Maxwell Jung
9319674

Homework 1

CMPSC 165B
October 16, 2022

1. (a) $(3B)^T = (3 \times \begin{bmatrix} 0 & 4 \\ 7 & 6 \\ 5 & 8 \\ 3 & 11 \end{bmatrix})^T = \begin{bmatrix} 0 & 12 \\ 21 & 18 \\ 15 & 24 \\ 9 & 33 \end{bmatrix}^T = \begin{bmatrix} 0 & 21 & 15 & 9 \\ 12 & 18 & 24 & 33 \end{bmatrix}$

   (b) $(A - B)^T$ is impossible because A and B are not the same shape.

   (c) $(2B^T - A)^T = (2\times \begin{bmatrix} 0 & 4 \\ 7 & 6 \\ 5 & 8 \\ 3 & 11 \end{bmatrix}^T - \begin{bmatrix} 4 & 1 & 3 & 6 \\ 2 & 7 & 5 & 3 \end{bmatrix})^T = (2\times \begin{bmatrix} 0 & 7 & 5 & 3 \\ 4 & 6 & 8 & 11 \end{bmatrix} - \begin{bmatrix} 4 & 1 & 3 & 6 \\ 2 & 7 & 5 & 3 \end{bmatrix})^T =$
   $\begin{bmatrix} -4 & 13 & 7 & 0 \\ 6 & 5 & 11 & 19 \end{bmatrix}^T = \begin{bmatrix} -4 & 6 \\ 13 & 5 \\ 7 & 11 \\ 0 & 19 \end{bmatrix}$

   (d) $(C + 2D^T + E)^T$ is impossible because E is different shape from C or D.

   (e) $(-A)^T E = \begin{bmatrix} -4 & -2 \\ -1 & -7 \\ -3 & -5 \\ -6 & -3 \end{bmatrix} \times \begin{bmatrix} -4 & 5 \\ 12 & 7 \end{bmatrix} = \begin{bmatrix} 4 & -34 \\ -80 & -54 \\ -48 & -50 \\ -12 & -51 \end{bmatrix}$

2. $AB = \begin{bmatrix} 2 & 7 & 3 \\ 1 & 0 & 9 \\ -1 & 2 & 10 \end{bmatrix} \times \begin{bmatrix} -2 & 0 & 3 \\ 2 & -1 & 7 \\ 6 & 4 & -3 \end{bmatrix} = \begin{bmatrix} 28 & . & . \\ . & . & . \\ . & . & . \end{bmatrix}$

   $BA = \begin{bmatrix} -2 & 0 & 3 \\ 2 & -1 & 7 \\ 6 & 4 & -3 \end{bmatrix} \times \begin{bmatrix} 2 & 7 & 3 \\ 1 & 0 & 9 \\ -1 & 2 & 10 \end{bmatrix} = \begin{bmatrix} -7 & . & . \\ . & . & . \\ . & . & . \end{bmatrix}$

   The first element is different; therefore, $AB \neq BA$.

3. (a) $\left\| \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \right\|_1 = |0|+|0|+|0| = 0$ $\left\| \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \right\|_2 = \sqrt{0^2 + 0^2 + 0^2} = 0$ $\left\| \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \right\|_\infty = max(|0|, |0|, |0|) =$
   0

   (b) $\left\| \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \right\|_1 = |1| + |2| + |3| = 6$ $\left\| \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \right\|_2 = \sqrt{1^2 + 2^2 + 3^2} = 3.742$

   $\left\| \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \right\|_\infty = max(|1|, |2|, |3|) = 3$

   (c) $\left\| \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix} \right\|_1 = |2| + |4| + |6| = 12$ $\left\| \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix} \right\|_2 = \sqrt{2^2 + 4^2 + 6^2} = 7.483$

$$\left\| \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix} \right\|_\infty = max(|2|, |4|, |6|) = 6$$

(d) Norm is by definition non-negative.
l_1 is non-negative because it's the sum of absolute values.
l_2 is non-negative because it's square root of the sum of non negative numbers.
l_∞ is non-negative because it's the maximum value of the absolute values.

