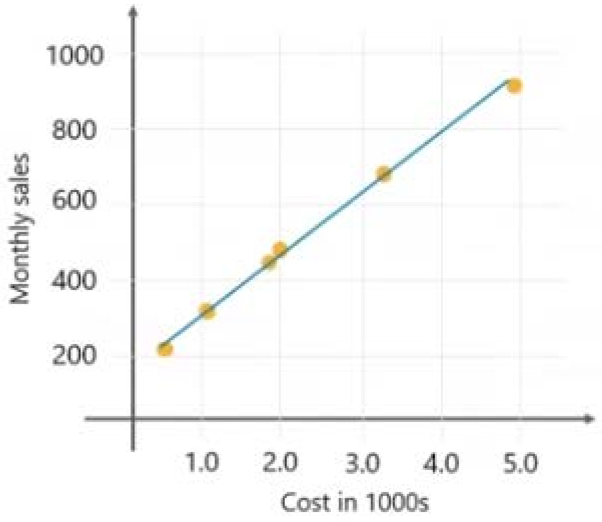
Logistic Regression

Linear Regression Use Case

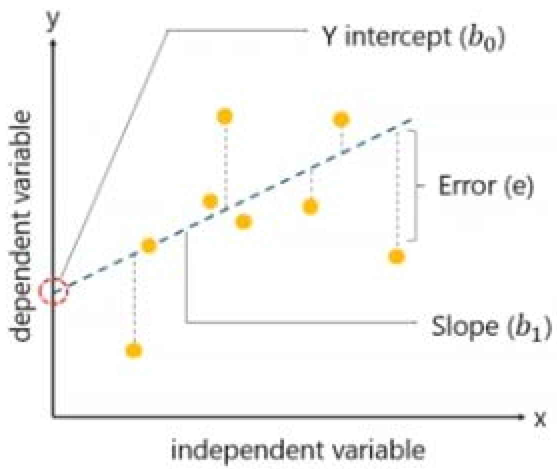
To forecast monthly sales by studying the relationship between the monthly e-commerce sales and the online advertising costs.

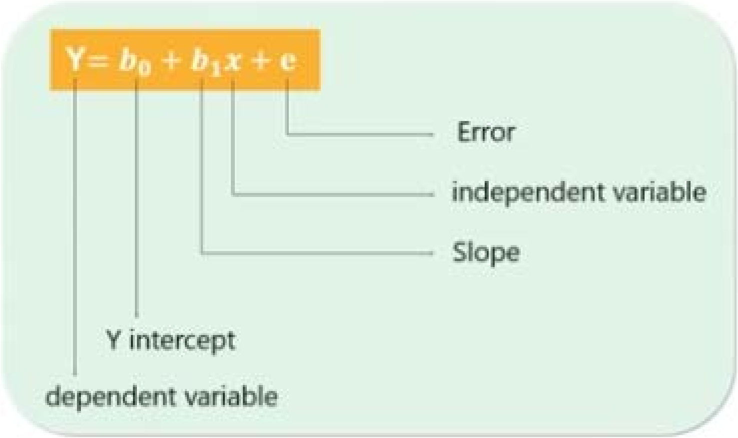
Monthly sales	Advertising cost In 1000s
200	0.5
900	5
450	1.9
680	3.2
490	2.0
300	1.0



What Is Linear Regression?

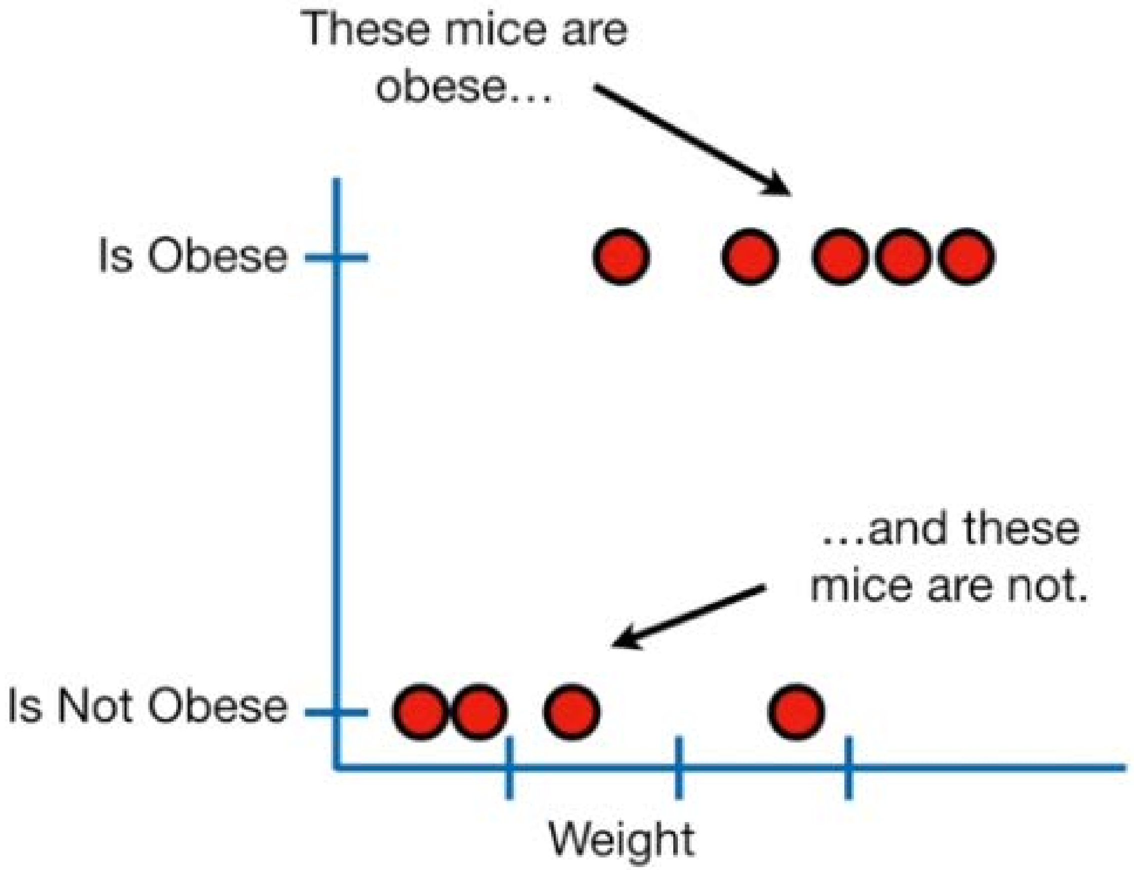
Linear Regression is a method to predict dependent variable (Y) based on values of independent variables (X). It can be used for the cases where we want to predict some continuous quantity.

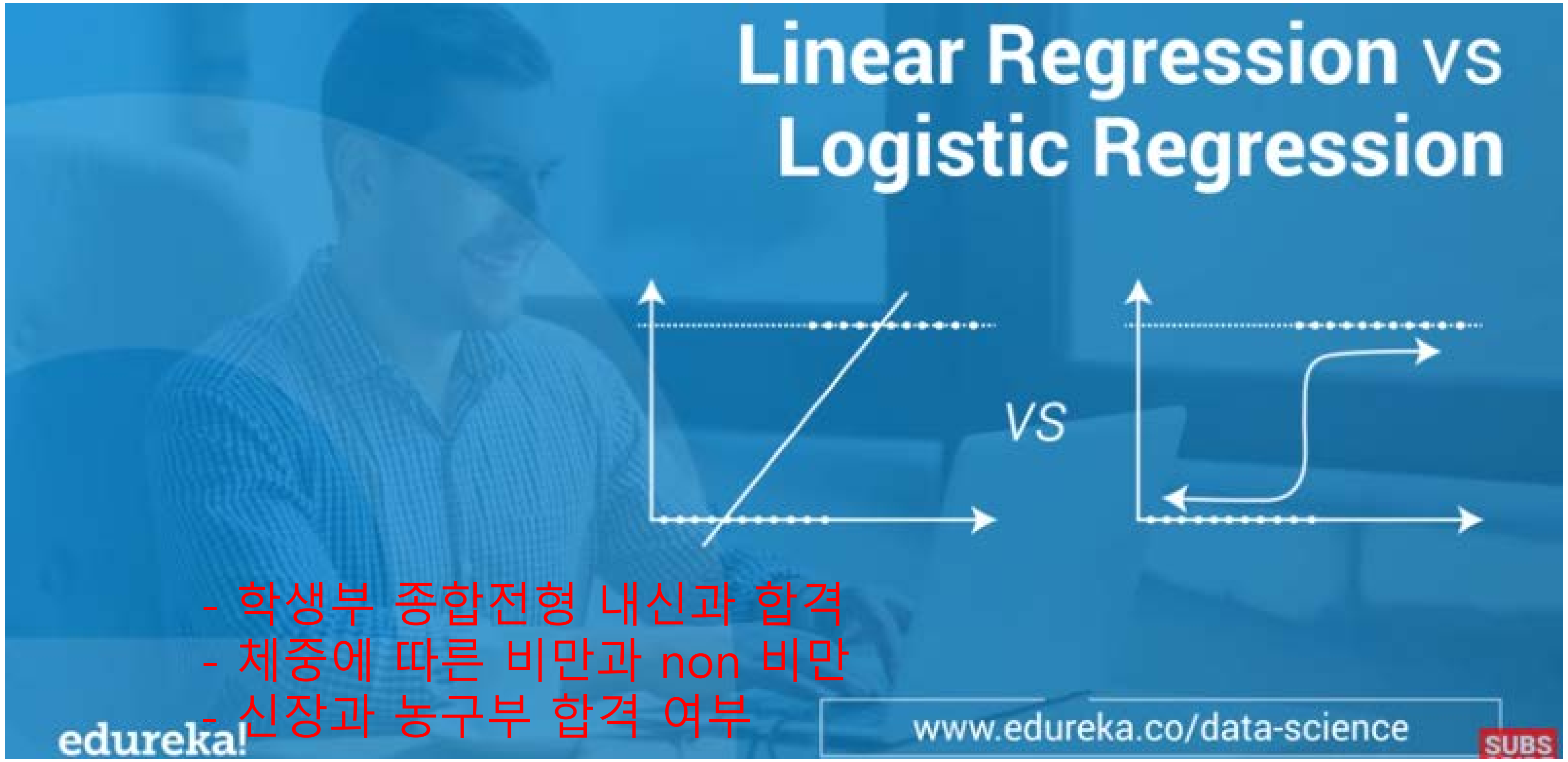




SUBS

- 학생부 종합전형 내신과 합격
- 체중에 따른 비만과 non 비만
- 신장과 농구부 합격 여부



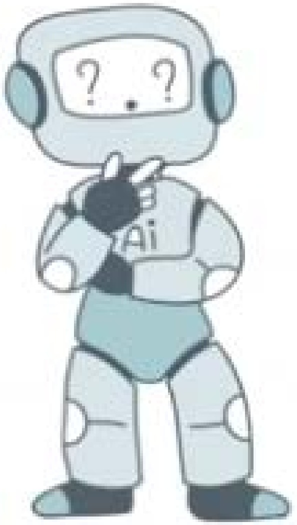


- 학생부 종합전형 내신과 합격
- 체중에 따른 비만과 non 비만
- 신장과 농구부 합격 여부

Types Of Machine Learning

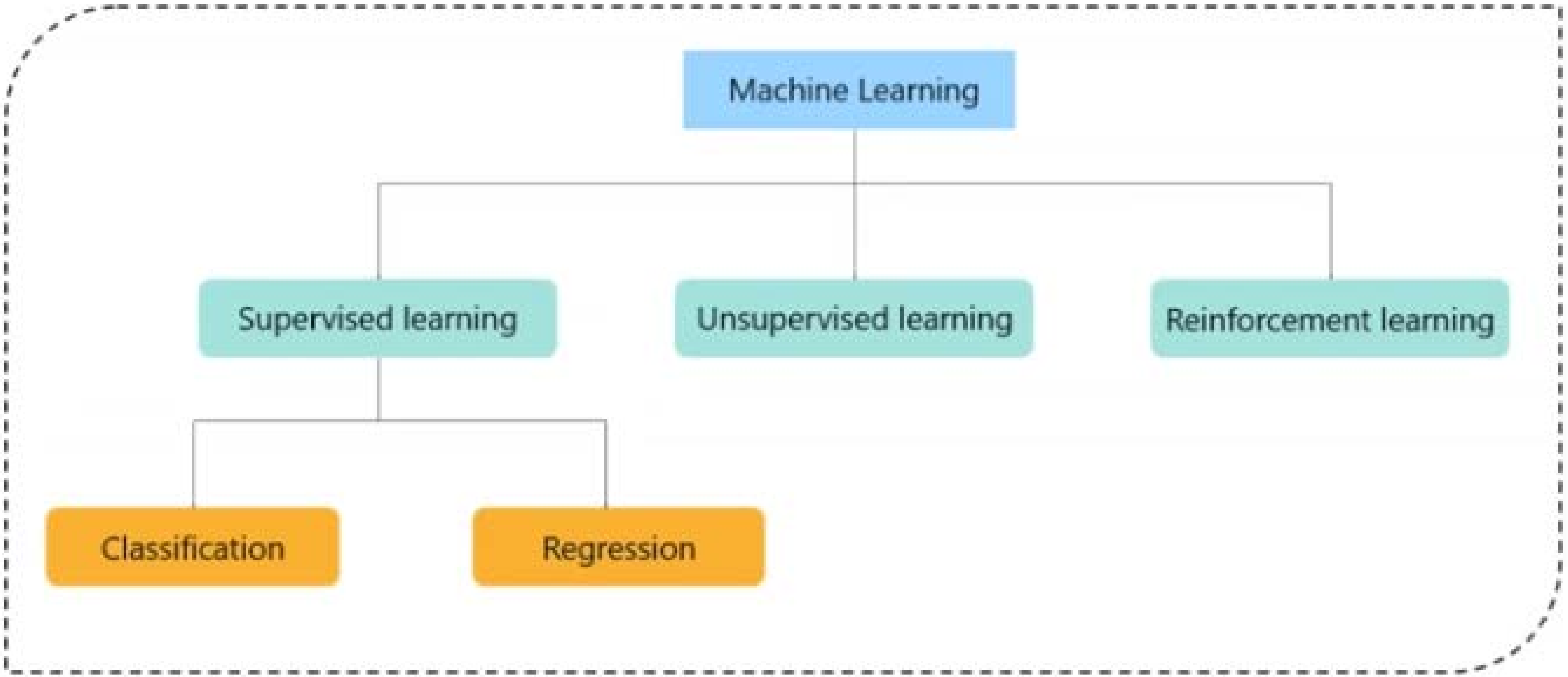
Supervised Learning

Unsupervised Learning



Reinforcement Learning

Regression And Classification

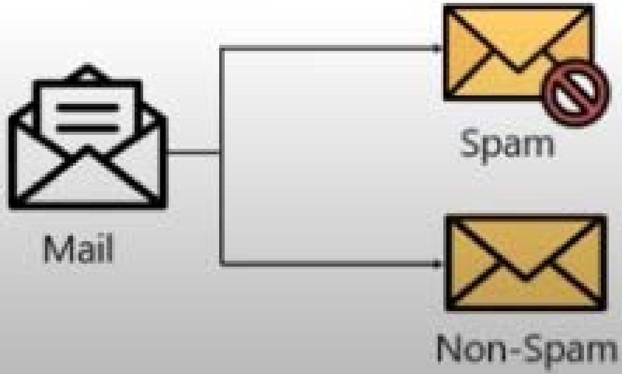


Regression Vs Classification

Classification

Classification is the task of predicting a discrete class label

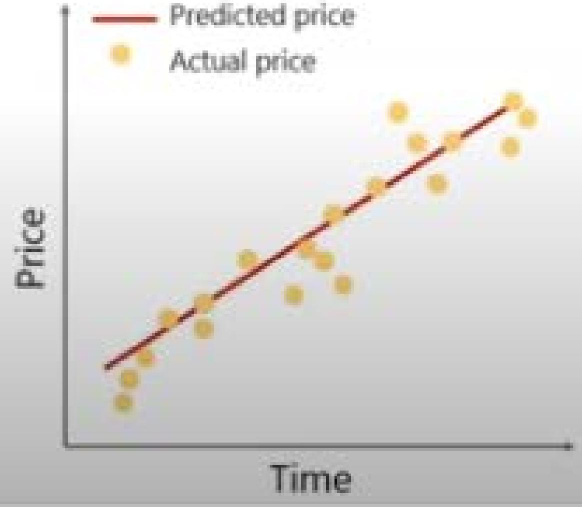
- In a classification problem data is classified into one of two or more classes
- A classification problem with two classes is called binary, more than two classes is called a multi-class classification

