e-14]]

[[ 0.07 ]
 [-0.042]
 [-0.06 ]]

[[ 0.08 ]
 [-0.01 ]
 [-0.013]]

[[0.09 ]
```

```
 [0.022]
 [0.032]]

[[ 0.08 ]
 [-0.011]
 [-0.028]]

[[ 0.07 ]
 [-0.046]
 [-0.042]]

[[ 0.08 ]
 [-0.014]
 [ 0.005]]

[[ 0.07 ]
 [-0.047]
 [-0.055]]

[[ 0.08 ]
 [-0.015]
 [-0.008]]

[[0.09 ]
 [0.017]
 [0.037]]

[[ 0.08 ]
 [-0.016]
 [-0.023]]

[[ 0.07 ]
 [-0.045]
 [-0.037]]

[[ 0.08 ]
 [-0.013]
 [ 0.01 ]]

[[ 0.07 ]
 [-0.046]
 [-0.05 ]]

[[ 0.08 ]
 [-0.014]
 [-0.003]]

[[ 0.07 ]
 [-0.047]
 [-0.063]]

[[ 0.08 ]
 [-0.015]
 [-0.016]]

[[0.09 ]
 [0.017]
 [0.029]]

[[ 0.08 ]
 [-0.016]
 [-0.031]]

[[ 0.07 ]
 [-0.039]
 [-0.044]]

[[ 0.08 ]
 [-0.007]
 [ 0.003]]

[[ 0.07 ]
 [-0.04 ]
 [-0.057]]
```

```
[[ 0.08 ]
 [-0.008]
 [-0.01 ]]

[[ 0.07 ]
 [-0.035]
 [-0.061]]

[[ 0.08 ]
 [-0.003]
 [-0.014]]

[[ 0.07 ]
 [-0.03 ]
 [-0.065]]

[[ 0.08 ]
 [ 0.002]
 [-0.018]]

[[0.09 ]
 [0.033]
 [0.031]]

[[ 8.00000000e-02]
 [-3.30152572e-14]
 [-2.90000000e-02]]

[[ 0.07 ]
 [-0.035]
 [-0.043]]

[[ 0.08 ]
 [-0.003]
 [ 0.004]]

[[ 0.07 ]
 [-0.036]
 [-0.056]]

[[ 0.08 ]
 [-0.004]
 [-0.009]]

[[ 0.07 ]
 [-0.037]
 [-0.069]]

[[ 0.08 ]
 [-0.005]
 [-0.022]]

[[0.09 ]
 [0.027]
 [0.023]]

[[ 0.08 ]
 [-0.006]
 [-0.037]]

[[ 0.07 ]
 [-0.041]
 [-0.051]]

[[ 0.08 ]
 [-0.009]
 [-0.004]]

[[ 0.07 ]
 [-0.042]
 [-0.064]]

[[ 0.08 ]
 [-0.01 ]
 [-0.017]]
```

```
[[0.09 ]
 [0.022]
 [0.028]]

[[ 0.08 ]
 [-0.011]
 [-0.032]]

[[ 0.07 ]
 [-0.041]
 [-0.046]]

[[ 0.08 ]
 [-0.009]
 [ 0.001]]

[[ 0.07 ]
 [-0.042]
 [-0.059]]

[[ 0.08 ]
 [-0.01 ]
 [-0.012]]

[[0.09 ]
 [0.022]
 [0.033]]

[[ 0.08 ]
 [-0.011]
 [-0.027]]

[[ 0.07 ]
 [-0.046]
 [-0.041]]

[[ 0.08 ]
 [-0.014]
 [ 0.006]]

[[ 0.07 ]
 [-0.047]
 [-0.054]]

[[ 0.08 ]
 [-0.015]
 [-0.007]]

[[0.09 ]
 [0.016]
 [0.042]]

[[ 0.08 ]
 [-0.017]
 [-0.018]]

[[ 0.07 ]
 [-0.047]
 [-0.032]]

[[ 0.08 ]
 [-0.015]
 [ 0.015]]

[[ 0.07 ]
 [-0.048]
 [-0.045]]

[[ 0.08 ]
 [-0.016]
 [ 0.002]]

[[ 0.07 ]
 [-0.049]
```

```
 [-0.058]]

[[ 0.08 ]
 [-0.017]
 [-0.011]]

[[0.09 ]
 [0.015]
 [0.034]]

[[ 0.08 ]
 [-0.018]
 [-0.026]]

[[ 0.07 ]
 [-0.041]
 [-0.039]]

[[ 0.08 ]
 [-0.009]
 [ 0.008]]

[[ 0.07 ]
 [-0.042]
 [-0.052]]

[[ 0.08 ]
 [-0.01 ]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.065]]

[[ 0.08 ]
 [-0.011]
 [-0.018]]

[[0.09 ]
 [0.021]
 [0.027]]

[[ 0.08 ]
 [-0.012]
 [-0.033]]

[[ 0.07 ]
 [-0.042]
 [-0.044]]

[[ 0.08 ]
 [-0.01 ]
 [ 0.003]]

[[ 0.07 ]
 [-0.043]
 [-0.057]]

[[ 0.08 ]
 [-0.011]
 [-0.01 ]]

[[0.09 ]
 [0.021]
 [0.035]]

[[ 0.08 ]
 [-0.012]
 [-0.025]]

[[ 0.07 ]
 [-0.047]
 [-0.039]]

[[ 0.08 ]
```

```
 [-0.015]
 [ 0.008]]

[[ 0.07 ]
 [-0.048]
 [-0.052]]

[[ 0.08 ]
 [-0.016]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.056]]

[[ 0.08 ]
 [-0.011]
 [-0.009]]

[[ 0.07 ]
 [-0.038]
 [-0.06 ]]

[[ 0.08 ]
 [-0.006]
 [-0.013]]

[[0.09 ]
 [0.025]
 [0.036]]

[[ 0.08 ]
 [-0.008]
 [-0.024]]

[[ 0.07 ]
 [-0.043]
 [-0.038]]

[[ 0.08 ]
 [-0.011]
 [ 0.009]]

[[ 0.07 ]
 [-0.044]
 [-0.051]]

[[ 0.08 ]
 [-0.012]
 [-0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.064]]

[[ 0.08 ]
 [-0.013]
 [-0.017]]

[[0.09 ]
 [0.019]
 [0.028]]

[[ 0.08 ]
 [-0.014]
 [-0.032]]

[[ 0.07 ]
 [-0.044]
 [-0.043]]

[[ 0.08 ]
 [-0.012]
 [ 0.004]]
```

```
[[ 0.07 ]
 [-0.045]
 [-0.056]]

[[ 0.08 ]
 [-0.013]
 [-0.009]]

[[0.09 ]
 [0.019]
 [0.036]]

[[ 0.08 ]
 [-0.014]
 [-0.024]]

[[ 0.07 ]
 [-0.044]
 [-0.038]]

[[ 0.08 ]
 [-0.012]
 [ 0.009]]

[[ 0.07 ]
 [-0.045]
 [-0.051]]

[[ 0.08 ]
 [-0.013]
 [-0.004]]

[[ 0.07 ]
 [-0.046]
 [-0.064]]

[[ 0.08 ]
 [-0.014]
 [-0.017]]

[[0.09 ]
 [0.018]
 [0.028]]

[[ 0.08 ]
 [-0.015]
 [-0.032]]

[[ 0.07 ]
 [-0.038]
 [-0.045]]

[[ 0.08 ]
 [-0.006]
 [ 0.002]]

[[ 0.07 ]
 [-0.039]
 [-0.058]]

[[ 0.08 ]
 [-0.007]
 [-0.011]]

[[ 0.07 ]
 [-0.034]
 [-0.062]]

[[ 0.08 ]
 [-0.002]
 [-0.015]]

[[0.09 ]
 [0.028]
 [0.035]]
```

```
[[ 0.08 ]
 [-0.005]
 [-0.025]]

[[ 0.07 ]
 [-0.04 ]
 [-0.039]]

[[ 0.08 ]
 [-0.008]
 [ 0.008]]

[[ 0.07 ]
 [-0.041]
 [-0.052]]

[[ 0.08 ]
 [-0.009]
 [-0.005]]

[[ 0.07 ]
 [-0.042]
 [-0.065]]

[[ 0.08 ]
 [-0.01 ]
 [-0.018]]

[[0.09 ]
 [0.022]
 [0.027]]

[[ 0.08 ]
 [-0.011]
 [-0.033]]