4. $X = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix} = \begin{bmatrix} x_{11} & x_{21} & \dots & x_{n1} \\ x_{12} & x_{22} & \dots & x_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1m} & x_{2m} & \dots & x_{nm} \end{bmatrix}$

$Y = \begin{bmatrix} y_1^T \\ y_2^T \\ \vdots \\ y_n^T \end{bmatrix} = \begin{bmatrix} y_{11} & y_{12} & \dots & y_{1p} \\ y_{21} & y_{22} & \dots & y_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ y_{n1} & y_{n2} & \dots & y_{np} \end{bmatrix}$

$$\sum_{i=1}^{n} x_i(y_i)^T = \sum_{i=1}^{n} \begin{bmatrix} x_{i1}y_{i1} & x_{i1}y_{i2} & \dots & x_{i1}y_{ip} \\ x_{i2}y_{i1} & x_{i2}y_{i2} & \dots & x_{i2}y_{ip} \\ \vdots & \vdots & \ddots & \vdots \\ x_{im}y_{i1} & x_{im}y_{i2} & \dots & x_{im}y_{ip} \end{bmatrix}$$

$$= \begin{bmatrix} x_{11}y_{11} + ... + x_{n1}y_{n1} & x_{11}y_{12} + ... + x_{n1}y_{n2} & \dots & x_{11}y_{1p} + ... + x_{n1}y_{np} \\ x_{12}y_{11} + ... + x_{n2}y_{n1} & x_{12}y_{12} + ... + x_{n2}y_{n2} & \dots & x_{12}y_{1p} + ... + x_{n2}y_{np} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1m}y_{11} + ... + x_{nm}y_{n1} & x_{1m}y_{12} + ... + x_{nm}y_{n2} & \dots & x_{1m}y_{1p} + ... + x_{nm}y_{np} \end{bmatrix}$$

$$= XY$$

5. $X = \begin{bmatrix} x_{11} & x_{21} & \dots & x_{n1} \\ x_{12} & x_{22} & \dots & x_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1m} & x_{2m} & \dots & x_{nm} \end{bmatrix}$

$X^TX = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{bmatrix} \begin{bmatrix} x_{11} & x_{21} & \dots & x_{n1} \\ x_{12} & x_{22} & \dots & x_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1m} & x_{2m} & \dots & x_{nm} \end{bmatrix}$

$$= \begin{bmatrix} x_1 \cdot x_1 & x_1 \cdot x_2 & \dots & x_1 \cdot x_n \\ x_2 \cdot x_1 & x_2 \cdot x_2 & \dots & x_2 \cdot x_n \\ \vdots & \vdots & \ddots & \vdots \\ x_n \cdot x_1 & x_n \cdot x_2 & \dots & x_n \cdot x_n \end{bmatrix}$$

$u \cdot v = v \cdot u$, so $X^T X$ is symmetric.

$X^T X$ is positive semi-definite because for any vector v, $v^T(X^T)Xv = (Xv)^T(Xv) = \|Xv\|_2^2 \geq 0$.

For the resulting matrix to be positive definite, $\|Xv\|_2^2 > 0$ must hold true for any non-zero vector v. $\|Xv\|_2^2 > 0$ means that any linear combination of the column vectors in X is a non-zero vector. In other words, all column vectors in X are linearly independent.

6. $g(x, y) = e^{(x+y)} + e^{3xy} + e^{y^4}$

   $\frac{\partial g}{\partial x} = e^{(x+y)} + 3ye^{3xy}$

   $\frac{\partial g}{\partial y} = e^{(x+y)} + 3xe^{3xy} + 4y^3e^{y^4}$

7. (a)

```python
import numpy as np
from numpy import linalg as LA

a = np.array([[2, 1, 3],
              [1, 1, 2],
              [3, 2, 5]])

eigenvectors = LA.eig(a)[1]

print(eigenvectors)

for i in range(3):
    print(a @ eigenvectors[:, i])
    print(a @ eigenvectors[:, i]/eigenvectors[:, i])
```