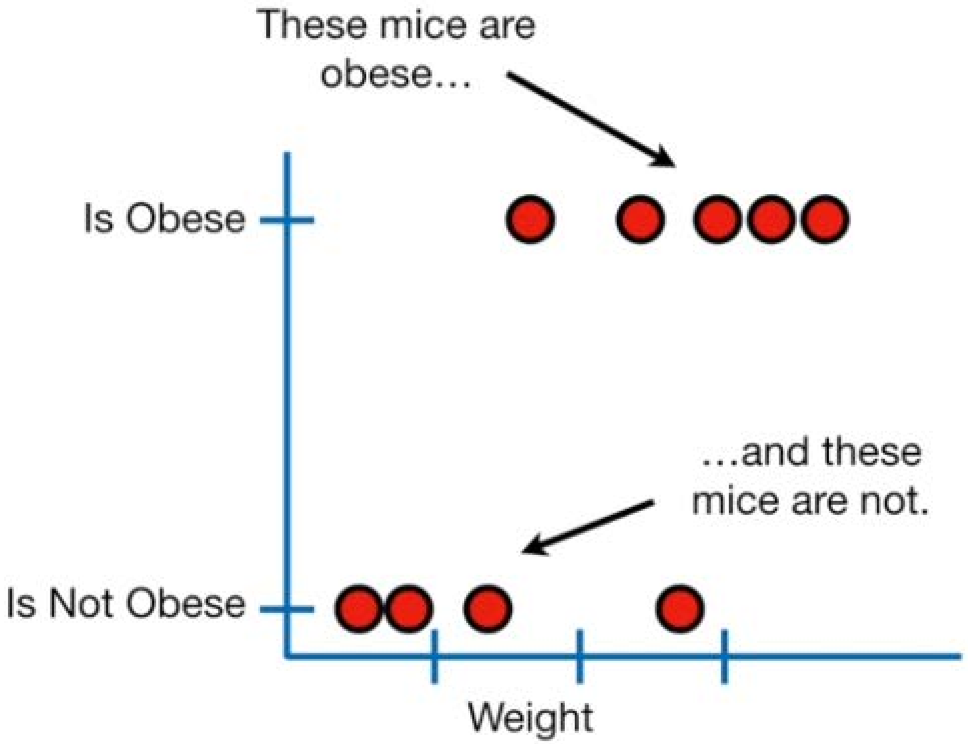


Regression

Regression is the task of predicting a continuous quantity

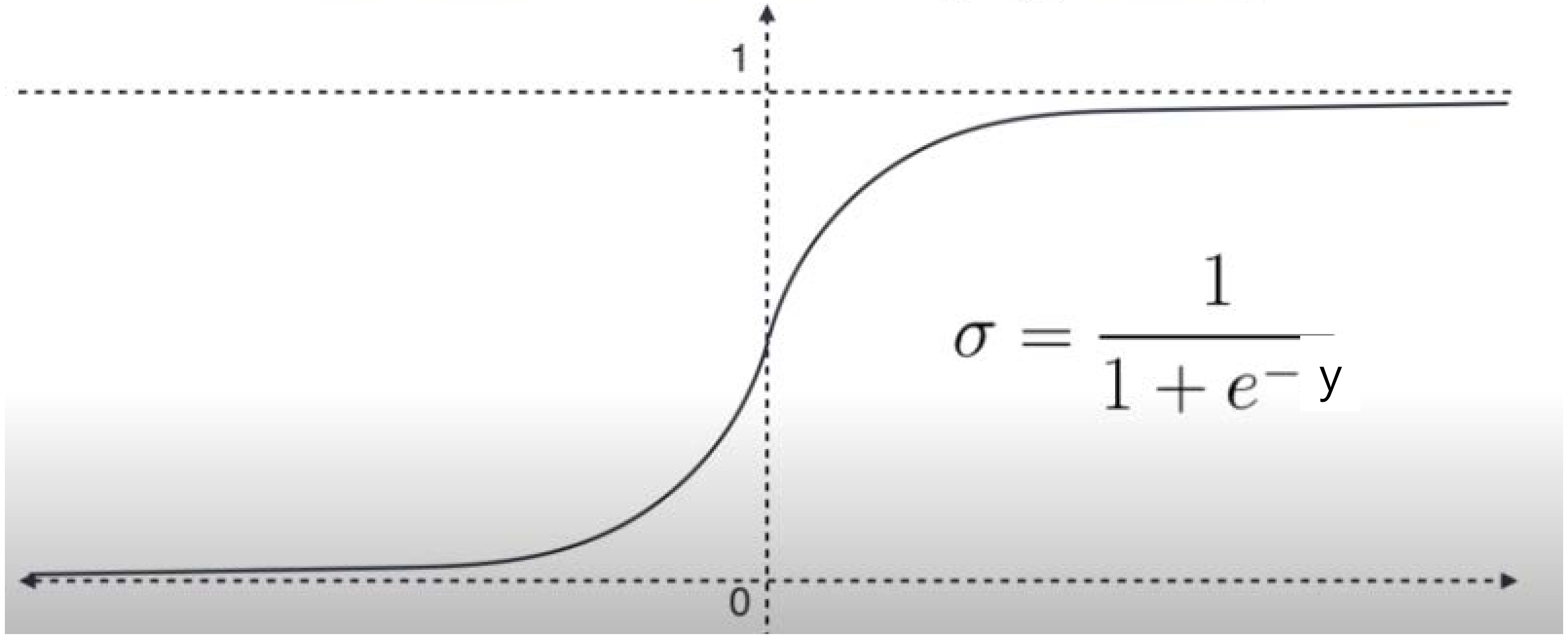
- A regression problem requires the prediction of a quantity
- A regression problem with multiple input variables is called a multivariate regression problem





9.8	합격	1
8.7	합격	1
7.8	합격	1
7.2	불합격	0
6.9	합격	1
6.2	불합격	0
5.9	합격	1
5.5	불합격	0
4.3	불합격	0
3.8	불합격	0
3.2	불합격	0
2.8	불합격	0

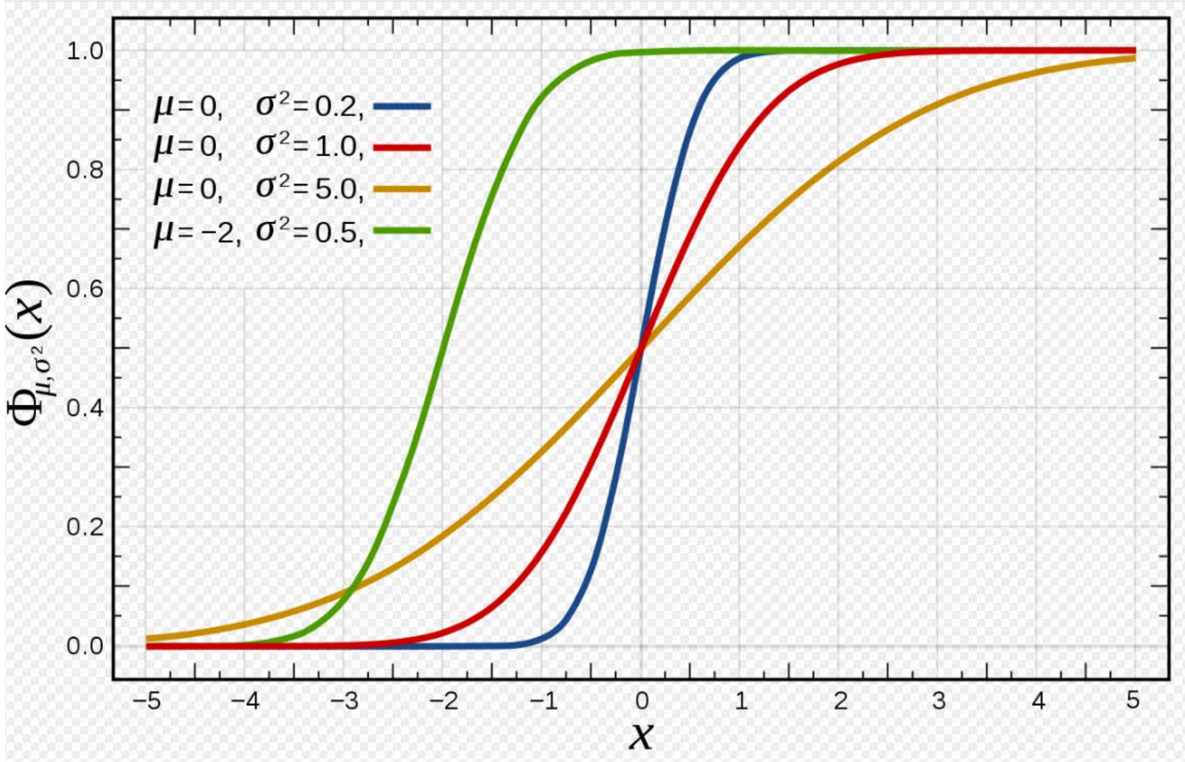
Activation function (sigmoid)



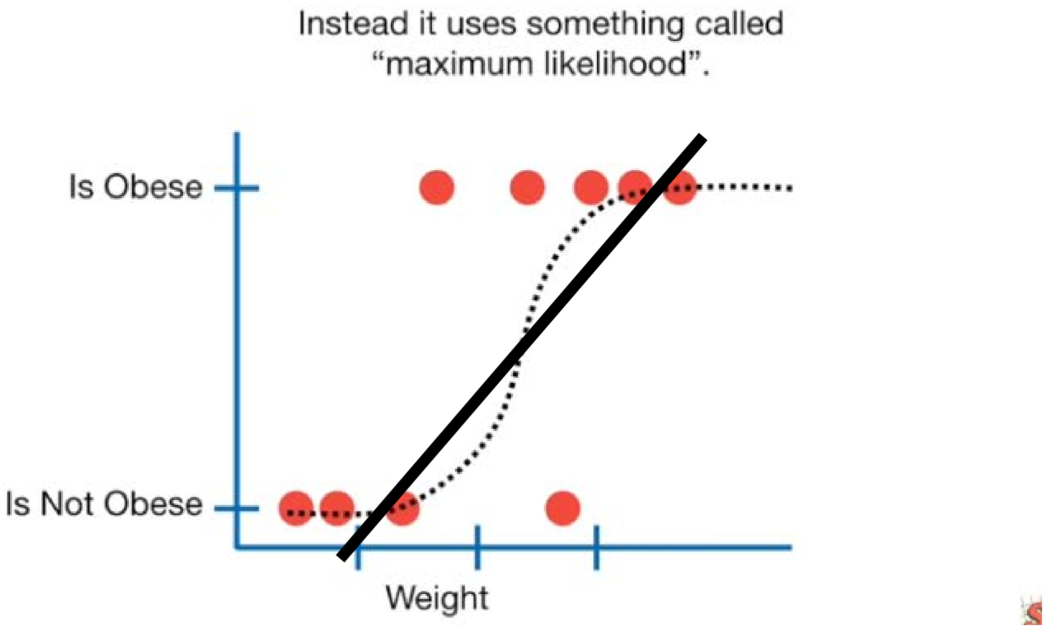
$y = ax + b$

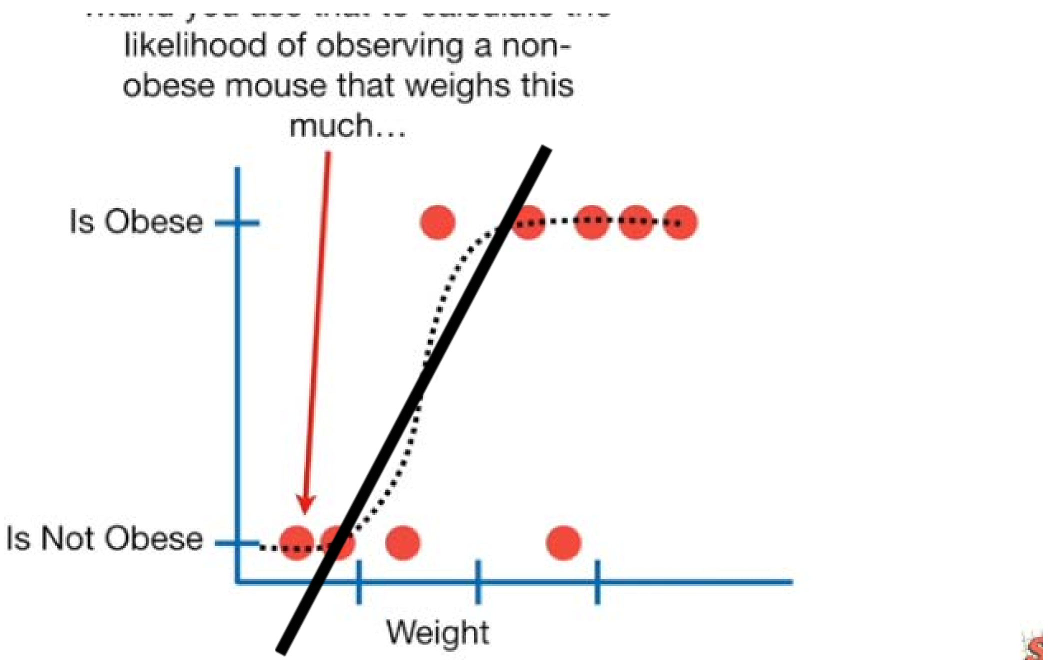
Likelihood estimation of logistic probability

- Cumulative Logistic distribution (sigmoid function)