[[ 0.07 ]
 [-0.041]
 [-0.047]]

[[ 8.00000000e-02]
 [-9.00000000e-03]
 [ 8.77908857e-14]]

[[ 0.07 ]
 [-0.042]
 [-0.06 ]]

[[ 0.08 ]
 [-0.01 ]
 [-0.013]]

[[0.09 ]
 [0.022]
 [0.032]]

[[ 0.08 ]
 [-0.011]
 [-0.028]]

[[ 0.07 ]
 [-0.046]
 [-0.042]]

[[ 0.08 ]
 [-0.014]
 [ 0.005]]

[[ 0.07 ]
 [-0.047]
 [-0.055]]

[[ 0.08 ]
 [-0.015]
```

```
 [-0.008]]

[[0.09 ]
 [0.017]
 [0.037]]

[[ 0.08 ]
 [-0.016]
 [-0.023]]

[[ 0.07 ]
 [-0.045]
 [-0.037]]

[[ 0.08 ]
 [-0.013]
 [ 0.01 ]]

[[ 0.07 ]
 [-0.046]
 [-0.05 ]]

[[ 0.08 ]
 [-0.014]
 [-0.003]]

[[ 0.07 ]
 [-0.047]
 [-0.063]]

[[ 0.08 ]
 [-0.015]
 [-0.016]]

[[0.09 ]
 [0.017]
 [0.029]]

[[ 0.08 ]
 [-0.016]
 [-0.031]]

[[ 0.07 ]
 [-0.039]
 [-0.044]]

[[ 0.08 ]
 [-0.007]
 [ 0.003]]

[[ 0.07 ]
 [-0.04 ]
 [-0.057]]

[[ 0.08 ]
 [-0.008]
 [-0.01 ]]

[[ 0.07 ]
 [-0.035]
 [-0.061]]

[[ 0.08 ]
 [-0.003]
 [-0.014]]

[[ 0.07 ]
 [-0.03 ]
 [-0.065]]

[[ 0.08 ]
 [ 0.002]
 [-0.018]]

[[0.09 ]
```

```
 [0.033]
 [0.031]]

[[ 8.00000000e-02]
 [-3.30568906e-14]
 [-2.90000000e-02]]

[[ 0.07 ]
 [-0.035]
 [-0.043]]

[[ 0.08 ]
 [-0.003]
 [ 0.004]]

[[ 0.07 ]
 [-0.036]
 [-0.056]]

[[ 0.08 ]
 [-0.004]
 [-0.009]]

[[ 0.07 ]
 [-0.037]
 [-0.069]]

[[ 0.08 ]
 [-0.005]
 [-0.022]]

[[0.09 ]
 [0.027]
 [0.023]]

[[ 0.08 ]
 [-0.006]
 [-0.037]]

[[ 0.07 ]
 [-0.041]
 [-0.051]]

[[ 0.08 ]
 [-0.009]
 [-0.004]]

[[ 0.07 ]
 [-0.042]
 [-0.064]]

[[ 0.08 ]
 [-0.01 ]
 [-0.017]]

[[0.09 ]
 [0.022]
 [0.028]]

[[ 0.08 ]
 [-0.011]
 [-0.032]]

[[ 0.07 ]
 [-0.041]
 [-0.046]]

[[ 0.08 ]
 [-0.009]
 [ 0.001]]

[[ 0.07 ]
 [-0.042]
 [-0.059]]
```

```
[[ 0.08 ]
 [-0.01 ]
 [-0.012]]

[[0.09 ]
 [0.022]
 [0.033]]

[[ 0.08 ]
 [-0.011]
 [-0.027]]

[[ 0.07 ]
 [-0.046]
 [-0.041]]

[[ 0.08 ]
 [-0.014]
 [ 0.006]]

[[ 0.07 ]
 [-0.047]
 [-0.054]]

[[ 0.08 ]
 [-0.015]
 [-0.007]]

[[0.09 ]
 [0.016]
 [0.042]]

[[ 0.08 ]
 [-0.017]
 [-0.018]]

[[ 0.07 ]
 [-0.047]
 [-0.032]]

[[ 0.08 ]
 [-0.015]
 [ 0.015]]

[[ 0.07 ]
 [-0.048]
 [-0.045]]

[[ 0.08 ]
 [-0.016]
 [ 0.002]]

[[ 0.07 ]
 [-0.049]
 [-0.058]]

[[ 0.08 ]
 [-0.017]
 [-0.011]]

[[0.09 ]
 [0.015]
 [0.034]]

[[ 0.08 ]
 [-0.018]
 [-0.026]]

[[ 0.07 ]
 [-0.041]
 [-0.039]]

[[ 0.08 ]
 [-0.009]
 [ 0.008]]
```

```
[[ 0.07 ]
 [-0.042]
 [-0.052]]

[[ 0.08 ]
 [-0.01 ]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.065]]

[[ 0.08 ]
 [-0.011]
 [-0.018]]

[[0.09 ]
 [0.021]
 [0.027]]

[[ 0.08 ]
 [-0.012]
 [-0.033]]

[[ 0.07 ]
 [-0.042]
 [-0.044]]

[[ 0.08 ]
 [-0.01 ]
 [ 0.003]]

[[ 0.07 ]
 [-0.043]
 [-0.057]]

[[ 0.08 ]
 [-0.011]
 [-0.01 ]]

[[0.09 ]
 [0.021]
 [0.035]]

[[ 0.08 ]
 [-0.012]
 [-0.025]]

[[ 0.07 ]
 [-0.047]
 [-0.039]]

[[ 0.08 ]
 [-0.015]
 [ 0.008]]

[[ 0.07 ]
 [-0.048]
 [-0.052]]

[[ 0.08 ]
 [-0.016]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.056]]

[[ 0.08 ]
 [-0.011]
 [-0.009]]

[[ 0.07 ]
 [-0.038]
```

```
 [-0.06 ]]

[[ 0.08 ]
 [-0.006]
 [-0.013]]

[[0.09 ]
 [0.025]
 [0.036]]

[[ 0.08 ]
 [-0.008]
 [-0.024]]

[[ 0.07 ]
 [-0.043]
 [-0.038]]

[[ 0.08 ]
 [-0.011]
 [ 0.009]]

[[ 0.07 ]
 [-0.044]
 [-0.051]]

[[ 0.08 ]
 [-0.012]
 [-0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.064]]

[[ 0.08 ]
 [-0.013]
 [-0.017]]

[[0.09 ]
 [0.019]
 [0.028]]

[[ 0.08 ]
 [-0.014]
 [-0.032]]

[[ 0.07 ]
 [-0.044]
 [-0.043]]

[[ 0.08 ]
 [-0.012]
 [ 0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.056]]

[[ 0.08 ]
 [-0.013]
 [-0.009]]

[[0.09 ]
 [0.019]
 [0.036]]

[[ 0.08 ]
 [-0.014]
 [-0.024]]

[[ 0.07 ]
 [-0.044]
 [-0.038]]

[[ 0.08 ]
```

```
 [-0.012]
 [ 0.009]]

[[ 0.07 ]
 [-0.045]
 [-0.051]]

[[ 0.08 ]
 [-0.013]
 [-0.004]]

[[ 0.07 ]
 [-0.046]
 [-0.064]]

[[ 0.08 ]
 [-0.014]
 [-0.017]]

[[0.09 ]
 [0.018]
 [0.028]]

[[ 0.08 ]
 [-0.015]
 [-0.032]]

[[ 0.07 ]
 [-0.038]
 [-0.045]]

[[ 0.08 ]
 [-0.006]
 [ 0.002]]

[[ 0.07 ]
 [-0.039]
 [-0.058]]

[[ 0.08 ]
 [-0.007]
 [-0.011]]

[[ 0.07 ]
 [-0.034]
 [-0.062]]

[[ 0.08 ]
 [-0.002]
 [-0.015]]

[[0.09 ]
 [0.028]
 [0.035]]

[[ 0.08 ]
 [-0.005]
 [-0.025]]

[[ 0.07 ]
 [-0.04 ]
 [-0.039]]

[[ 0.08 ]
 [-0.008]
 [ 0.008]]

[[ 0.07 ]
 [-0.041]
 [-0.052]]

[[ 0.08 ]
 [-0.009]
 [-0.005]]
```

```
[[ 0.07 ]
 [-0.042]
 [-0.065]]

[[ 0.08 ]
 [-0.01 ]
 [-0.018]]

[[0.09 ]
 [0.022]
 [0.027]]