$$\begin{bmatrix} 2 & 1 & 3 \\ 1 & 1 & 2 \\ 3 & 2 & 5 \end{bmatrix} \begin{bmatrix} -0.49079864 \\ -0.31970025 \\ -0.81049889 \end{bmatrix} = \begin{bmatrix} -3.73279419 \\ -2.43149666 \\ -6.16429085 \end{bmatrix} = 7.60555128 \begin{bmatrix} -0.49079864 \\ -0.31970025 \\ -0.81049889 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 1 & 3 \\ 1 & 1 & 2 \\ 3 & 2 & 5 \end{bmatrix} \begin{bmatrix} -0.65252078 \\ 0.75130448 \\ 0.0987837 \end{bmatrix} = \begin{bmatrix} -0.25738599 \\ 0.29635109 \\ 0.0389651 \end{bmatrix} = 0.39444872 \begin{bmatrix} -0.65252078 \\ 0.75130448 \\ 0.0987837 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 1 & 3 \\ 1 & 1 & 2 \\ 3 & 2 & 5 \end{bmatrix} \begin{bmatrix} -0.57735027 \\ -0.57735027 \\ 0.57735027 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = 0 \begin{bmatrix} -0.57735027 \\ -0.57735027 \\ 0.57735027 \end{bmatrix}$$

(b) $A = \begin{bmatrix} -0.49079864 & -0.65252078 & -0.57735027 \\ -0.31970025 & 0.75130448 & -0.57735027 \\ -0.81049889 & 0.0987837 & 0.57735027 \end{bmatrix}$.

$$\begin{bmatrix} 7.60555128 & 0 & 0 \\ 0 & 0.39444872 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot$$
$$\begin{bmatrix} -0.49079864 & -0.31970025 & -0.81049889 \\ -0.65252078 & 0.75130448 & 0.0987837 \\ -0.57735027 & -0.57735027 & 0.57735027 \end{bmatrix}$$

(c) Rank of a square matrix is equal to the number of non-zero eigenvalues. There are 2 non-zero eigenvalues of A, so rank of A is 2.

(d) A is not positive definite because there is 1 non-positive eigenvalue.

(e) A is positive semi-definite because all eigenvalues are non-negative.

(f) A is singular because one of its eigenvalues is 0.

8. (a) $w^T x = w \cdot x = |w||x|cos(\theta)$ where θ is the angle between the vectors w and x. $w^T x$ is positive when $\theta < 90°$ and negative when $\theta > 90°$. The positive and negative regions are separated by the condition $w^T x = 0$.

$$w^T x = \begin{bmatrix} w_0 & w_1 & w_2 \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix} = w_0 + w_1 x_1 + w_2 x_2 = 0$$

$w_0 + w_1 x_1 + w_2 x_2 = 0$ is a general equation for a line.
In slope-intercept form, $x_2 = -\frac{w_1}{w_2} x_1 - \frac{w_0}{w_2}$

(b) $w^T = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \implies x_2 = -\frac{2}{3} x_1 - \frac{1}{3}$
$w^T = \begin{bmatrix} -1 & -2 & -3 \end{bmatrix} \implies x_2 = -\frac{2}{3} x_1 - \frac{1}{3}$

Both w's represent the same line (since each line/hyperplane has 2 normal vectors of opposite direction).