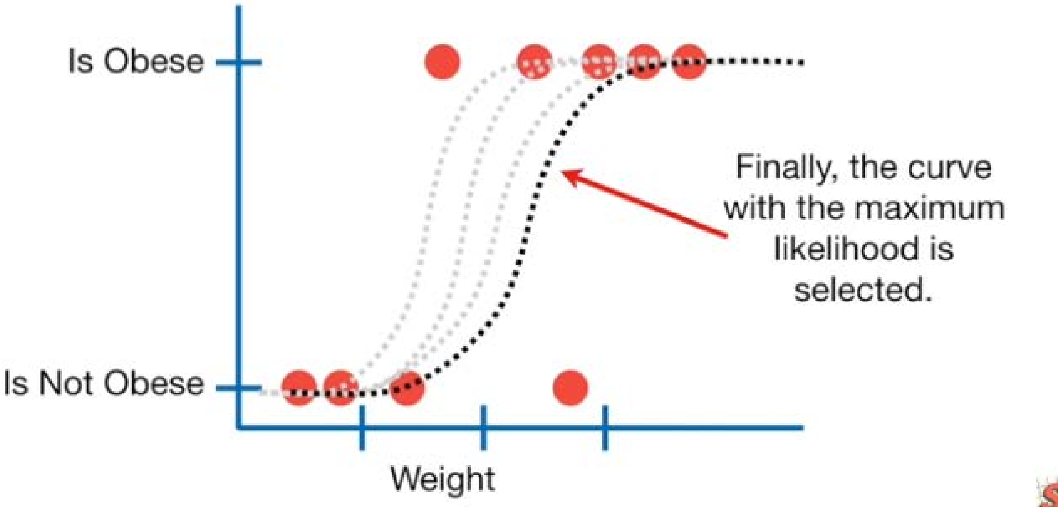


$$\frac{1}{1+e^{-(x-\mu)/s}}$$





Y = aX + b 를 계수
a와 b를 변화시키면
서 error가 가장 적게
만들어주는
regression을 구한다



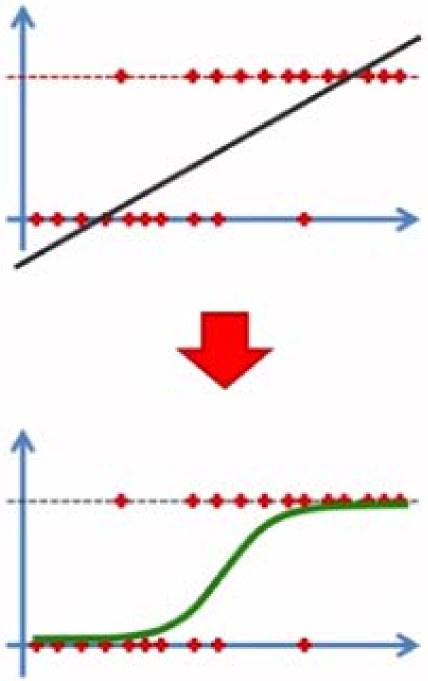
Logistic Regression

$$y = b_0 + b_1 * x$$

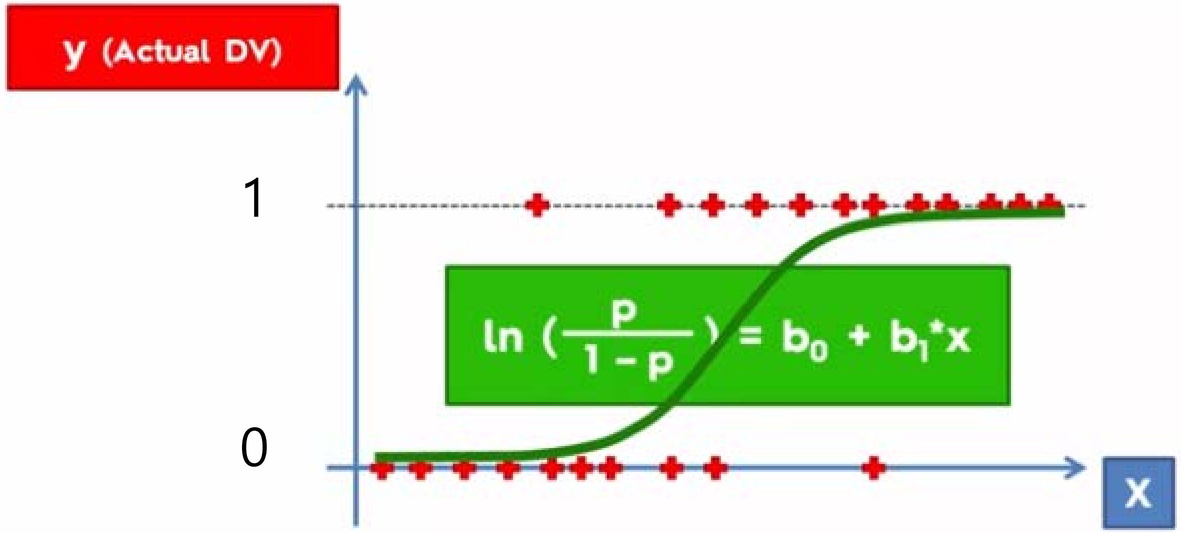
Sigmoid Function

$$p = \frac{1}{1 + e^{-y}}$$

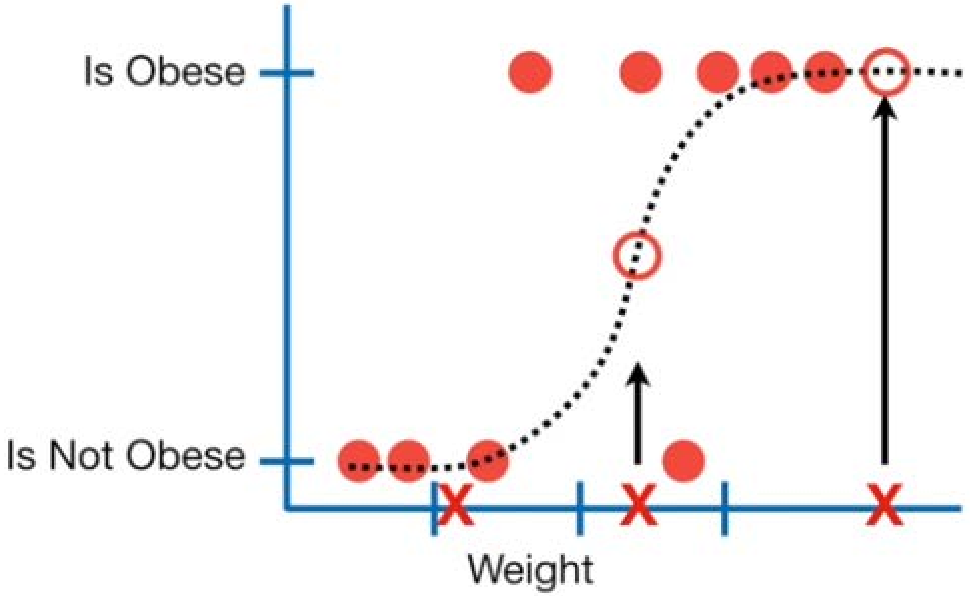
$$\ln\left(\frac{p}{1-p}\right) = b_0 + b_1 * x$$



Logistic Regression



Although logistic regression tells the probability that a mouse is obese or not, it's usually used for classification.



```
  Incomes GymVisits State Hours PayOrNot
1     100         4    TX   9.3      Yes
2      50         3    CA   4.8       No
3     100         4    TX   8.9      Yes
4     100         2    NY   6.5      Yes
5      50         2    MD   4.2       No
6      80         2    CA   6.2       No
> 

> logAnalysis <- glm(PayOrNot~
```

Logistic Regression Example

Joe wants to sell virtual goods in his fitness app, but he's not sure if users would pay for the virtual goods. He's interested in answers to the following questions:

(1) If a user's usage hours can influence whether she would buy the virtual goods in the app.

(2) If a user's yearly incomes and gym visit frequency can influence whether she would buy the virtual goods in the app.

Report and explain the analysis results.

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)    
(Intercept) -10.4147     1.9555  -5.326 1.00e-07 ***
Hours         1.3167     0.2595   5.075 3.88e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

$$\Pr(Yes) = \frac{1}{1 + e^{-y}}$$

$$y = -10.41 + 1.32 * Hours$$

e.g., A user uses the app for 10 hours.

y = -10.41 + 1.32 * 10 = 2.79

Pr (Yes) = 1/(1+exp(-2.79)) =0.942

Logistic Regression Example

Joe wants to sell virtual goods in his fitness app, but he's not sure if users would pay for the virtual goods. He's interested in answers to the following questions:

(1) If a user's usage hours can influence whether she would buy the virtual goods in the app.

(2) If a user's yearly incomes and gym visit frequency can influence whether she would buy the virtual goods in the app.

Report and explain the analysis results.

Multiple Predictors

An entertainment analyst wants to predict the likelihood for a movie to win an Oscar. His assumption is the more nominations a movie receives, the more likely it is for the movie to win an Oscar. Let p denote the probability for a movie to win an Oscar, x_1 denote the number of nominations the movie receives, x_2 denote the number of Golden Globes the movie receives, and x_3 denote whether the movie is a comedy. Thus,

$$p = \frac{1}{1+e^{-(b_0 + b_1 * x_1 + b_2 * x_2 + b_3 * x_3)}}$$

Analysis Results – Coefficient & p-value

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)    
(Intercept) -11.45136    2.33997  -4.894 9.89e-07 ***
Incomes       0.08719    0.02214   3.939 8.19e-05 ***
GymVisits     0.99202    0.30036   3.303 0.000957 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

$$\Pr(Yes) = \frac{1}{1+e^{-y}}$$

$$y = -11.45 + 0.09 * Incomes + 0.99 * GymVisits$$

e.g., A user has $90000 yearly incomes, and goes to gym 5 times a week.

y = -11.45 + 0.09*90 + 0.99*5 = 1.6

Pr (Yes) = 1/(1+exp(-1.6))

Recap of Logistic Regression

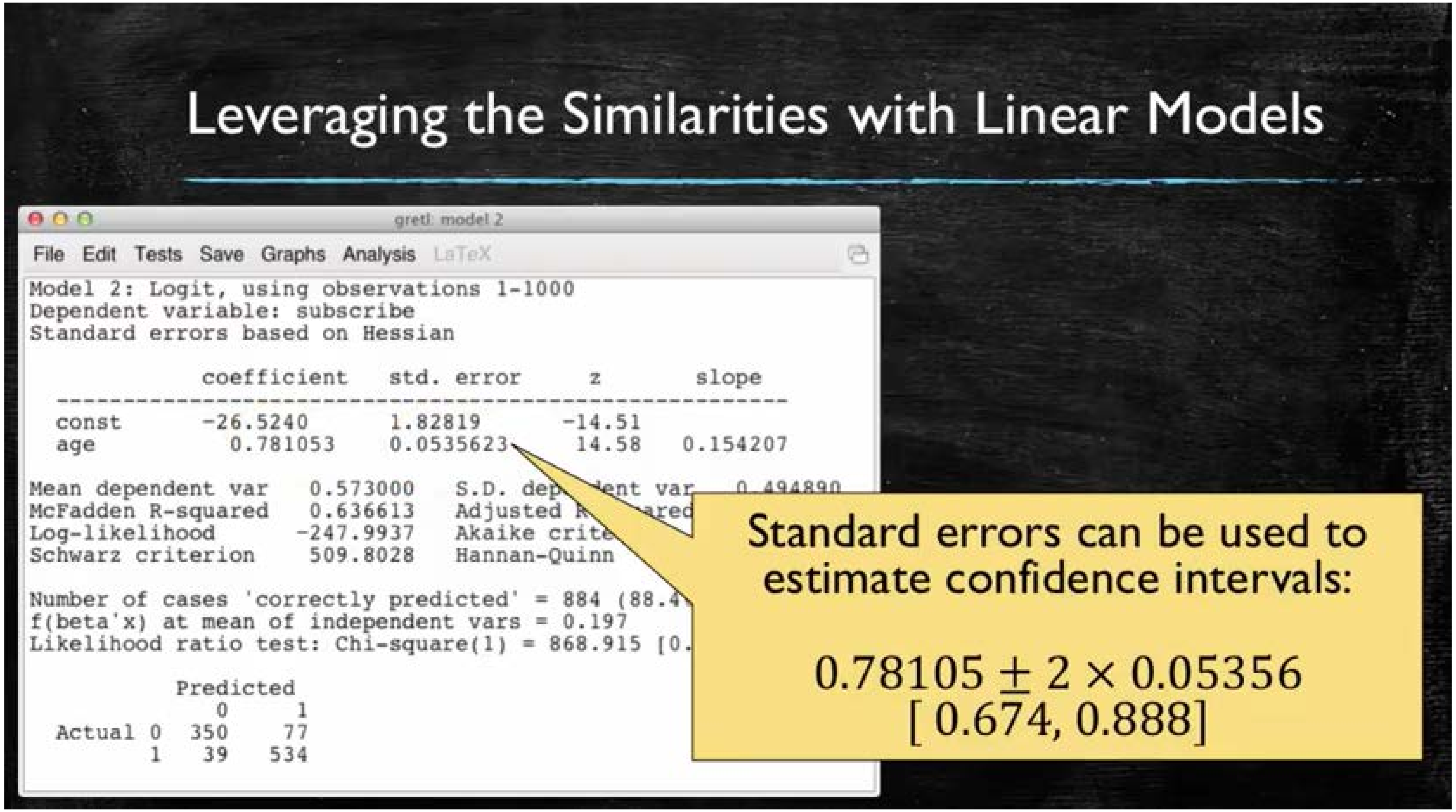
- In prior video we discussed why we can't (shouldn't) use linear models when we have a binary dependent variable.
- Instead, we use a logistic regression with the model:

$$\ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

- If we call that expression y^*, we have something that looks very familiar, a linear function:

$$y^* = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

나이에 따른 신문 구독 여부



What changed?

- We can no longer interpret the (magnitude of) the coefficients as we did before.
- What is the meaning of 0.78 in our estimated model?

$$\ln\left(\frac{p}{1-p}\right) = -26.524 + 0.781\, age$$

- For every unit increase of *age*, $\ln\left(\frac{p}{1-p}\right)$ increases 0.78 units.
- But what is $\ln\left(\frac{p}{1-p}\right)$? Remember we called it y^*

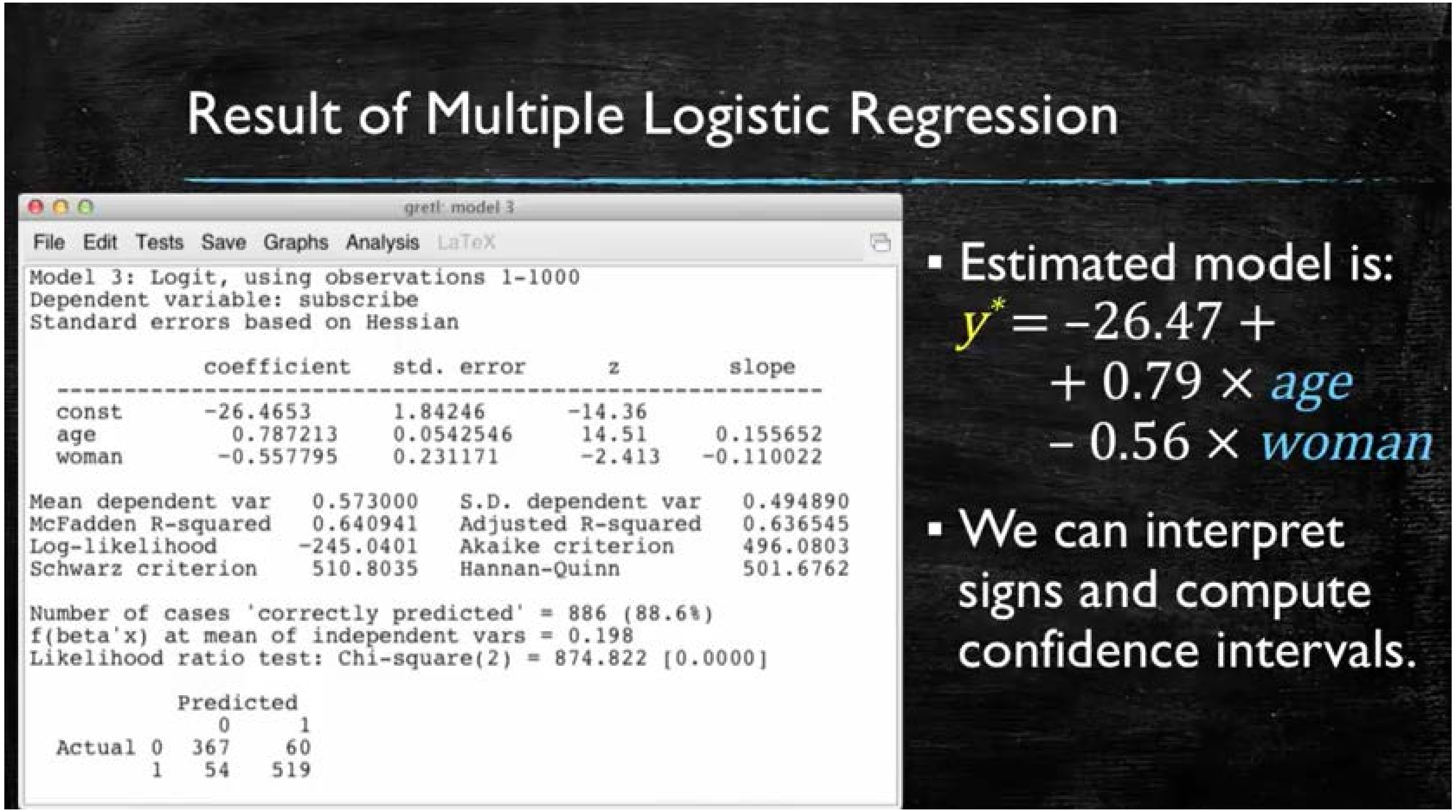
From y^* to p

- If we have: $y^* = \ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1\, age$
- Then: $p = \dfrac{\exp(\beta_0 + \beta_1\, age)}{\exp(\beta_0 + \beta_1\, age) + 1} = \dfrac{e^{\beta_0 + \beta_1\, age}}{e^{\beta_0 + \beta_1\, age} + 1}$
- Or simply put: $p = \dfrac{\exp(y^*)}{\exp(y^*) + 1} = \dfrac{e^{y^*}}{e^{y^*} + 1}$