[[ 0.08 ]
 [-0.011]
 [-0.033]]

[[ 0.07 ]
 [-0.041]
 [-0.047]]

[[ 8.0000000e-02]
 [-9.0000000e-03]
 [ 8.7901908e-14]]

[[ 0.07 ]
 [-0.042]
 [-0.06 ]]

[[ 0.08 ]
 [-0.01 ]
 [-0.013]]

[[0.09 ]
 [0.022]
 [0.032]]

[[ 0.08 ]
 [-0.011]
 [-0.028]]

[[ 0.07 ]
 [-0.046]
 [-0.042]]

[[ 0.08 ]
 [-0.014]
 [ 0.005]]

[[ 0.07 ]
 [-0.047]
 [-0.055]]

[[ 0.08 ]
 [-0.015]
 [-0.008]]

[[0.09 ]
 [0.017]
 [0.037]]

[[ 0.08 ]
 [-0.016]
 [-0.023]]

[[ 0.07 ]
 [-0.045]
 [-0.037]]

[[ 0.08 ]
 [-0.013]
 [ 0.01 ]]

[[ 0.07 ]
 [-0.046]
 [-0.05 ]]
```

```
[[ 0.08 ]
 [-0.014]
 [-0.003]]

[[ 0.07 ]
 [-0.047]
 [-0.063]]

[[ 0.08 ]
 [-0.015]
 [-0.016]]

[[0.09 ]
 [0.017]
 [0.029]]

[[ 0.08 ]
 [-0.016]
 [-0.031]]

[[ 0.07 ]
 [-0.039]
 [-0.044]]

[[ 0.08 ]
 [-0.007]
 [ 0.003]]

[[ 0.07 ]
 [-0.04 ]
 [-0.057]]

[[ 0.08 ]
 [-0.008]
 [-0.01 ]]

[[ 0.07 ]
 [-0.035]
 [-0.061]]

[[ 0.08 ]
 [-0.003]
 [-0.014]]

[[ 0.07 ]
 [-0.03 ]
 [-0.065]]

[[ 0.08 ]
 [ 0.002]
 [-0.018]]

[[0.09 ]
 [0.033]
 [0.031]]

[[ 8.00000000e-02]
 [-3.30985239e-14]
 [-2.90000000e-02]]

[[ 0.07 ]
 [-0.035]
 [-0.043]]

[[ 0.08 ]
 [-0.003]
 [ 0.004]]

[[ 0.07 ]
 [-0.036]
 [-0.056]]

[[ 0.08 ]
 [-0.004]
```

```
 [-0.009]]

[[ 0.07 ]
 [-0.037]
 [-0.069]]

[[ 0.08 ]
 [-0.005]
 [-0.022]]

[[0.09 ]
 [0.027]
 [0.023]]

[[ 0.08 ]
 [-0.006]
 [-0.037]]

[[ 0.07 ]
 [-0.041]
 [-0.051]]

[[ 0.08 ]
 [-0.009]
 [-0.004]]

[[ 0.07 ]
 [-0.042]
 [-0.064]]

[[ 0.08 ]
 [-0.01 ]
 [-0.017]]

[[0.09 ]
 [0.022]
 [0.028]]

[[ 0.08 ]
 [-0.011]
 [-0.032]]

[[ 0.07 ]
 [-0.041]
 [-0.046]]

[[ 0.08 ]
 [-0.009]
 [ 0.001]]

[[ 0.07 ]
 [-0.042]
 [-0.059]]

[[ 0.08 ]
 [-0.01 ]
 [-0.012]]

[[0.09 ]
 [0.022]
 [0.033]]

[[ 0.08 ]
 [-0.011]
 [-0.027]]

[[ 0.07 ]
 [-0.046]
 [-0.041]]

[[ 0.08 ]
 [-0.014]
 [ 0.006]]

[[ 0.07 ]
```

```
 [-0.047]
 [-0.054]]

[[ 0.08 ]
 [-0.015]
 [-0.007]]

[[0.09 ]
 [0.016]
 [0.042]]

[[ 0.08 ]
 [-0.017]
 [-0.018]]

[[ 0.07 ]
 [-0.047]
 [-0.032]]

[[ 0.08 ]
 [-0.015]
 [ 0.015]]

[[ 0.07 ]
 [-0.048]
 [-0.045]]

[[ 0.08 ]
 [-0.016]
 [ 0.002]]

[[ 0.07 ]
 [-0.049]
 [-0.058]]

[[ 0.08 ]
 [-0.017]
 [-0.011]]

[[0.09 ]
 [0.015]
 [0.034]]

[[ 0.08 ]
 [-0.018]
 [-0.026]]

[[ 0.07 ]
 [-0.041]
 [-0.039]]

[[ 0.08 ]
 [-0.009]
 [ 0.008]]

[[ 0.07 ]
 [-0.042]
 [-0.052]]

[[ 0.08 ]
 [-0.01 ]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.065]]

[[ 0.08 ]
 [-0.011]
 [-0.018]]

[[0.09 ]
 [0.021]
 [0.027]]
```

```
[[ 0.08 ]
 [-0.012]
 [-0.033]]

[[ 0.07 ]
 [-0.042]
 [-0.044]]

[[ 0.08 ]
 [-0.01 ]
 [ 0.003]]

[[ 0.07 ]
 [-0.043]
 [-0.057]]

[[ 0.08 ]
 [-0.011]
 [-0.01 ]]

[[0.09 ]
 [0.021]
 [0.035]]

[[ 0.08 ]
 [-0.012]
 [-0.025]]

[[ 0.07 ]
 [-0.047]
 [-0.039]]

[[ 0.08 ]
 [-0.015]
 [ 0.008]]

[[ 0.07 ]
 [-0.048]
 [-0.052]]

[[ 0.08 ]
 [-0.016]
 [-0.005]]

[[ 0.07 ]
 [-0.043]
 [-0.056]]

[[ 0.08 ]
 [-0.011]
 [-0.009]]

[[ 0.07 ]
 [-0.038]
 [-0.06 ]]

[[ 0.08 ]
 [-0.006]
 [-0.013]]

[[0.09 ]
 [0.025]
 [0.036]]

[[ 0.08 ]
 [-0.008]
 [-0.024]]

[[ 0.07 ]
 [-0.043]
 [-0.038]]

[[ 0.08 ]
 [-0.011]
 [ 0.009]]
```

```
[[ 0.07 ]
 [-0.044]
 [-0.051]]

[[ 0.08 ]
 [-0.012]
 [-0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.064]]

[[ 0.08 ]
 [-0.013]
 [-0.017]]

[[0.09 ]
 [0.019]
 [0.028]]

[[ 0.08 ]
 [-0.014]
 [-0.032]]

[[ 0.07 ]
 [-0.044]
 [-0.043]]

[[ 0.08 ]
 [-0.012]
 [ 0.004]]

[[ 0.07 ]
 [-0.045]
 [-0.056]]

[[ 0.08 ]
 [-0.013]
 [-0.009]]

[[0.09 ]
 [0.019]
 [0.036]]

[[ 0.08 ]
 [-0.014]
 [-0.024]]

[[ 0.07 ]
 [-0.044]
 [-0.038]]

[[ 0.08 ]
 [-0.012]
 [ 0.009]]

[[ 0.07 ]
 [-0.045]
 [-0.051]]

[[ 0.08 ]
 [-0.013]
 [-0.004]]

[[ 0.07 ]
 [-0.046]
 [-0.064]]

[[ 0.08 ]
 [-0.014]
 [-0.017]]

[[0.09 ]
 [0.018]
```

```
 [0.028]]

[[ 0.08 ]
 [-0.015]
 [-0.032]]

[[ 0.07 ]
 [-0.038]
 [-0.045]]

[[ 0.08 ]
 [-0.006]
 [ 0.002]]

[[ 0.07 ]
 [-0.039]
 [-0.058]]

[[ 0.08 ]
 [-0.007]
 [-0.011]]

[[ 0.07 ]
 [-0.034]
 [-0.062]]

[[ 0.08 ]
 [-0.002]
 [-0.015]]

[[0.09 ]
 [0.028]
 [0.035]]

[[ 0.08 ]
 [-0.005]
 [-0.025]]

[[ 0.07 ]
 [-0.04 ]
 [-0.039]]

[[ 0.08 ]
 [-0.008]
 [ 0.008]]

[[ 0.07 ]
 [-0.041]
 [-0.052]]

[[ 0.08 ]
 [-0.009]
 [-0.005]]