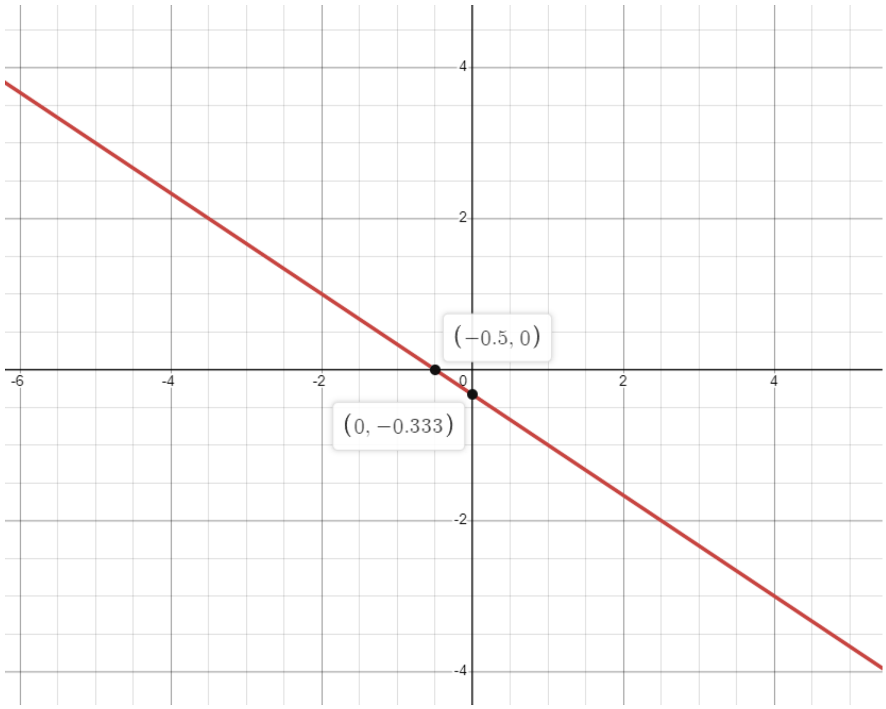

Figure 1: line representing $w^T x = 0$

9. (a) Refer to part (b).
   (b) The binary classifier was updated 11 times for the example below. g is close to f in this case, but in other cases, they are noticeably different.

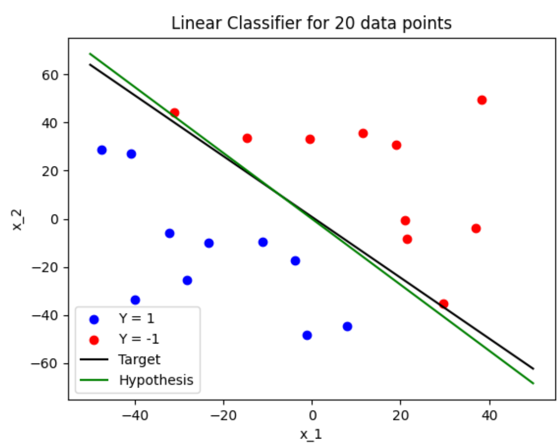

Figure 2: Linear Classifier for 20 data points

   (c) Compared to (b), g is noticeably different from f. The number of updates was smaller at 5 instead of 11.

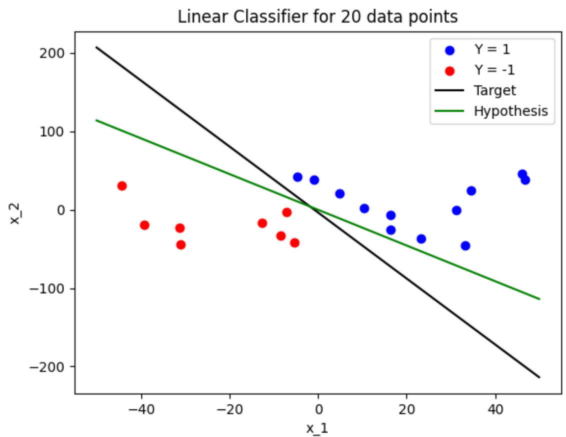

Figure 3: Linear Classifier for 20 data points

(d) As number of data points increases, g approaches f. The number of updates also generally increases.

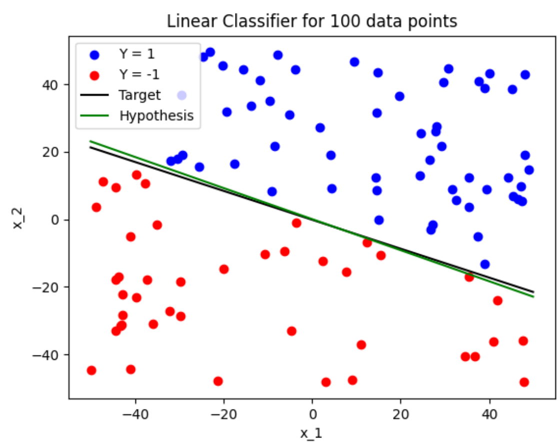

Figure 4: Linear Classifier for 100 data points

(e) As number of data points increases, g approaches f. The number of updates also generally increases.

