23/06/28

Chapter 4

Binary Classification

- determine T/F of an object
- Linear Regression

  Linear Function

  w1 * x1 + w2 * x2 + b = z (There can be multiple weights)

  -> z = b + $\sum w_i x_i$

- Step Function

  if input >= 0, return 1
  else, return -1

- Perceptron Algorithm

  Linear Function -> z-> Step Function ->backpropagation -> y_hat

- Adaline Algorithm (Adaptive Linear Neuron)

  Linear Function -> z -> backpropagation -> Step Function -> y_hat

- Logistic regression

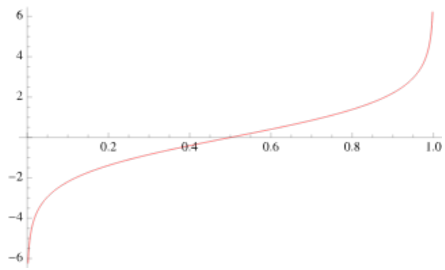
  Linear Function -> z -> Activation Function -> a -> backpropagation
  -> Threshold Function (Step Function) -> y_hat

  Activation Function : Non-Linear, Sigmoid Function

  - converts z into a value between 0~1

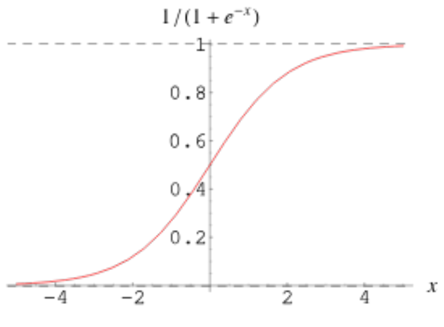
  - OR (odds ratio) = $\frac{p}{1-p}$

  - logit(p) (Logit Function) = $\log(\frac{p}{1-p})$ ⇨



  - Sigmoid Function (Logistic Function) : Inverse function of logit function

$$p = \frac{1}{1+e^{-z}} \quad (0 < p < 1)$$



23/06/28

In Conclusion, Logistic Regression consists of

Activation function -> adjusts z to a value between 0 and 1 (interpret as probability)
Step function -> adjust a to 0 or 1 (greater than 0.5?)

- BackPropagation or Logistic Regression

We cannot use the same loss function as linear regression but Increase the percentage of correct output data

-> Cross Entropy Loss Function

$$L = -(y\log(a) + (1-y)(log(1-a))$$

a : output of Activation(Sigmoid) function
y : target

because binary classification has targets of 0 or 1,

y	L
1	$-\log(a)$
0	$-\log(1-a)$

if we make both values of L minimum (difference is minimum), a approaches y. In order to achieve minimum, differentiate by w(weight), b(intercept)

$$\frac{d}{dw_i}L = -(y-a)x_i \qquad \frac{d}{db}L = -(y-a)1$$

Update w, b based on logistic loss function

$$w_i = w_i - \frac{dL}{dw_i} = w_i + (y-a)x_i$$

$$b = b - \frac{dL}{db} = b + (y-a)1$$