[[ 0.07 ]
 [-0.042]
 [-0.065]]

[[ 0.08 ]
 [-0.01 ]
 [-0.018]]

[[0.09 ]
 [0.022]
 [0.027]]

[[ 0.08 ]
 [-0.011]
 [-0.033]]

[[ 0.07 ]
 [-0.041]
 [-0.047]]

[[ 8.00000000e-02]
```

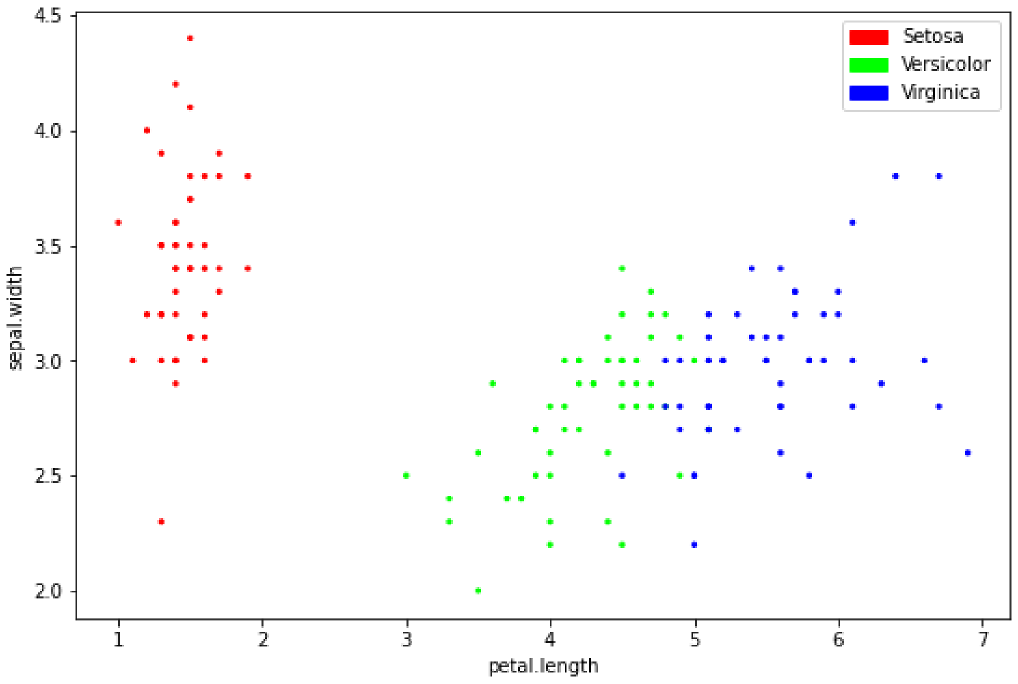
```
 [-9.00000000e-03]
 [ 8.80129303e-14]]

[[ 0.07 ]
 [-0.042]
 [-0.06 ]]

[[ 0.08 ]
 [-0.01 ]
 [-0.013]]

[[0.09 ]
 [0.022]
 [0.032]]

[[ 0.08 ]
 [-0.011]
 [-0.028]]

[[ 0.07 ]
 [-0.046]
 [-0.042]]

[[ 0.08 ]
 [-0.014]
 [ 0.005]]

[[ 0.07 ]
 [-0.047]
 [-0.055]]

[[ 0.08 ]
 [-0.015]
 [-0.008]]

[[0.09 ]
 [0.017]
 [0.037]]

[[ 0.08 ]
 [-0.016]
 [-0.023]]

[[ 0.07 ]
 [-0.045]
 [-0.037]]

[[ 0.08 ]
 [-0.013]
 [ 0.01 ]]

[[ 0.07 ]
 [-0.046]
 [-0.05 ]]

[[ 0.08 ]
 [-0.014]
 [-0.003]]

[[ 0.07 ]
 [-0.047]
 [-0.063]]

[[ 0.08 ]
 [-0.015]
 [-0.016]]

[[0.09 ]
 [0.017]
 [0.029]]

[[ 0.08 ]
 [-0.016]
 [-0.031]]
```

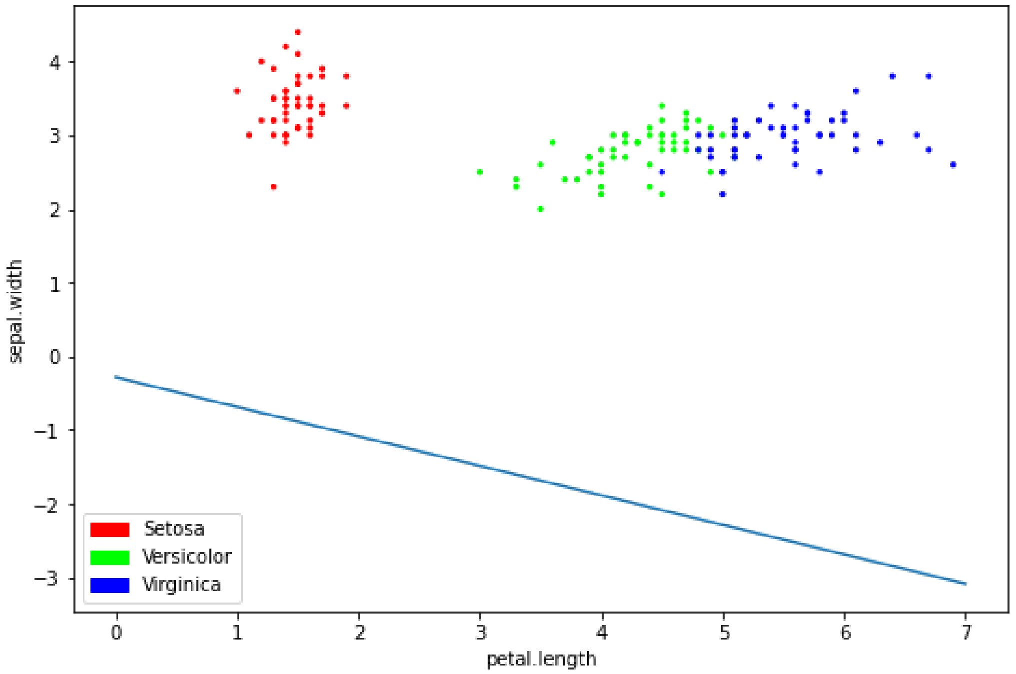
```
[[ 0.07 ]
 [-0.039]
 [-0.044]]

[[ 0.08 ]
 [-0.007]
 [ 0.003]]

[[ 0.07 ]
 [-0.04 ]
 [-0.057]]

[[ 0.08 ]
 [-0.008]
 [-0.01 ]]

[[ 0.07 ]
 [-0.035]
 [-0.061]]

[[ 0.08 ]
 [-0.003]
 [-0.014]]

[[ 0.07 ]
 [-0.03 ]
 [-0.065]]

[[ 0.08 ]
 [ 0.002]
 [-0.018]]

[[0.09 ]
 [0.033]
 [0.031]]

[[ 8.00000000e-02]
 [-3.31401573e-14]
 [-2.90000000e-02]]

[[ 0.07 ]
 [-0.035]
 [-0.043]]

