	Pred P	Pred N
Actual P	TP	FN
Actual N	FP	TN

Precision = TP / (TP + FP) | FPR = FP / (FP + TN)

Recall (TP rate) = TP / (TP + FN) | F1 score = $2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$

Accuracy = (TP + TN) / Total

under fitting → train and validation acc low | overfitting → train high validation low

training accuracy decreases with more data, Validation acc increases

Roc Curve → TPR on y axis, FPR on x axis, AUC Agg perf measure

AUC → Prob model, ranks random pos over random negative, = 1 perfect, = .5 Random guess

underfitting is bias → inherit error in the mod

overfitting is high variance → over specialized

linear regression: $\lambda = 0$ no reg, too big parameter → 0

$$\min_{\theta} = \frac{1}{2} \sum_{i=1}^{m} \left(y^i - \theta^T \cdot x^i\right)^2 + \frac{\lambda}{2} \|\theta\|^2$$

Regularized loss

$J(\theta)$: cost/loss func

$R(\theta)$: Regularizer

λ Regularizer Parameter

$$J_\lambda(\theta) = J(\theta) + \lambda R(\theta)$$

$$g = g\left(\sum_{i=1}^{n} w_i \cdot x_i\right)$$ linear combo inputs, non activation linear function

$$g = g(w_0 + w^T x)$$

$$X = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad W = \begin{pmatrix} w_1 \\ \vdots \\ w_n \end{pmatrix}$$

Sigmoid Function

$g(z) = \sigma(z) = \frac{1}{1 + e^{-z}}$

$g'(z) = \sigma(z)(1 - \sigma(z))$

Hyperbolic Tangent

$g(z) = \sigma(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$

$g'(z) = 1 - g(z)^2$

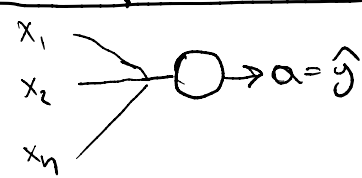
Rectified Linear Unit (ReLU)

$g(z) = \max\{0, z\}$

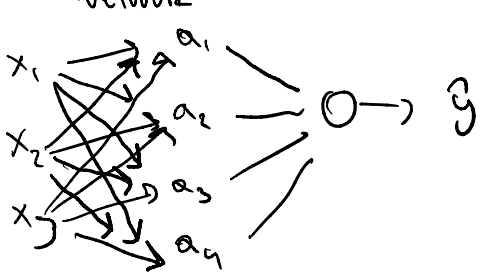
$g'(z) = \begin{cases} 1, & n > 0 \\ 0, & \text{otherwise} \end{cases}$

All activation Func

Perceptron



Neural Network

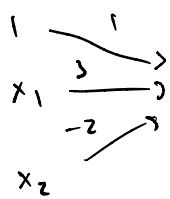


$$z_1 = \begin{pmatrix} z_1^1 \\ z_2^1 \\ z_3^1 \\ z_4^1 \end{pmatrix} = \begin{pmatrix} w_1^{1T} \\ w_2^{1T} \\ w_3^{1T} \\ w_4^{1T} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} b_1^1 \\ b_2^1 \\ b_3^1 \\ b_4^1 \end{pmatrix}$$

when going to next layer index Subscript

$z_1^1 = w_1^{1T} x + b_1^1$, $a_1^1 = \sigma(z_1^1)$ $z^1 = w_3^{1T} x + b_3^1$, $a_3^1 = \sigma(z_3^1)$

$z_2^1 = w_2^{1T} x + b_2^1$, $a_2^1 = \sigma(z_2^1)$ $z_4^1 = w_4^1 x + b_4^1$, $a_4^1 = \sigma(z_4^1)$



$\hat{y} = g(w_0 + w^T x)$

$= g\left(1 + \begin{pmatrix} 3 \\ -2 \end{pmatrix}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}\right)$

$= g(1 + 3x_1 - 2x_2)$

given input $\begin{pmatrix} -1 \\ 2 \end{pmatrix}$

$\hat{y} = g(1 + 3(-1) - 2(2))$

$= g(-6) = \frac{1}{1 + e^{-(-6)}} = 0.002$

$z^{11} = W^1 x^1 + b^1$

$z^{12} = W^1 x^2 + b^1$

$z^{13} = W^1 x^3 + b^1$

$z^1 = W^1 X + b^1 \rightarrow W^1 \begin{pmatrix} | & | & | \\ x^1 & x^2 & x^3 \\ | & | & | \end{pmatrix}$