From y^* to p in Excel

Variable	Coefficients	35y	36y	25y	26y	45y	46y
Constant	-26.524000	1	1	1	1	1	1
Age	0.781053	35	36	25	26	45	46
y* = ln(p/(1-p))		0.813	1.594	-7	-6.22	8.623	9.404
p = exp(y*)/(exp(y*)+1)		0.693	0.831	9E-04	0.002	1	1
Change			0.138		0.001		1E-04

나이와 성별에 따른 신문 구독 여부



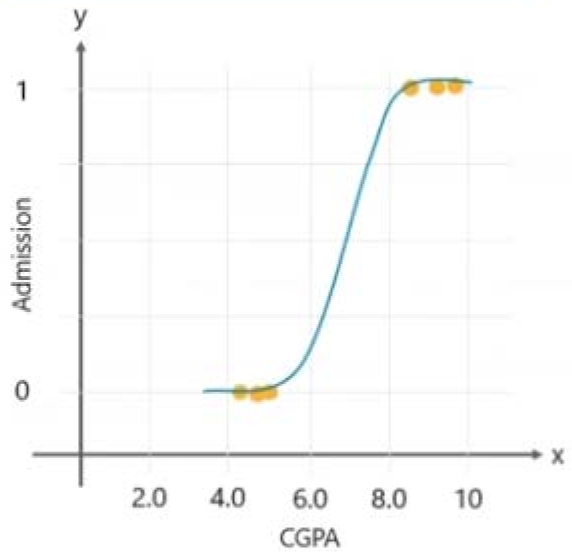
From y^* to p in Excel

Variable	Coefficients	35y W	35y M	36y W	36 M
Constant	-26.465300	1	1	1	1
Age	0.787213	35	35	36	36
Woman	-0.557795	1	0	1	0
y* = ln(p/(1-p))		0.529	1.087	1.317	1.874
p = exp(y*)/(exp(y*)+1)		0.629	0.748	0.789	0.867
Change			0.119		0.078

Logistic Regression Use Case

To predict if a student will get admitted to a school based on his CGPA.

Admission	CGPA
0	4.2
0	5.1
0	5.5
1	8.2
1	9.0
1	9.1

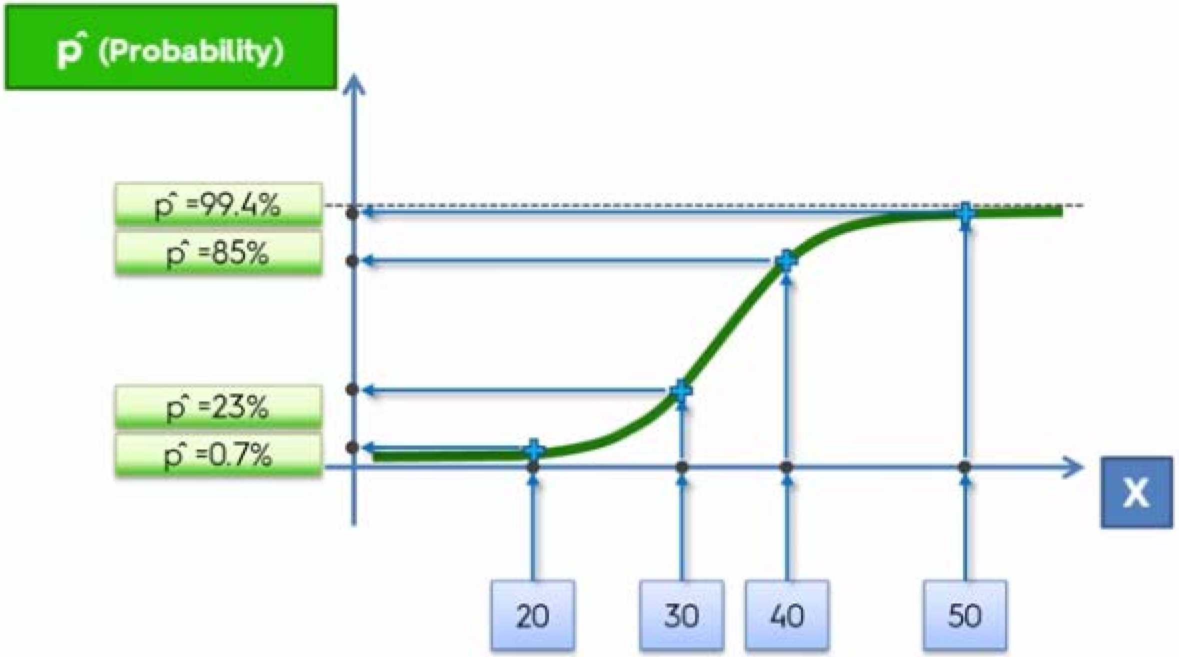


$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost(h_\theta(x^{(i)}), y^{(i)})$$

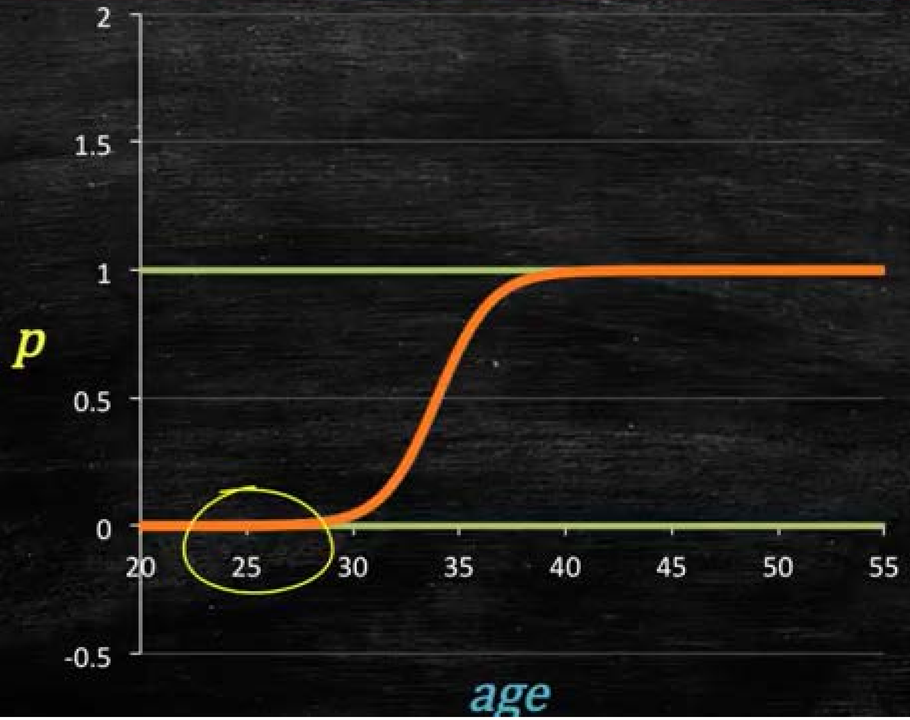
Cross Entropy

$$J(\theta) = \frac{1}{m}[\sum_{i=1}^{m} -y^{(i)} log(h_\theta(x^{(i)})) + (1 - y^{(i)}) log(1 - h_\theta(x^{(i)}))]$$

Logistic Regression

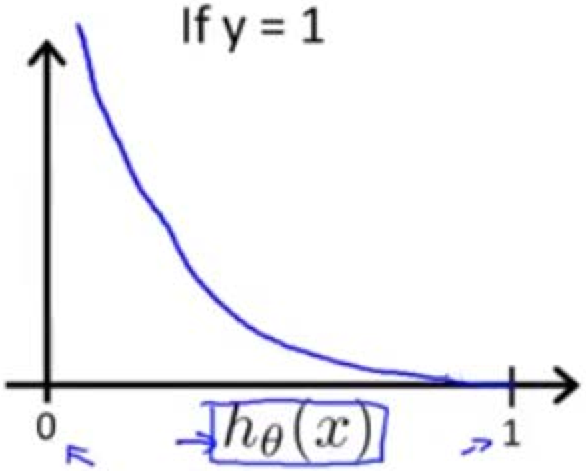


Logistic Model Plot



Logistic regression cost function

$$\mathrm{Cost}(h_\theta(x), y) = \begin{cases} \boxed{-\log(h_\theta(x))} & \text{if } y = 1 \\ -\log(1 - h_\theta(x)) & \text{if } y = 0 \end{cases}$$



Cost $= 0$ if $y = 1, h_\theta(x) = 1$

But as $h_\theta(x) \to 0$

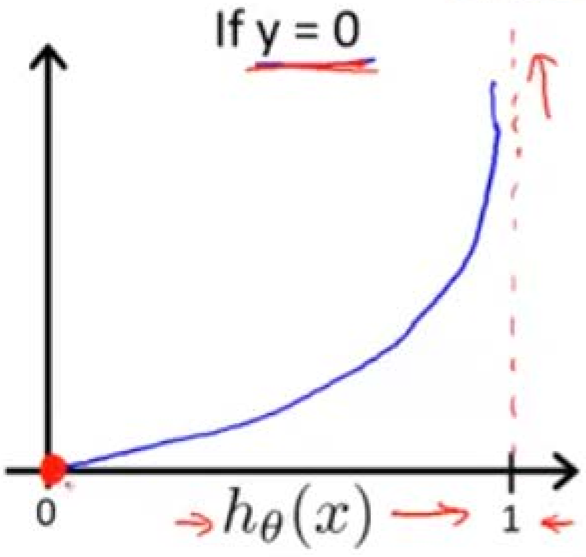
$Cost \to \infty$

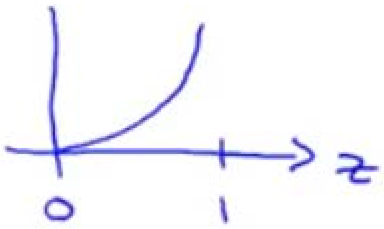
Captures intuition that if $h_\theta(x) = 0$, (predict $P(y = 1|x; \theta) = 0$), but $y = 1$, we'll penalize learning algorithm by a ver large cost.

9.8	합격	1
8.7	합격	1
7.8	합격	1
7.2	불합격	0
6.9	합격	1
6.2	불합격	0
5.9	합격	1
5.5	불합격	0
4.3	불합격	0
3.8	불합격	0
3.2	불합격	0
2.8	불합격	0

Logistic regression cost function

$$\mathrm{Cost}(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & \text{if } y = 1 \\ \boxed{-\log(1 - h_\theta(x))} & \text{if } y = 0 \end{cases}$$





Andrew

Cost function

Linear regression: $J(\theta) = \boxed{\frac{1}{m}\sum_{i=1}^{m} \cancel{\frac{1}{2}\left(h_\theta(x^{(i)}) - y^{(i)}\right)^2} \; \mathrm{cost}(h_\theta(x^{(i)}), y)}$

$$\mathrm{Cost}(h_\theta(x^{(i)}), y^{(i)}) = \frac{1}{2}\left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

Logistic Regression Cost Function

cross - entropy

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost(h_\theta(x^{(i)}), y^{(i)})$$

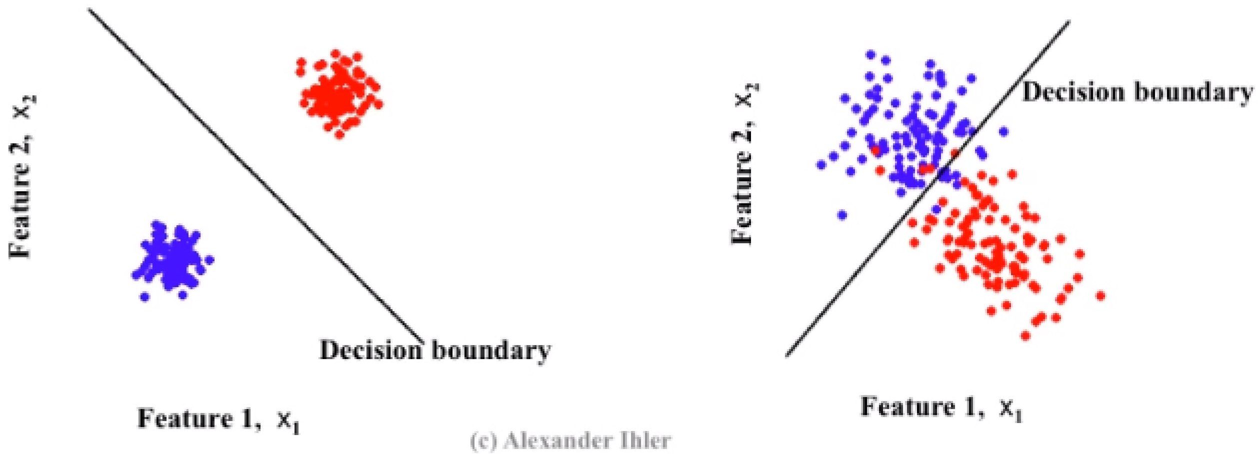
$$J(\theta) = \frac{1}{m}[\sum_{i=1}^{m} -y^{(i)} log(h_\theta(x^{(i)})) + (1 - y^{(i)}) log(1 - h_\theta(x^{(i)}))]$$

$$m = number\ of\ samples$$

Linear Classifiers (Perceptrons)

- Linear Classifiers
  - a linear classifier is a mapping which partitions feature space using a linear function (a straight line, or a hyperplane)
  - separates the two classes using a straight line in feature space
  - in 2 dimensions the decision boundary is a straight line