[[ 0.08 ]
 [-0.003]
 [ 0.004]]
```

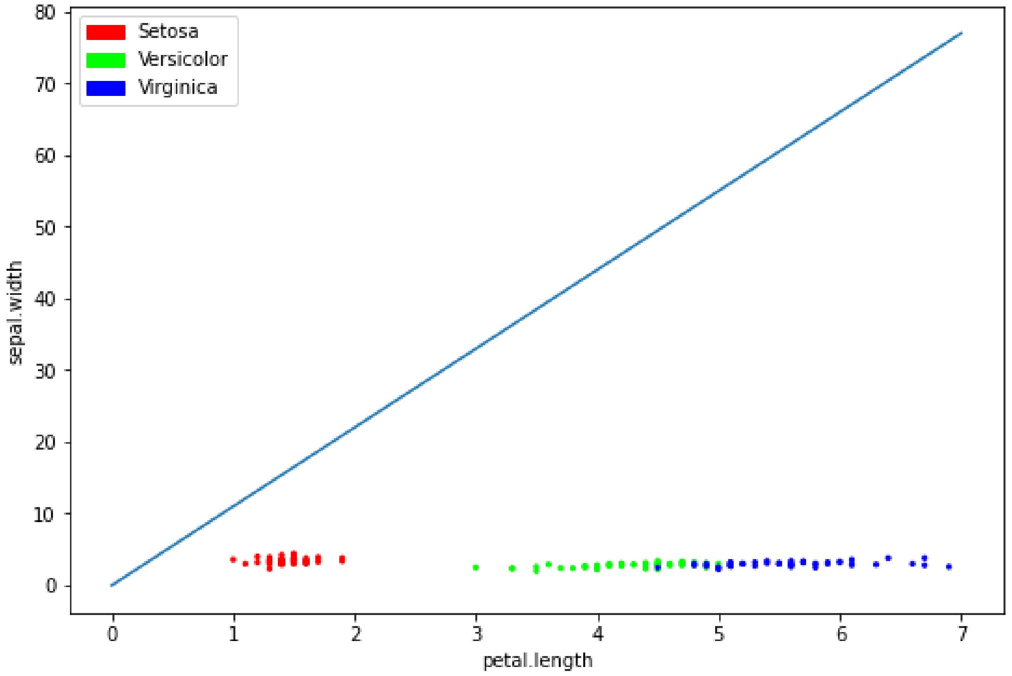
In [ ]:

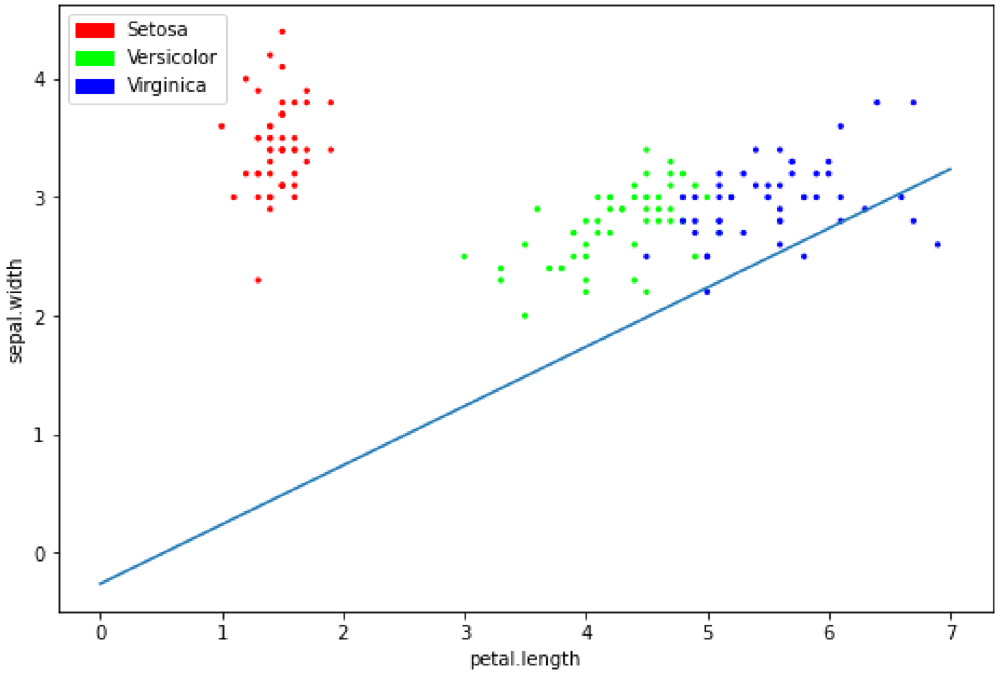
```
for w in weight_vector1:
  cm  = ListedColormap(['#FF0000', '#00FF00', '#0000FF'])
  patch0 = mpatches.Patch(color = '#FF0000', label = 'Setosa')
  patch1 = mpatches.Patch(color = '#00FF00', label = 'Versicolor')
  patch2 = mpatches.Patch(color = '#0000FF', label = 'Virginica')

  x1 = np.arange(8)

  plt.figure(figsize = (9, 6))
  plt.scatter(df['petal.length'], df['sepal.width'], c = [0] * 50 + [1] * 50 + [2] * 50, cmap = cm, s = 5)
  plt.xlabel('petal.length')
  plt.ylabel('sepal.width')
  plt.legend(handles = [patch0, patch1, patch2])