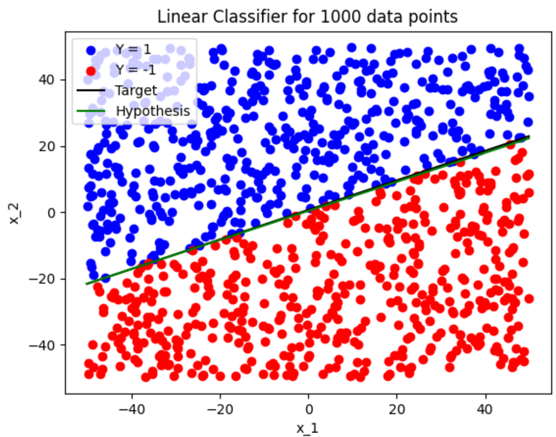

Figure 5: Linear Classifier for 1000 data points

Code used to generate the examples:

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

def main():
    dimension = 2
    data_set_count = 1000
    min_value = -50
    max_value = 50

    data_set = generate_data_set(dimension, data_set_count, min_value, max_value)
    f = data_set['f']
    x = data_set['x']
    y = data_set['y']

    positive = x[y > 0]
    negative = x[y < 0]

    g = 2*np.random.rand(dimension+1)-1

    t = 0
    while True:
        improved_g = improve(g, x, y)
        if np.array_equal(g, improved_g):
            break
        else:
            g = improved_g
        t += 1

    print(f'updated perception {t} times')
```

```
    plt.scatter(positive[:, 1], positive[:, 2], c='blue')
    plt.scatter(negative[:, 1], negative[:, 2], c='red')

    plot(f, min_value, max_value, color='black')
    plot(g, min_value, max_value, color='green')

    plt.xlabel('x_1')
    plt.ylabel('x_2')
    plt.title(f'Linear Classifier for {data_set_count} data points')
    plt.legend(['Y = 1', 'Y = -1', 'Target', 'Hypothesis'])
    plt.show()

def plot(normal, min, max, color):
    x = np.arange(min, max, 0.1)
    m = -normal[1]/normal[2]
    b = -normal[0]/normal[2]
    y = m*x+b

    plt.plot(x, y, c=color)

def improve(g, x, y):
    expected = np.sign(g @ x.T)
    actual = y
    misclassified = np.not_equal(expected, actual)

    if misclassified.any():
        missed_idx = np.argwhere(misclassified).T[0]
        i = np.random.randint(len(missed_idx))
        random_missed_index = missed_idx[i]

        x_star = x[random_missed_index]
        y_star = y[random_missed_index]

        return g + y_star*x_star
    else:
        return g

def generate_data_set(dimension, n, min_value, max_value):
    f = 2*np.random.rand(dimension+1)-1

    ones_vector = np.ones((n, 1))
    x = (max_value-min_value)*np.random.rand(n, dimension)-(max_value-min_value)/2
    x = np.hstack((ones_vector, x))
    y = np.sign(f @ x.T)

    return {'f': f, 'x': x, 'y': y}

if __name__ == '__main__':
    main()
```

10. (a) $p(orange) = p(orange \cap r) + p(orange \cap b) + p(orange \cap g) + p(orange \cap y) = 0.1\frac{2}{5+2+3} + 0.4\frac{1}{3+1+7} + 0.3\frac{8}{1+8+4} + 0.2\frac{9}{1+9+2} \approx 0.391$

    (b) $p(r|mango) = \frac{p(mango|r)p(r)}{p(mango)} = \frac{(\frac{3}{5+2+3})(0.1)}{[\frac{3}{5+2+3})(0.1)+\frac{7}{3+1+7})(0.4)+\frac{4}{1+8+4})(0.3)+\frac{2}{1+9+2})(0.2)]} \approx 0.0731$