Linearly separable data   Linearly non-separable data



E-mail spam classifier

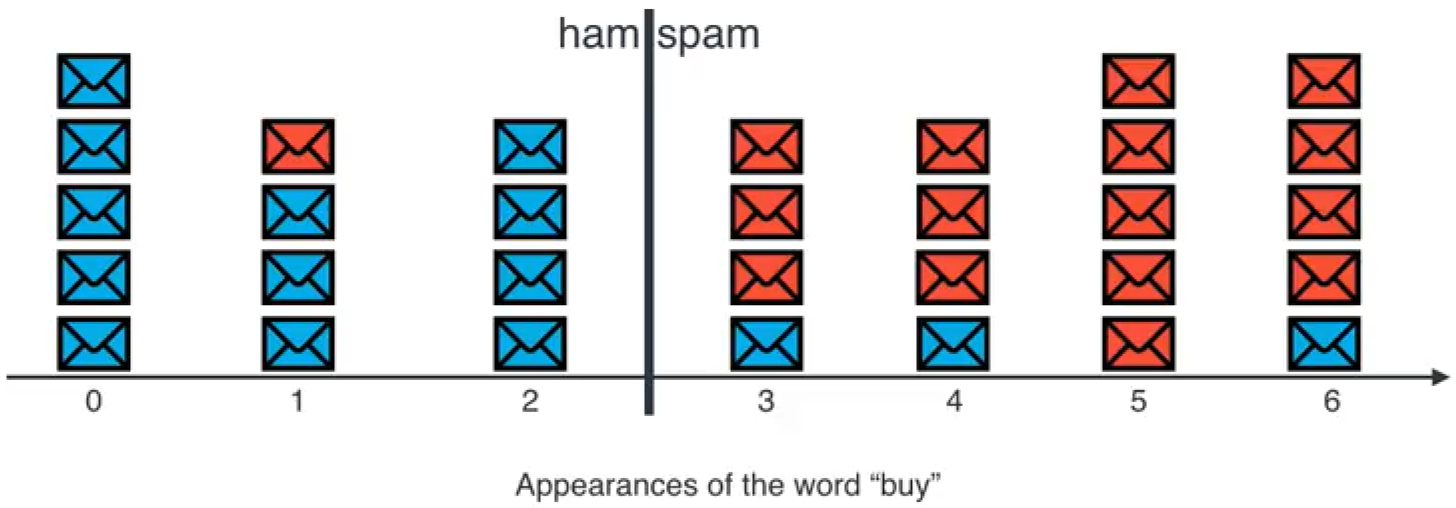
Spam

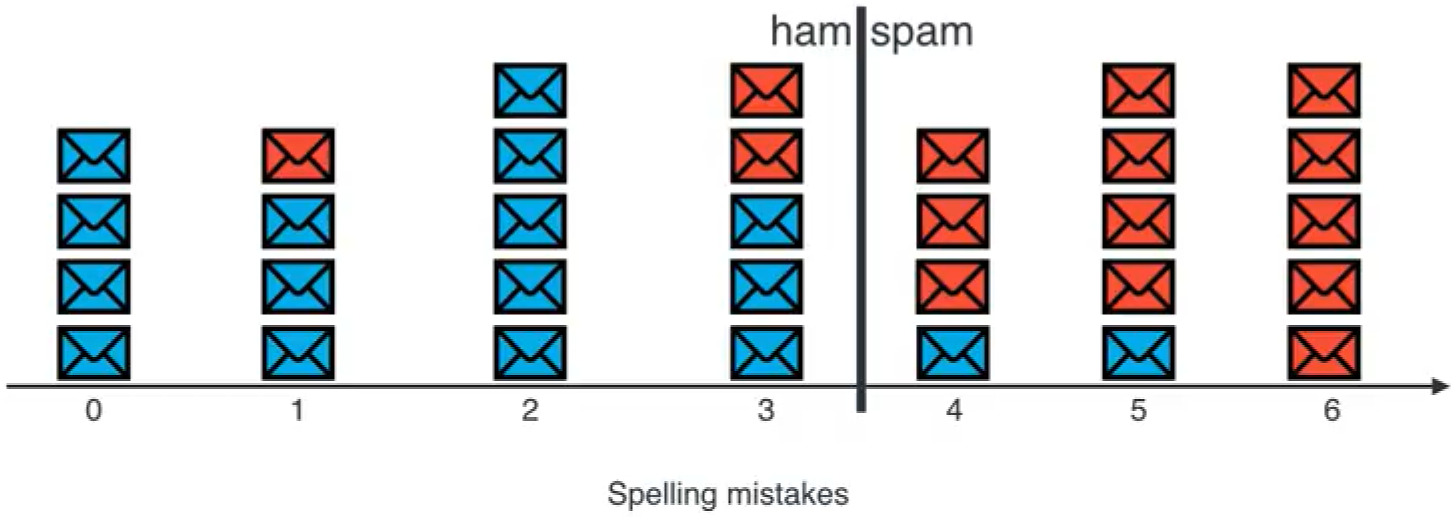
Buy, l0ts of money, now, che@p buy buy free mon3y

Non-spam (ham)

Hello grandson, I made cookies. Love, Grandma

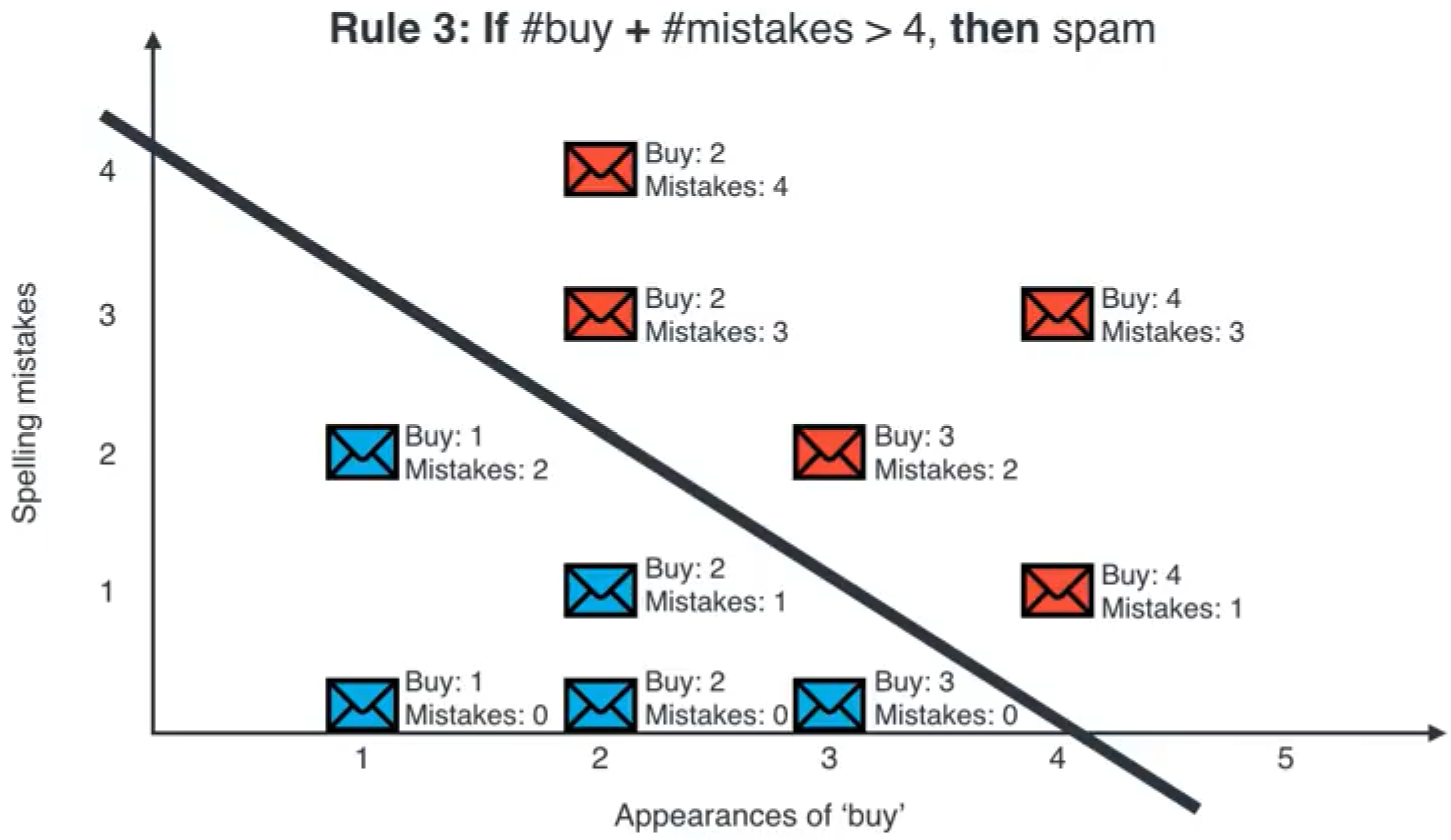
Rule 1: If #appearances of the word ‘buy’ > 2, then spam



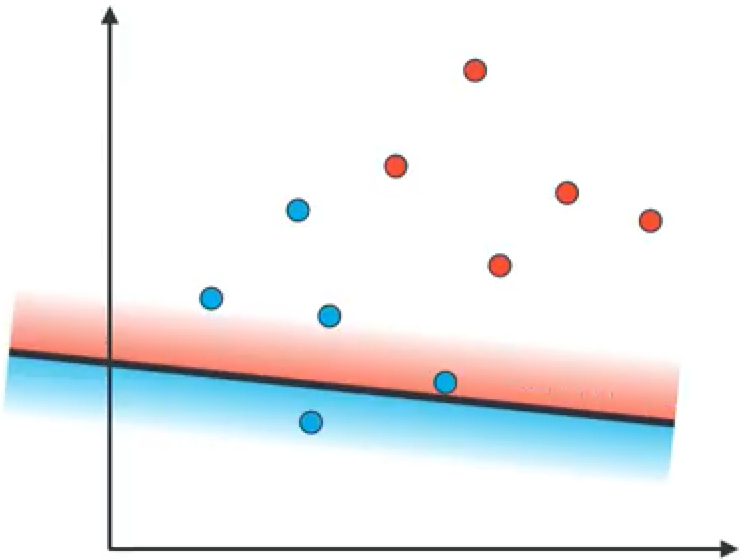

ham
spam
0
1
2
3
4
5
6
Spelling mistakes

E-mails

- Buy: **4**, Mistakes: **3**
- Buy: **1**, Mistakes: **2**
- Buy: **0**, Mistakes: **2**
- Buy: **2**, Mistakes: **1**
- Buy: **4**, Mistakes: **1**
- Buy: **0**, Mistakes: **3**
- Buy: **2**, Mistakes: **3**
- Buy: **0**, Mistakes: **1**
- Buy: **2**, Mistakes: **4**
- Buy: **3**, Mistakes: **2**


Rule 3: If #buy + #mistakes > 4, then spam
Spelling mistakes
Appearances of ‘buy’
Buy: 2
Mistakes: 4
Buy: 2
Mistakes: 3
Buy: 4
Mistakes: 3
Buy: 1
Mistakes: 2
Buy: 3
Mistakes: 2
Buy: 2
Mistakes: 1
Buy: 4
Mistakes: 1
Buy: 1
Mistakes: 0
Buy: 2
Mistakes: 0
Buy: 3
Mistakes: 0

Perceptron algorithm



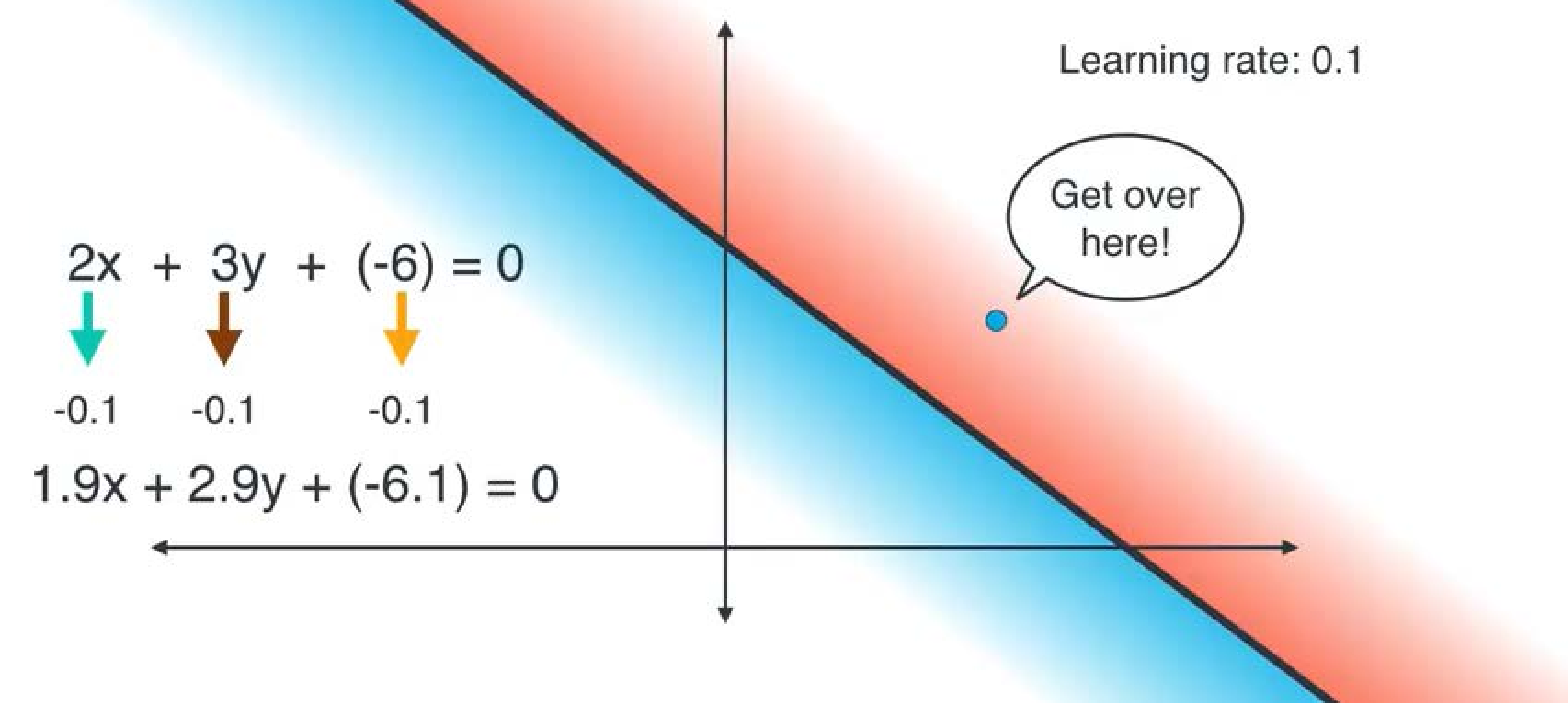
Step 1: Start with a random line
with blue and red sides.