  slope, c = -w[2] / w[1], -w[0] / w[1]
  plt.plot(x1, slope * x1 + c)
  plt.show()
```

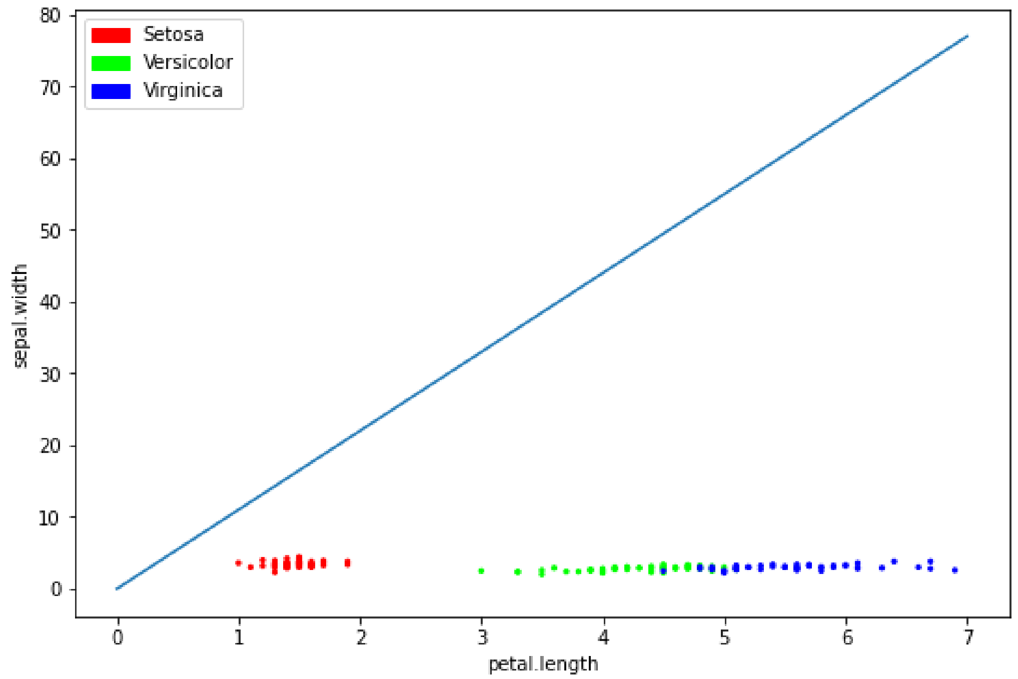
```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:15: RuntimeWarning: invalid value encountered in true_divide
  from ipykernel import kernelapp as app
```

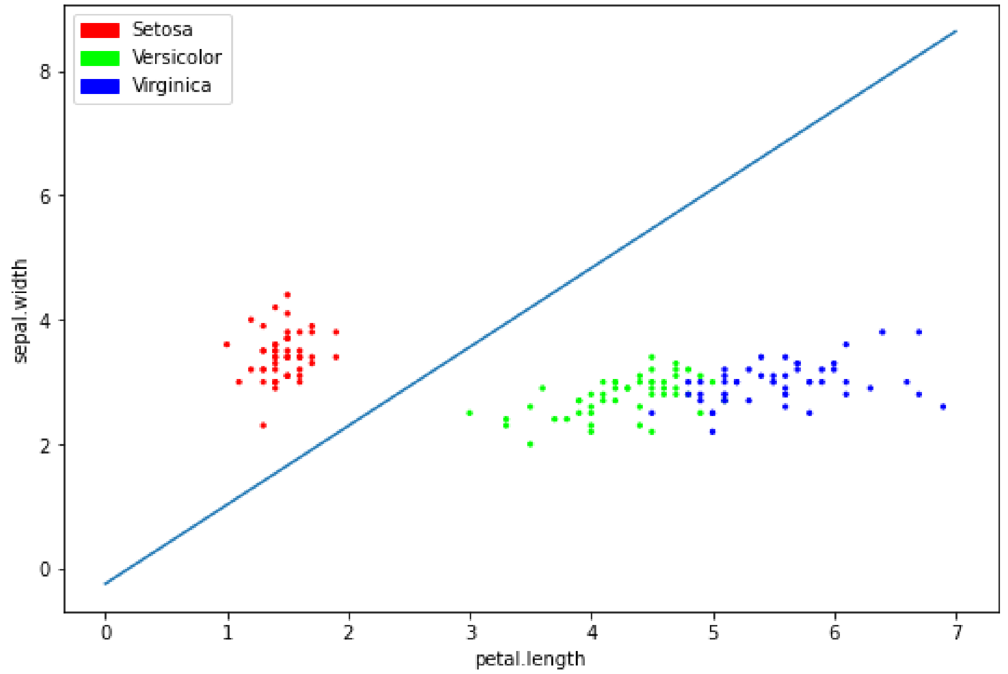

Setosa
Versicolor
Virginica
sepal.width
petal.length
4.5
4.0
3.5
3.0
2.5
2.0
1
2
3
4
5
6
7


Setosa
Versicolor
Virginica
sepal.width
petal.length
4
3
2
1
0
−1
−2
−3
0
1
2
3
4
5
6
7


Setosa
Versicolor
Virginica
sepal.width
petal.length
0
10
20
30
40
50
60
70
80
0
1
2
3
4
5
6
7

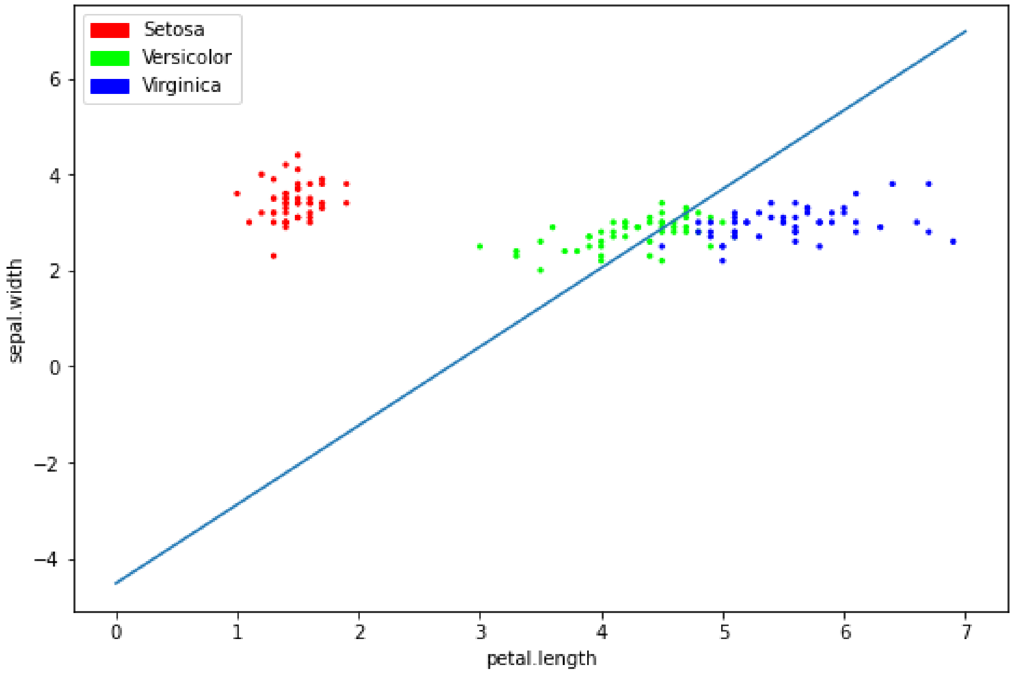

Setosa
Versicolor
Virginica
sepal.width
petal.length
0
1
2
3
4
0
1
2
3
4
5
6
7





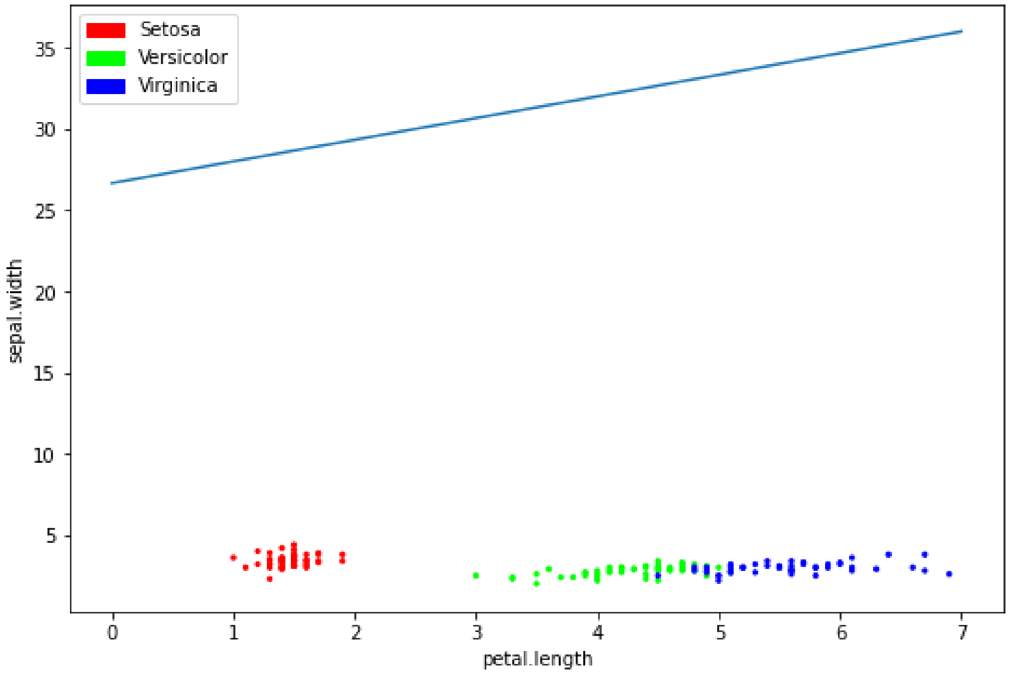
In [ ]:

```
w = W2
cm  = ListedColormap(['#FF0000', '#00FF00', '#0000FF'])
patch0 = mpatches.Patch(color = '#FF0000', label = 'Setosa')
patch1 = mpatches.Patch(color = '#00FF00', label = 'Versicolor')
patch2 = mpatches.Patch(color = '#0000FF', label = 'Virginica')

x1 = np.arange(8)

plt.figure(figsize = (9, 6))
plt.scatter(df['petal.length'], df['sepal.width'], c = [0] * 50 + [1] * 50 + [2] * 50, cmap = cm, s = 5)
plt.xlabel('petal.length')
plt.ylabel('sepal.width')
plt.legend(handles = [patch0, patch1, patch2])

slope, c = -w[2] / w[1], -w[0] / w[1]
plt.plot(x1, slope * x1 + c)
plt.show()
```



In [ ]:

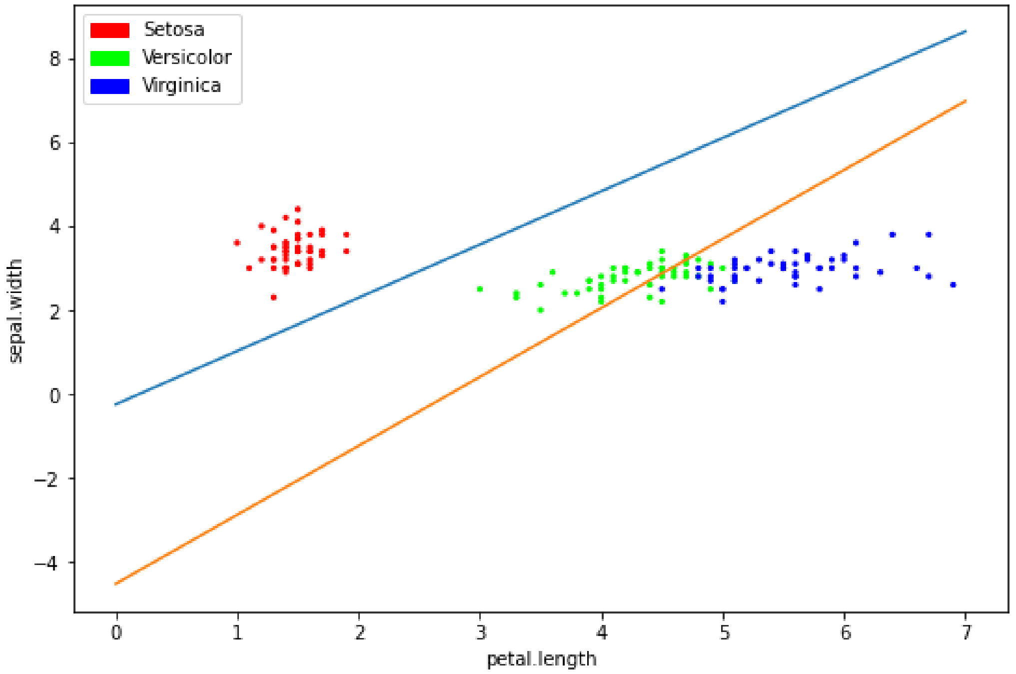
```
w = W3
cm  = ListedColormap(['#FF0000', '#00FF00', '#0000FF'])
patch0 = mpatches.Patch(color = '#FF0000', label = 'Setosa')
patch1 = mpatches.Patch(color = '#00FF00', label = 'Versicolor')
patch2 = mpatches.Patch(color = '#0000FF', label = 'Virginica')

x1 = np.arange(8)

plt.figure(figsize = (9, 6))
plt.scatter(df['petal.length'], df['sepal.width'], c = [0] * 50 + [1] * 50 + [2] * 50, cmap = cm, s = 5)
plt.xlabel('petal.length')
plt.ylabel('sepal.width')
plt.legend(handles = [patch0, patch1, patch2])

slope, c = -w[2] / w[1], -w[0] / w[1]
plt.plot(x1, slope * x1 + c)
plt.show()
```



In [ ]:

```
cm  = ListedColormap(['#FF0000', '#00FF00', '#0000FF'])
patch0 = mpatches.Patch(color = '#FF0000', label = 'Setosa')
```

```
patch1 = mpatches.Patch(color = '#00FF00', label = 'Versicolor')
patch2 = mpatches.Patch(color = '#0000FF', label = 'Virginica')

x1 = np.arange(8)

plt.figure(figsize = (9, 6))
plt.scatter(df['petal.length'], df['sepal.width'], c = [0] * 50 + [1] * 50 + [2] * 50, cmap = cm, s = 5)
plt.xlabel('petal.length')
plt.ylabel('sepal.width')
plt.legend(handles = [patch0, patch1, patch2])

w = W1
slope, c = -w[2] / w[1], -w[0] / w[1]
plt.plot(x1, slope * x1 + c)

w = W2
slope, c = -w[2] / w[1], -w[0] / w[1]
plt.plot(x1, slope * x1 + c)
plt.show()
```



SVM

Defining the necessary functions used

In [ ]:

```
def SVM(C1, C2):
  x1 = np.array(C1.to_numpy())
  x2 = np.array(C2.to_numpy())

  x = np.vstack((x1, x2))
  y = np.hstack((np.ones(len(x1)), np.ones(len(x2)) * -1))

  X = np.dot(x, np.transpose(x))
  Y = np.outer(y, y)

  n = X.shape[0]

  H = matrix(Y * X)
  f = matrix(np.ones(n) * -1)
  A = matrix(np.diag(np.ones(n) * -1))
  a = matrix(np.zeros(n))
  B = matrix(y, (1, n))
  b = matrix(0.0)
```

```
  # solve QP problem
  solution = solvers.qp(H, f, A, a, B, b)
  print('')

  # Lagrange multipliers
  t1 = np.ravel(solution['x'])

  # Support vectors have non zero lagrange multipliers
  for i in range(0, len(t1)):
    if(t1[i] < 1e-4):
      t1[i] = 0.0
    else:
      t1[i] = round(t1[i], 4)

  t1 = np.array(t1)
  print(t1)

  # Weight vector
  w = np.zeros(X.shape[1])
  for i in range(X.shape[1]):
    w[i] = t1[i] * y[i]

  w = np.dot(w, x)

  # Intercept
  b = 0
  for i in range(len(t1)):
    if(t1[i] != 0.0):
      b = round(1 - np.dot(w, np.transpose(x[i])), 4)
      break

  return w, b
```

In [ ]:

```
def decision_boundary(labels, C1, W1, b1, C2, W2, b2, C3, W3, b3):
  x = np.arange(8)

  plt.figure(figsize = (9, 6))
  plt.scatter(C1['petal.length'], C1['sepal.width'], label = 'Setosa', color = '#FF0000')
  plt.scatter(C2['petal.length'], C2['sepal.width'], label = 'Versicolor', color = '#00FF00')
  plt.scatter(C3['petal.length'], C3['sepal.width'], label = 'Virginica', color = '#0000FF')

  plt.suptitle("Iris Dataset Classification", fontweight = "bold")
  plt.ylabel("Petal length (in cm)", fontweight = "bold")
  plt.xlabel("Sepal width (in cm)", fontweight = "bold")

  w = W1
  b = b1
  print(w, "\n")
  slope, c = -w[0] / w[1], -b / w[1]
  plt.plot(x, slope * x + c, label = "Decision Boundary - Setosa and Versicolor")

  w = W3
  b = b3
  print(w, "\n")
  slope, c = -w[0] / w[1], -b / w[1]
  plt.plot(x, slope * x + c, label = "Decision Boundary - Setosa and Virginica")
  plt.legend()
  plt.show()

  plt.figure(figsize = (9, 6))
  plt.scatter(C1['petal.length'], C1['sepal.width'], label = 'Setosa', color = '#FF0000')
  plt.scatter(C2['petal.length'], C2['sepal.width'], label = 'Versicolor', color = '#00FF00')
  plt.scatter(C3['petal.length'], C3['sepal.width'], label = 'Virginica', color = '#0000FF')

  plt.suptitle("Iris Dataset Classification", fontweight = "bold")
  plt.ylabel("Petal length (in cm)", fontweight = "bold")
  plt.xlabel("Sepal width (in cm)", fontweight = "bold")
```

```
    w = W1
    b = b1
    print(w, "\n")
    slope, c = -w[0] / w[1], -b / w[1]
    plt.plot(x, slope * x + c, label = "Decision Boundary - Setosa and Versicolor")

    w = W3
    b = b3
    print(w, "\n")
    slope, c = -w[0] / w[1], -b / w[1]
    plt.plot(x, slope * x + c, label = "Decision Boundary - Setosa and Virginica")

    w = W2
    b = b2
    print(w, "\n")
    slope, c = -w[0] / w[1], -b / w[1]
    plt.plot(x, slope * x + c, label = "Decision Boundary - Versicolor and Virginica")
    plt.legend()
    plt.show()
```

Code

In [ ]:

```
C1 = df.loc[df.variety == 'Setosa'].drop(['variety'], axis = 1)
print(C1.head(5))
print(C1.tail(5))
C1.shape
```

```
   sepal.width  petal.length
0          3.5           1.4
1          3.0           1.4
2          3.2           1.3
3          3.1           1.5
4          3.6           1.4
    sepal.width  petal.length
45          3.0           1.4
46          3.8           1.6
47          3.2           1.4
48          3.7           1.5
49          3.3           1.4
```

Out[ ]: (50, 2)

In [ ]:

```
C2 = df.loc[df.variety == 'Versicolor'].drop(['variety'], axis = 1)
print(C2.head(5))
print(C2.tail(5))
C2.shape
```

```
    sepal.width  petal.length
50          3.2           4.7
51          3.2           4.5
52          3.1           4.9
53          2.3           4.0
54          2.8           4.6
    sepal.width  petal.length
95          3.0           4.2
96          2.9           4.2
97          2.9           4.3
98          2.5           3.0
99          2.8           4.1
```

Out[ ]: (50, 2)

In [ ]:

```
C3 = df.loc[df.variety == 'Virginica'].drop(['variety'], axis = 1)
print(C3.head(5))
print(C3.tail(5))
C3.shape
```

```
     sepal.width  petal.length
```

```
100          3.3           6.0
101          2.7           5.1
102          3.0           5.9
103          2.9           5.6
104          3.0           5.8
     sepal.width  petal.length
145          3.0           5.2
146          2.5           5.0
147          3.0           5.2
148          3.4           5.4
149          3.0           5.1
```

Out[ ]:
```
(50, 2)
```

In [ ]:
```python
w1, b1 = SVM(C1, C2)

print(w1)
print('')
print(b1)
```

```
     pcost       dcost       gap    pres   dres
 0: -4.3867e+00 -8.1716e+00  3e+02  1e+01  2e+00
 1: -2.3798e+00 -2.5258e+00  2e+01  1e+00  2e-01
 2: -4.3479e-01 -1.7538e+00  2e+00  5e-02  6e-03
 3: -6.7595e-01 -1.0544e+00  5e-01  1e-02  1e-03
 4: -8.1435e-01 -1.1260e+00  4e-01  4e-03  5e-04
 5: -1.0085e+00 -1.0271e+00  2e-02  8e-05  1e-05
 6: -1.0251e+00 -1.0253e+00  2e-04  9e-07  1e-07
 7: -1.0253e+00 -1.0253e+00  2e-06  9e-09  1e-09
 8: -1.0253e+00 -1.0253e+00  2e-08  9e-11  1e-11
Optimal solution found.

