Convex Optimization

Lab 1: Gradient Descent/Ascent

Lecturer: *Dr*. Wan-Lei Zhao

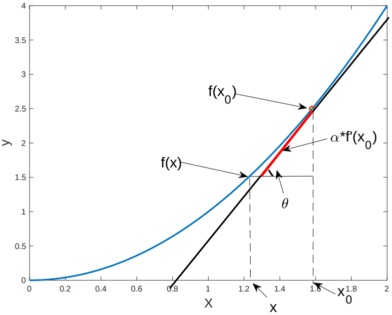
Autumn Semester 2024

Contact: wlzhao@xmu.edu.cn

Find out Extreme values by Gradient Descent

1. *Initialize*(x)
2. Repeat
   a. $x^{+} := x - \alpha f'(x)$
   b. $x = x^{+}$

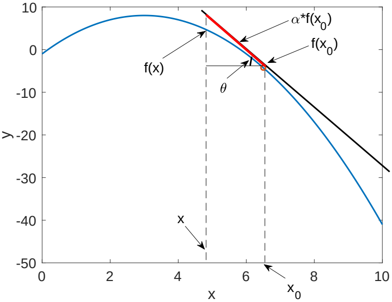
- Once we reaches the extreme, $f'(x) = 0$
- The procedure converges



Find out Extreme values by Gradient Ascent

1. *Initialize*(x)
2. Repeat
   a. $x^+ := x + \alpha f'(x)$
   b. $x = x^+$

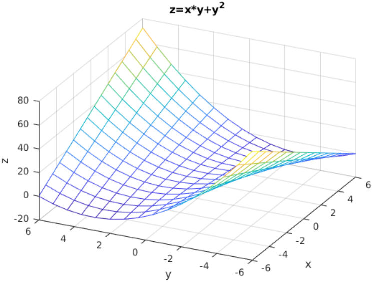
- Once we reaches the extreme, $f'(x) = 0$
- The procedure converges

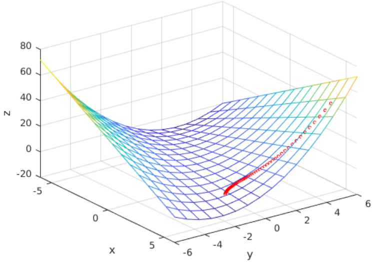


- Requirements:
  1. Implement the gradient descent and ascent procedure in Matlab
  2. Find out the extreme values for following functions
  3. Visualize the descent/ascent steps of your procedure
  4. Build the animation for the procedure, if possible

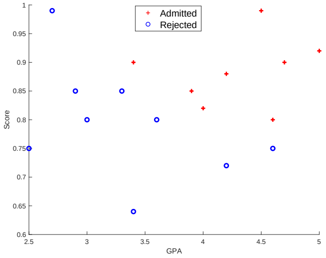
Problem-1

- Given function $z = x * y + y^2,\ x, y \in [-6,\ 6]$
- The initial point is $x = 5, y = 5$
- Please find the local minimal of the function

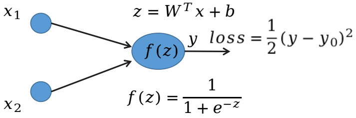




Problem-2: train a two-class neural-network classifier



(c)



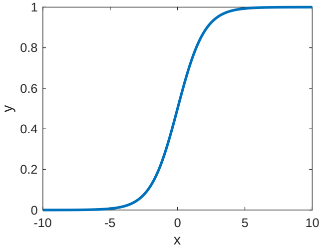
(d)