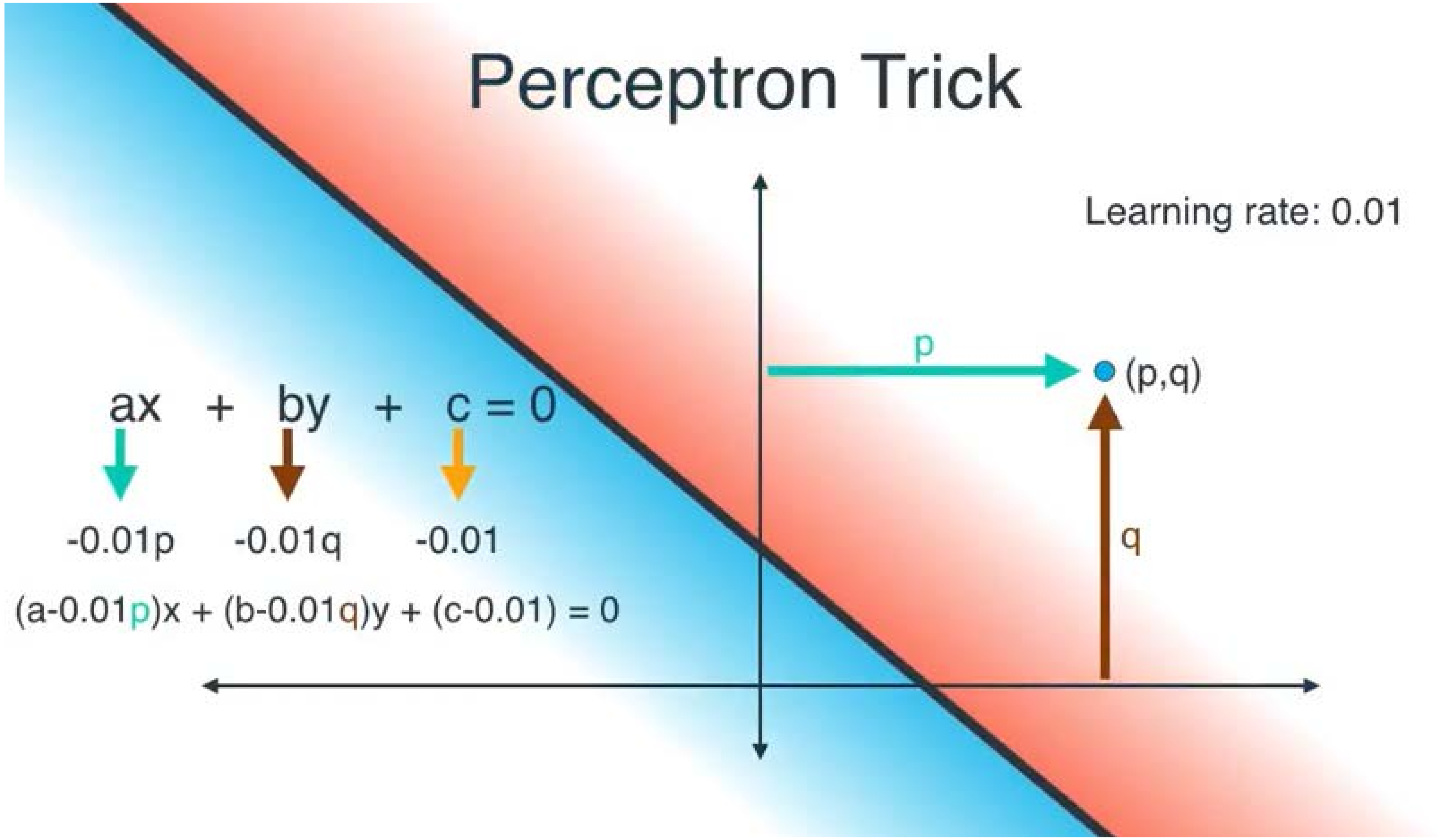
Step 2: Pick a large number. 1000
(number of repetitions, or epochs)

Step 3: (repeat 1000 times)

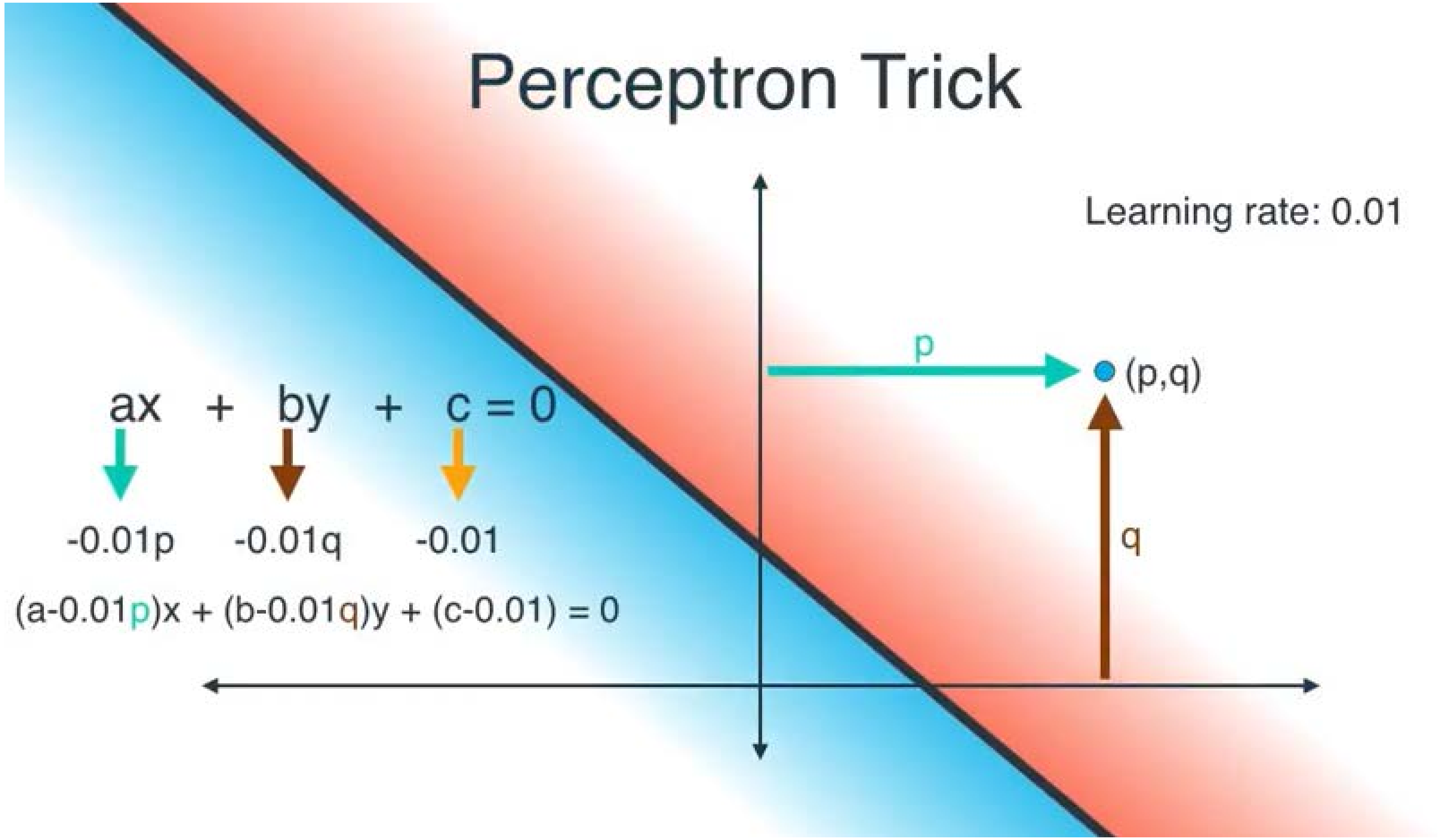
- Pick random point
- If point is correctly classified:
  - Do nothing
- If point is incorrectly classified
  - Move line towards point

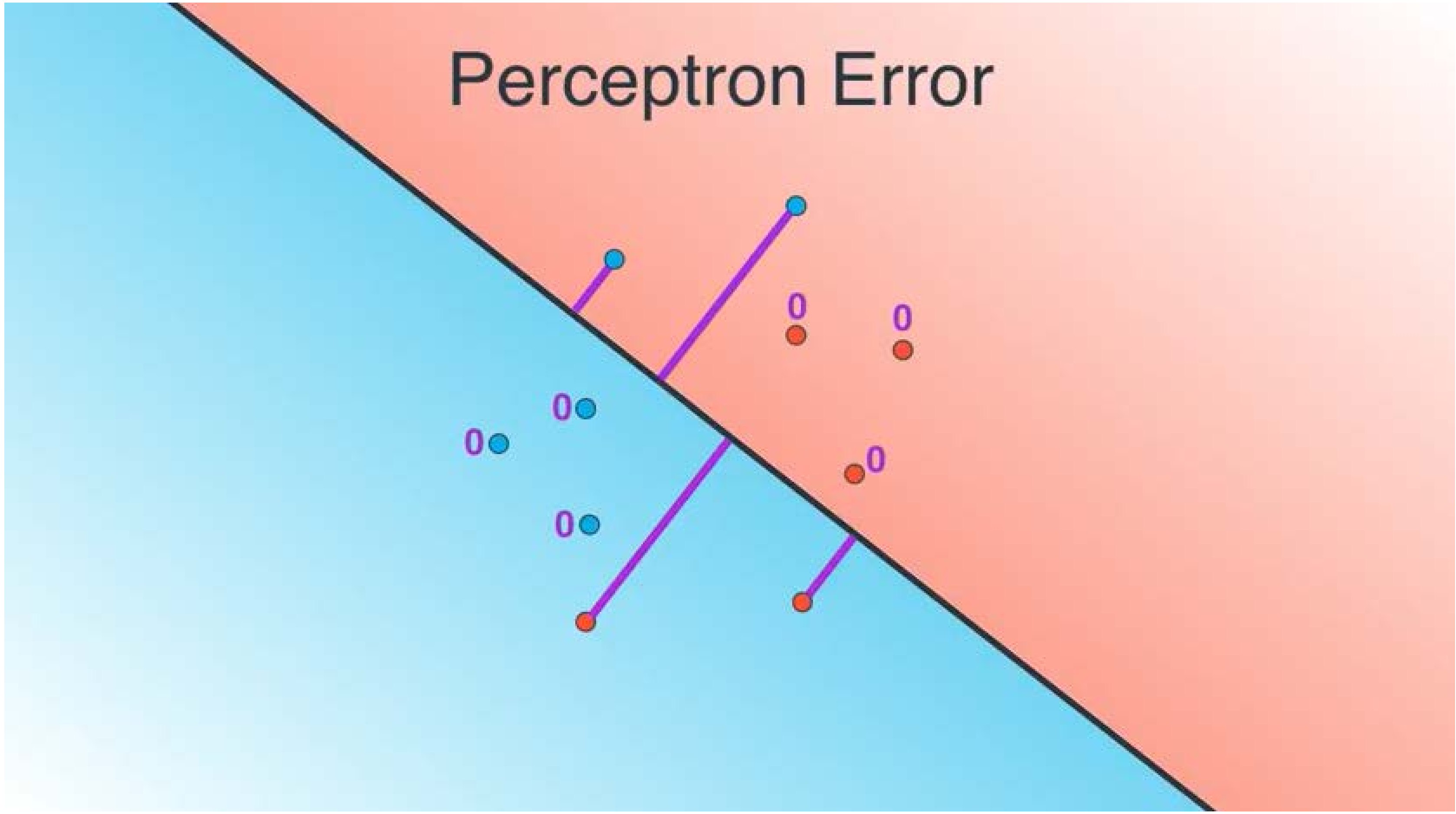
Perceptron Trick





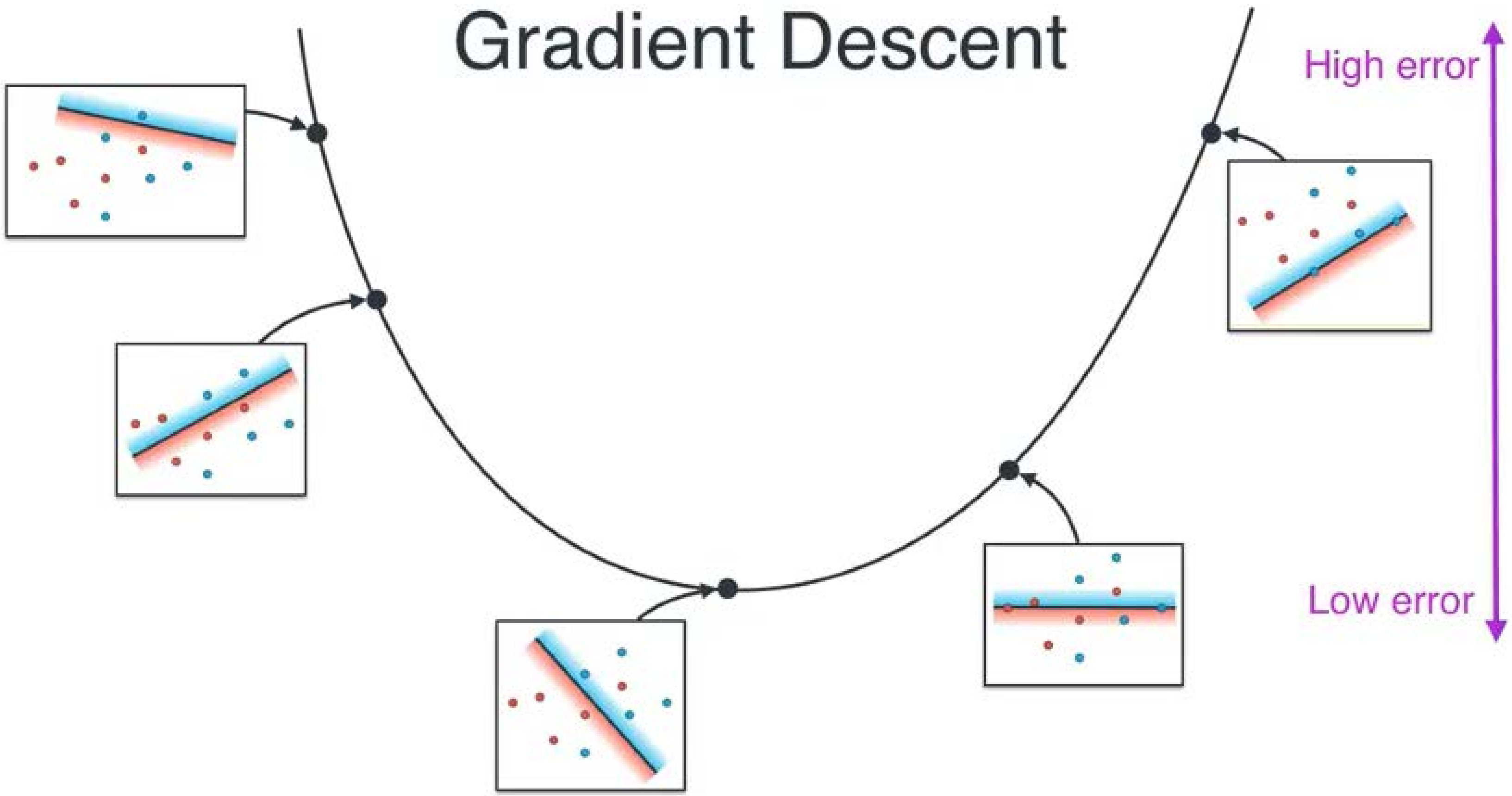
Perceptron Trick
Learning rate: 0.01
p
(p,q)
q
ax + by + c = 0
-0.01p
-0.01q
-0.01
(a-0.01p)x + (b-0.01q)y + (c-0.01) = 0