[0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.8098 0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.2155 0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     1.0253 0.    ]
[ 0.68572 -1.25713]

1.0571
```

In [ ]:
```python
w2, b2 = SVM(C2, C3)

print(w2)
print('')
print(b2)
```

```
     pcost       dcost       gap    pres   dres
 0: -3.7394e+01 -8.7261e+01  5e+02  2e+01  2e+00
 1: -1.0739e+02 -1.7754e+02  3e+02  1e+01  2e+00
 2: -5.7593e+02 -8.7585e+02  4e+02  1e+01  2e+00
 3: -1.9989e+03 -2.2838e+03  3e+02  1e+01  1e+00
 4: -6.2946e+03 -6.9640e+03  7e+02  1e+01  1e+00
 5: -3.5722e+04 -3.7828e+04  2e+03  1e+01  1e+00
 6: -1.6095e+05 -1.6859e+05  8e+03  1e+01  1e+00
 7: -2.7869e+06 -2.8212e+06  3e+04  9e+00  1e+00
 8: -1.2140e+08 -1.2212e+08  7e+05  9e+00  1e+00
 9: -1.9228e+10 -1.9236e+10  8e+06  9e+00  1e+00
10: -1.9394e+10 -1.9402e+10  8e+06  9e+00  1e+00
11: -2.1446e+10 -2.1454e+10  8e+06  9e+00  1e+00
12: -3.4380e+10 -3.4393e+10  1e+07  9e+00  1e+00
13: -5.4784e+10 -5.4805e+10  2e+07  9e+00  1e+00
14: -8.7577e+10 -8.7607e+10  3e+07  9e+00  1e+00
15: -1.0218e+11 -1.0222e+11  3e+07  9e+00  1e+00
Terminated (singular KKT matrix).

[2.71521160e+07 4.51595280e+06 1.29133612e+08 1.00265211e+08
```

```
 3.09216898e+08 1.42980907e+08 4.25115472e+06 1.24416108e+07
 1.96826025e+08 2.17717942e+07 1.22555173e+08 8.75290549e+06
 7.67649435e+07 1.58491041e+08 2.83907018e+05 2.47893925e+05
 3.68052772e+07 4.61959347e+07 1.00816888e+09 8.83969315e+07
 1.09546196e+07 2.12476560e+07 2.32690275e+09 5.51639458e+09
 1.57936433e+06 1.95865133e+07 1.61922907e+08 2.70030424e+08
 8.40604842e+07 1.41692613e+07 5.99425006e+07 2.47557397e+07
 2.40241351e+07 3.66390422e+10 2.30503621e+07 7.90295844e+06
 4.95320836e+07 4.32821956e+08 1.63526678e+07 2.19136307e+08
 3.34342989e+08 4.28761312e+07 2.65803368e+08 2.20079195e+07
 5.62148082e+07 1.59587525e+08 5.83123718e+07 3.85951082e+06
 1.63510780e+09 1.03242757e+08 8.28841307e+07 2.61588909e+08
 2.35253914e+07 6.27638197e+07 3.70348945e+07 8.94810821e+07
 2.68823443e+10 1.80964094e+07 3.07022922e+08 3.14754482e+07
 3.52233325e+08 2.51425758e+09 5.19114959e+07 2.13739902e+07
 1.19027991e+08 1.44055382e+08 5.18845100e+07 8.34998928e+07
 1.26029271e+06 4.46210403e+07 4.89155288e+08 4.28792523e+08
 4.12928047e+04 2.17695331e+08 1.59905611e+08 1.80865194e+07
 4.76400125e+09 1.83710018e+09 6.17354594e+07 3.24900781e+07
 5.78027985e+09 1.06941424e+08 6.17370471e+07 1.19094824e+08
 1.37617004e+08 8.42083709e+07 8.68930538e+07 5.50808776e+07
 1.26831614e+09 2.95476071e+08 1.01801411e+09 8.46133030e+08
 2.20642492e+08 6.11327938e+07 1.59593590e+08 6.12728556e+08
 8.96855032e+06 6.14535479e+08 1.36892256e+08 2.36350952e+08]
[0.00036621 0.00012207]

0.9983
```

In [ ]:

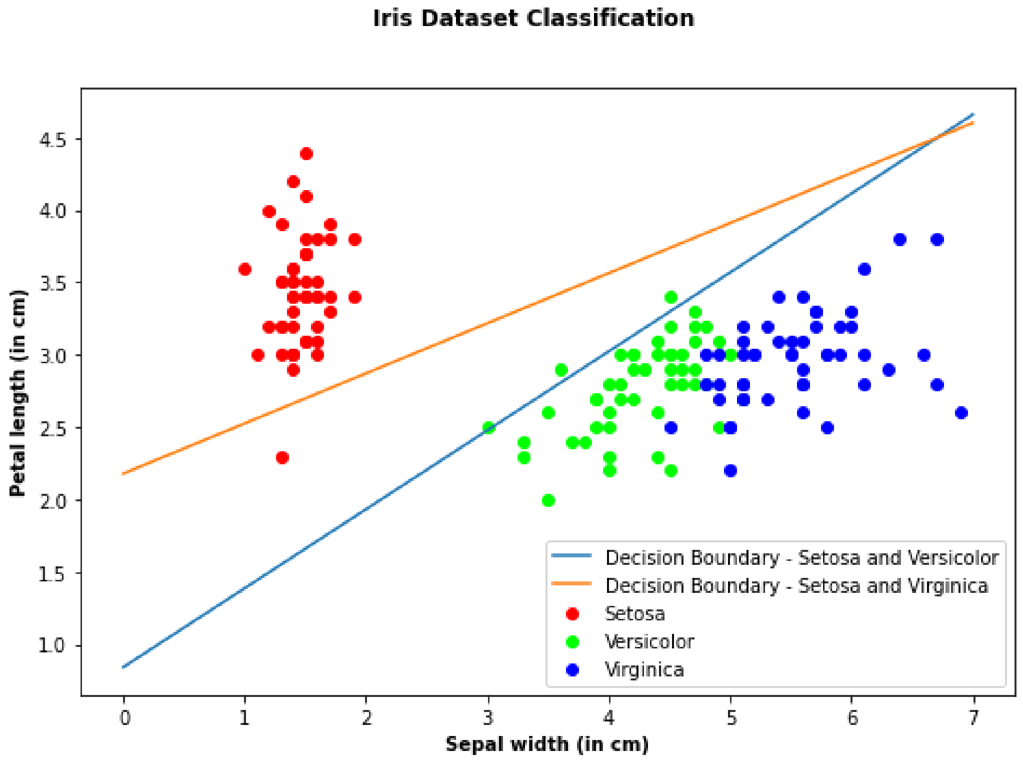
```python
w3, b3 = SVM(C1, C3)

print(w3)
print('')
print(b3)
```

```
     pcost       dcost       gap    pres   dres
 0: -3.3137e+00 -4.9765e+00  2e+02  2e+01  1e+00
 1: -1.6126e-01 -8.1531e-01  1e+01  9e-01  8e-02
 2:  1.9106e-02 -6.5205e-01  9e-01  1e-02  1e-03
 3: -1.5487e-01 -2.7974e-01  1e-01  7e-04  6e-05
 4: -2.2572e-01 -2.8972e-01  6e-02  1e-04  1e-05
 5: -2.5983e-01 -2.6459e-01  5e-03  1e-05  8e-07
 6: -2.6415e-01 -2.6421e-01  5e-05  1e-07  9e-09
 7: -2.6420e-01 -2.6420e-01  5e-07  1e-09  9e-11
 8: -2.6420e-01 -2.6420e-01  5e-09  1e-11  9e-13
Optimal solution found.

[0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.2642 0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.2642 0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.
 0.     0.     0.     0.     0.     0.     0.     0.     0.     0.    ]
[ 0.23778 -0.68692]

1.4967
```

In [ ]:

```python
decision_boundary(["Setosa", "Versicolor", "Virginica"], C1, w1, b1, C2, w2, b2, C3, w3, b3)
```

```
[ 0.68572 -1.25713]

[ 0.23778 -0.68692]
```




```
[ 0.68572 -1.25713]

[ 0.23778 -0.68692]

[0.00036621 0.00012207]
```