Figure: The problem and the framework of the neural network

$$f(z) = \frac{1}{1+e^{-z}} \tag{1}$$

$$f'(z) = f(z)(1 - f(z)) \tag{2}$$

To work out the gradient for all the variables



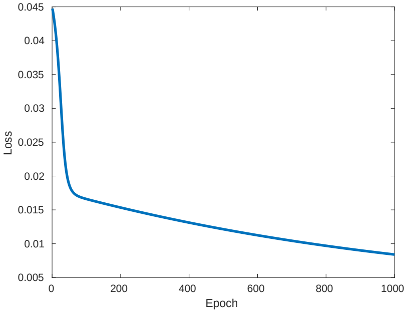
- Loss function: $E = \frac{1}{2}(y - y_0)^2$
- The variables in the hidden layer: x_1, x_2, and b
- According to chain of derivative, we have

$$\frac{\partial E}{\partial w_1} = (y - y_0) \cdot f(z) \cdot (1 - f(z)) \cdot x_1 \quad (3)$$

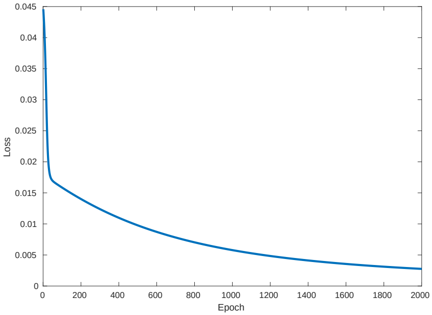
$$\frac{\partial E}{\partial w_2} = (y - y_0) \cdot f(z) \cdot (1 - f(z)) \cdot x_2 \quad (4)$$

$$\frac{\partial E}{\partial b} = (y - y_0) \cdot f(z) \cdot (1 - f(z)) \quad (5)$$

Your task and the Answer



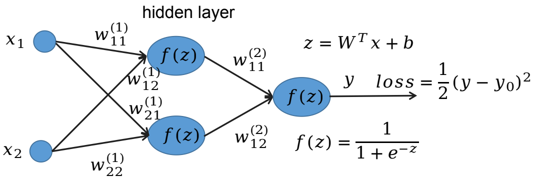
(a) epoch=1,000



(b) epoch=2,000

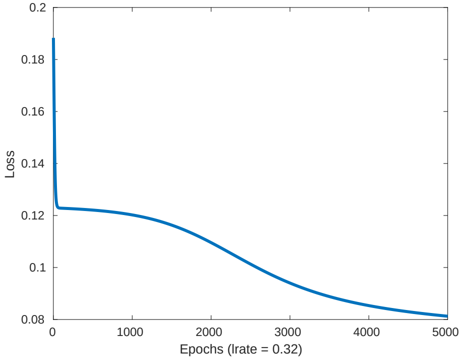
- Please try to implement the training of the network based on gradent descent
- Train the simple network with the provided data
- Resutls for reference: $w_1 = 0.337$, $w_2 = -0.3204$, $b = -1.0062$
- Resutls for reference: $w_1 = 0.9138$, $w_2 = -0.8317$, $b = -2.7849$

A Two-layer Network (1)



- Try to train the above network by gradient descent
- The weights to be learned are:
- $W_{11} = [w_1, w_2, b],\ W_{12} = [w_1, w_2, b],\ W_{21} = [w_1, w_2, b]$

A Two-layer Network (2)



- Resutls for reference: $W_{11} = [1.0155, -0.8313, -2.4823]^T$, $W_{12} = [0.8804, -0.5581, -2.2472]^T$
- $W_{21} = [3.8164, 2.6523, -4.2436]^T$

A Two-layer Network (3)

```
function [Y21] = Lay2Predict(X)
    W11 = [1.0155, -0.8313, -2.4823]';
    W12 = [0.8804, -0.5581, -2.2472]';
    W21 = [3.8164,  2.6523, -4.2436]';
    Y11 = sigmoid(X*W11);
    Y12 = sigmoid(X*W12);
    X21 = [Y11, Y12, 1];
    Y21 = sigmoid(X21*W21);
end
```

Code for prediction

Problem-3

- Given function $z = x^3 + y^3$, $x, y \in [-6,\ 6]$
- The initial point is $x = -5, y = -5$
- Please find the local maximal of the function