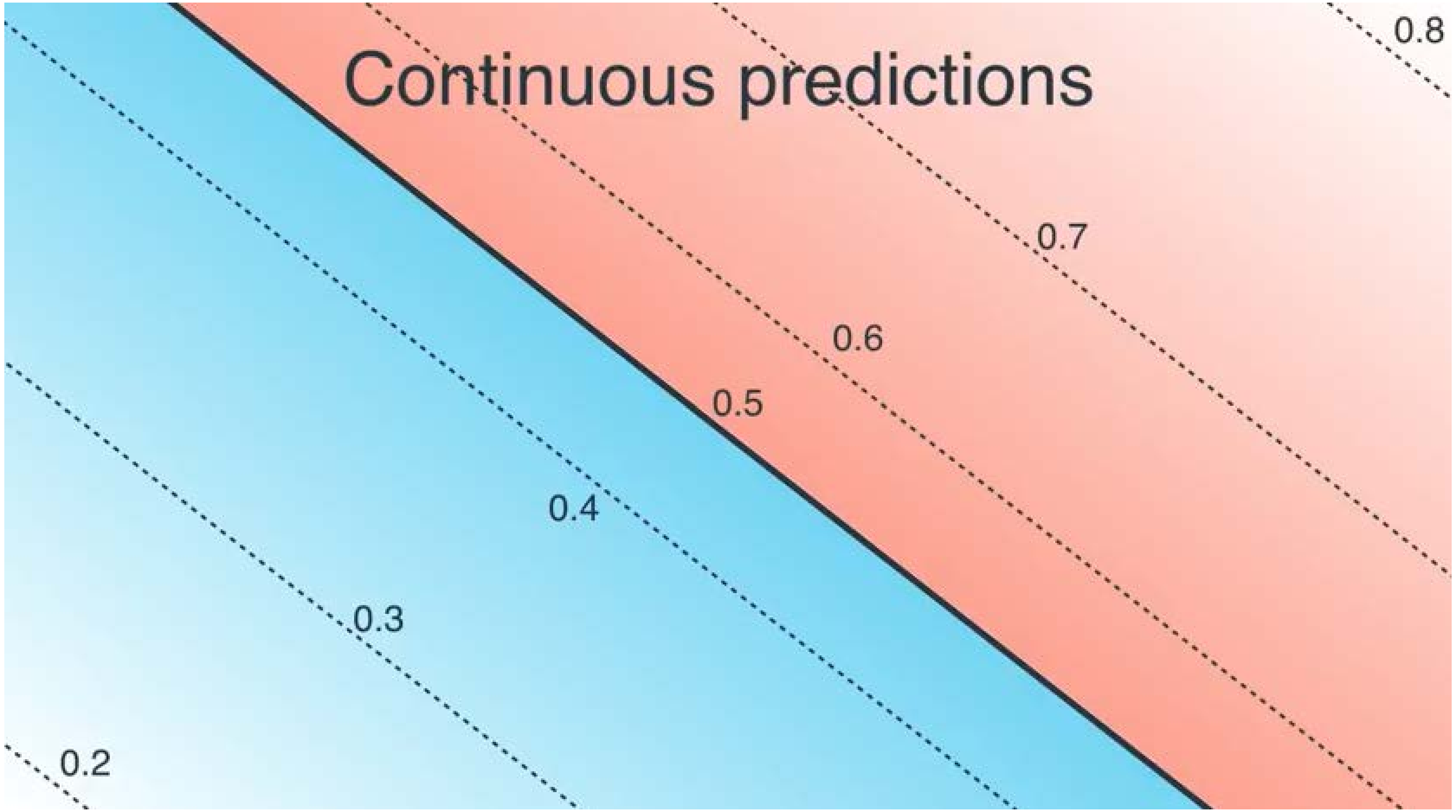

Perceptron Trick
Learning rate: 0.01
p
(p,q)
ax + by + c = 0
-0.01p
-0.01q
-0.01
q
(a-0.01p)x + (b-0.01q)y + (c-0.01) = 0

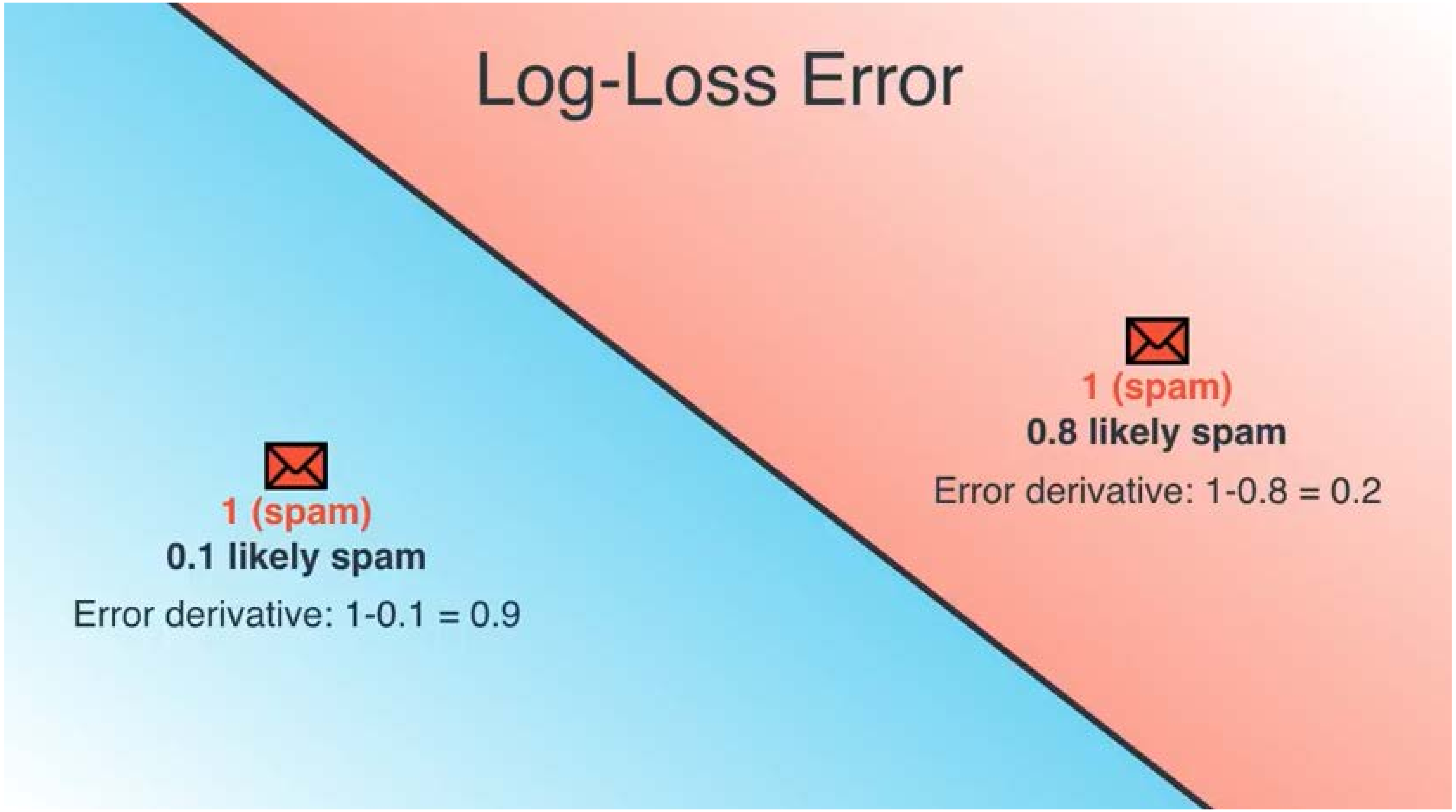


Perceptron Error
0
0
0
0
0
0
0

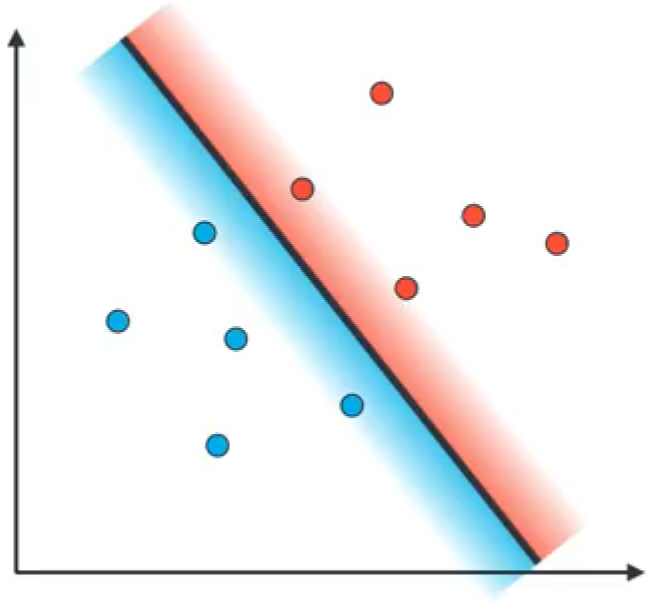
Gradient Descent




Continuous predictions
0.8
0.7
0.6
0.5
0.4
0.3
0.2


Log-Loss Error
1 (spam)
0.8 likely spam
Error derivative: 1-0.8 = 0.2
1 (spam)
0.1 likely spam
Error derivative: 1-0.1 = 0.9

Logistic regression algorithm



Step 1: Start with a random line of equation $ax + by + c = 0$

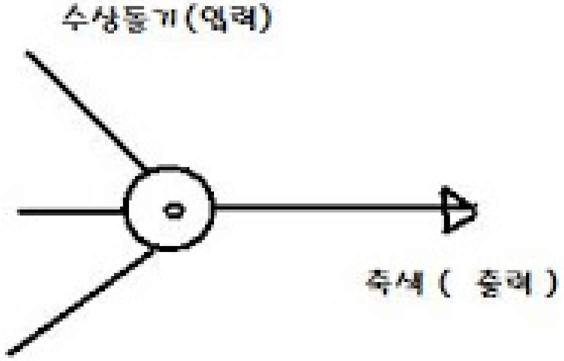
Step 2: Pick a large number. 1000 (number of repetitions, or epochs)

Step 3: Pick a small number. 0.01 (learning rate)

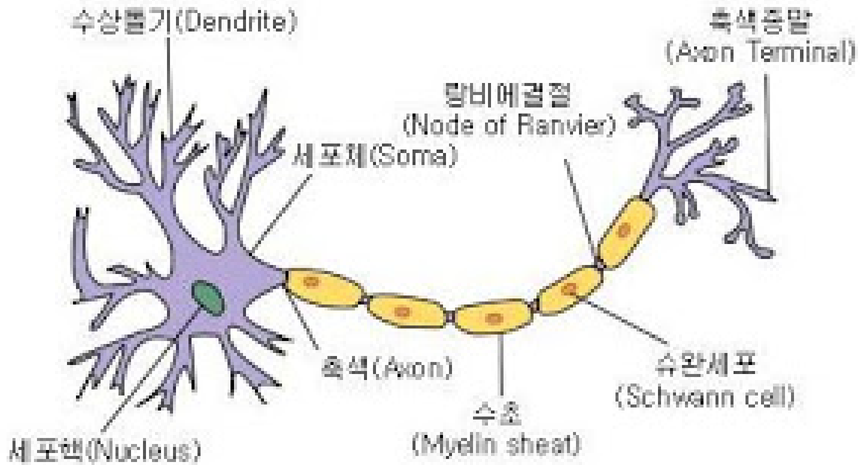
Step 4: (repeat 1000 times)

- Pick random point (p,q)
  - Add $0.01(y - \hat{y})p$ to a
  - Add $0.01(y - \hat{y})q$ to b
  - Add $0.01(y - \hat{y})$ to c

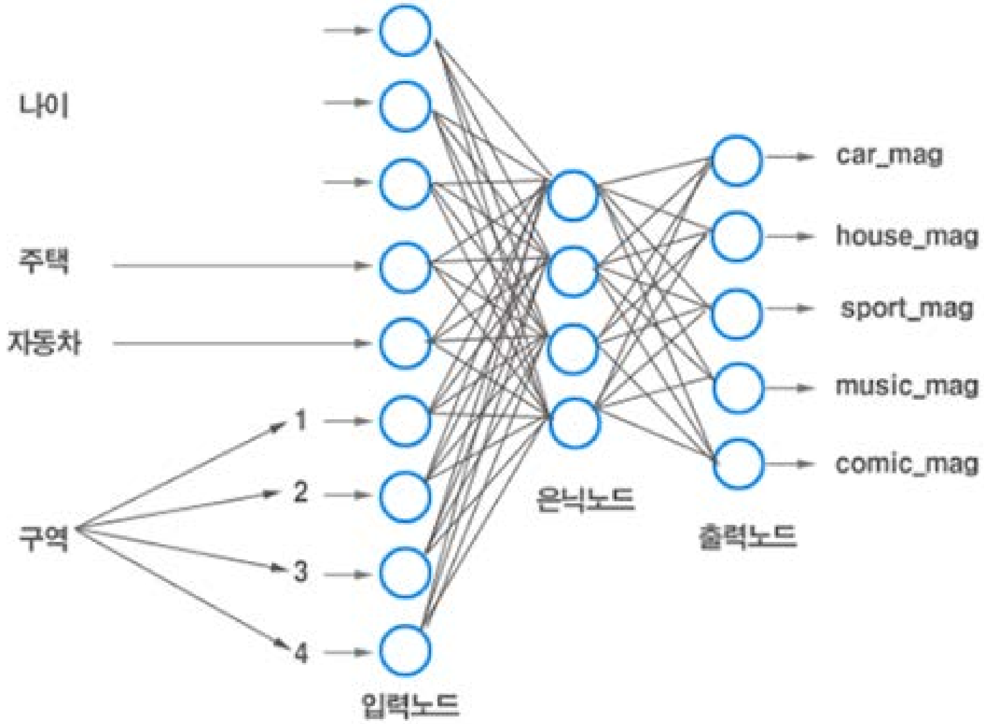
Step 5: Enjoy your fitted line!



수상돌기(입력)
축색(출력)



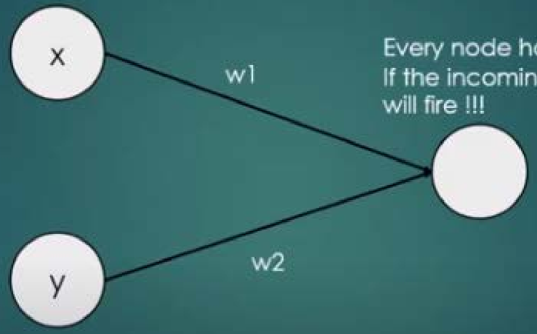
수상돌기(Dendrite)
축색종말
(Axon Terminal)
랑비에결절
(Node of Ranvier)
세포체(Soma)
축색(Axon)
수초
(Myelin sheat)
슈완세포
(Schwann cell)
세포핵(Nucleus)



나이
주택
자동차
구역
1
2
3
4
입력노드
은닉노드
출력노드
car_mag
house_mag
sport_mag
music_mag
comic_mag

AND logical relation

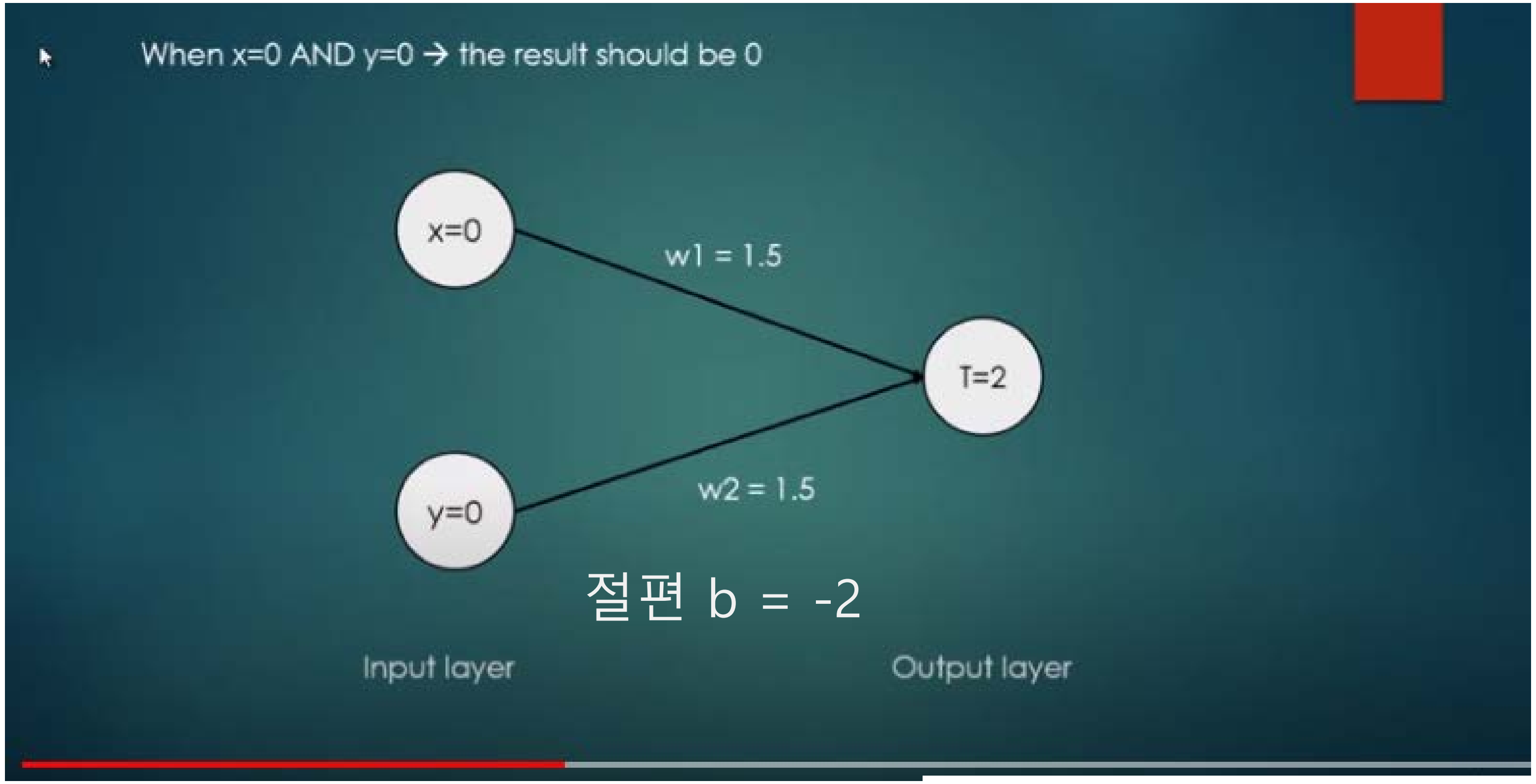
x	y	x AND y
0	0	0
0	1	0
1	0	0
1	1	1

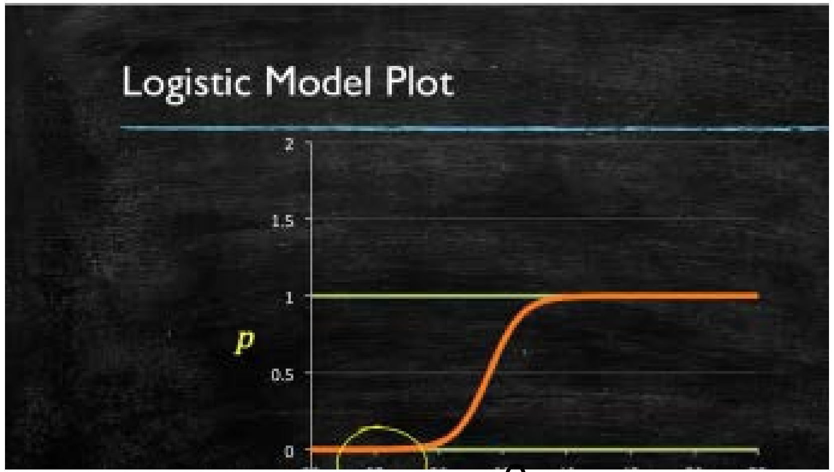


Every node has a threshold T
If the incoming signal > T -> the neuron will fire !!!

Input layer

Output layer

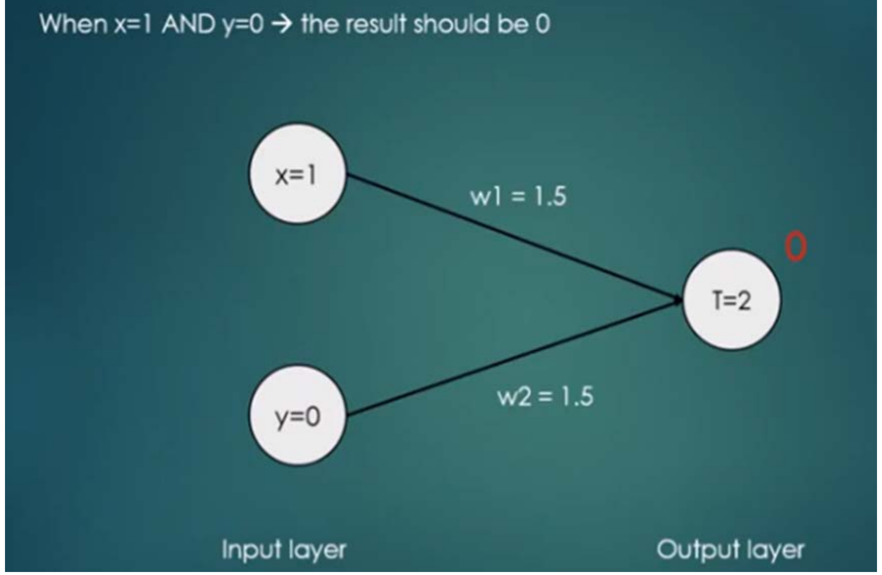




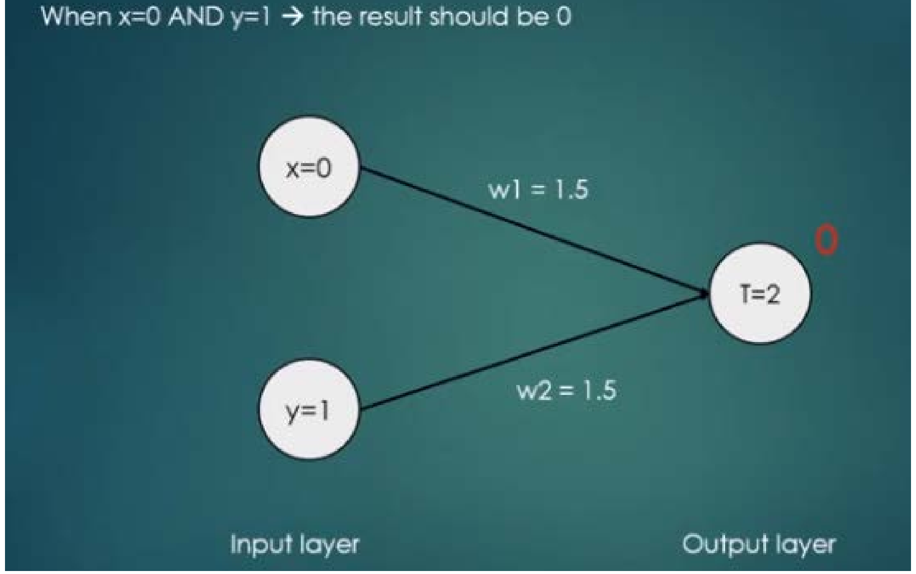
0

Input signal = 0*1.5 + 0*1.5 – 2 = -2

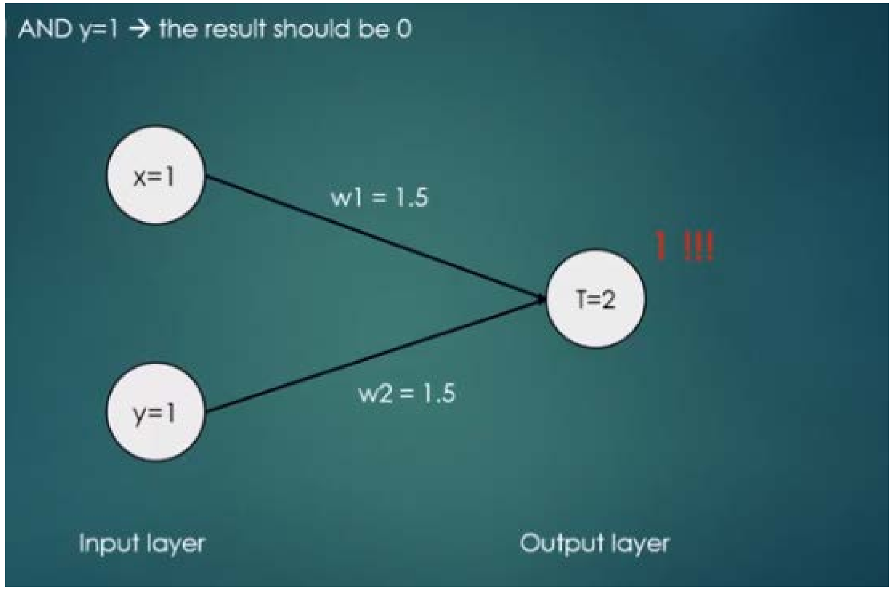
오른쪽 logistic regression에서 -2 = output = 0



1.5*1 + 1.5* 0 – 2 = -0.5 ➔ 0

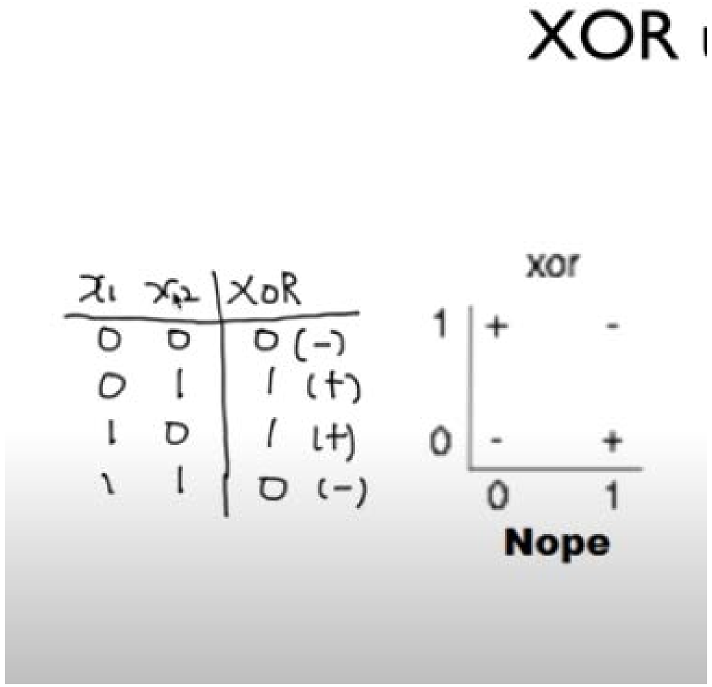


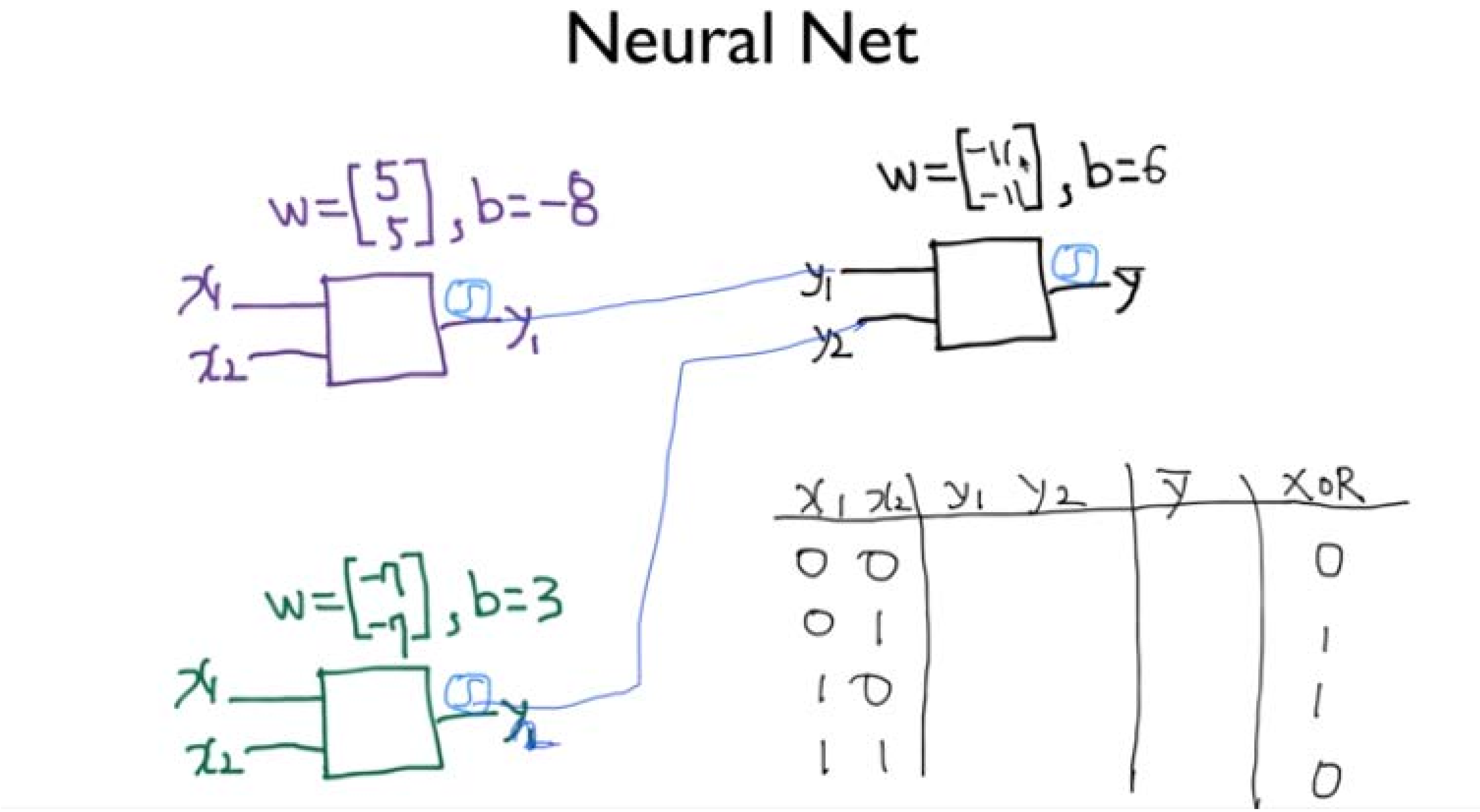
1.5*0 + 1.5* 1 – 2 = -0.5 ➔ 0



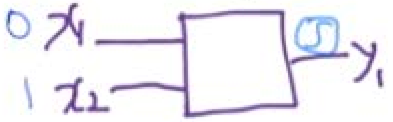
1.5*1 + 1.5* 1 – 2 = 1 ➔ 1

xor problem solver : Multi Layer Perceptron

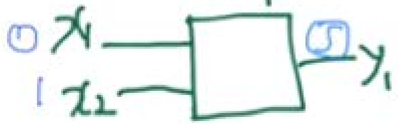




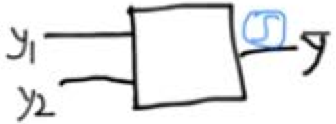
$w=\begin{bmatrix}5\\5\end{bmatrix}, b=-8$



$w=\begin{bmatrix}-7\\-7\end{bmatrix}, b=3$



$w=\begin{bmatrix}-11\\-11\end{bmatrix}, b=6$



$[0\ 0]\begin{bmatrix}5\\5\end{bmatrix}-8=-8, \quad y_1=S(-8)=0 \quad 10^{-10}$

$[0\ 0]\begin{bmatrix}-7\\-7\end{bmatrix}+3=3, \quad y_2=S(3)=1$

$[0\ 1]\begin{bmatrix}-11\\-11\end{bmatrix}+6=-11+6=-5$

$\bar{Y}=S(-5)=0$

$[0\ 1]\begin{bmatrix}5\\5\end{bmatrix}-8=0+5-8=-3, \text{ sigmoid}(-3)=0$

$[1\ 0]\begin{bmatrix}5\\5\end{bmatrix}-8=5+0-8=\underline{-3}, \text{ sigmoid}(-3)=\underline{0}$

$[0\ 1]\begin{bmatrix}-7\\-7\end{bmatrix}+3=0+-7+3=\underline{-4}, \text{ sigmoid}(-4)=0$

$[1\ 0]\begin{bmatrix}-7\\-7\end{bmatrix}+3=-7+0+3=\underline{-4}, \text{ sigmoid}(-4)=0$

$[0\ 0]\begin{bmatrix}-11\\-11\end{bmatrix}+6=0+0+6=$

$\text{sigmoid}(6)=1$

$[0\ 0]\begin{bmatrix}-11\\-11\end{bmatrix}+6=0+0+6=6$

$\text{sigmoid}(6)=1$

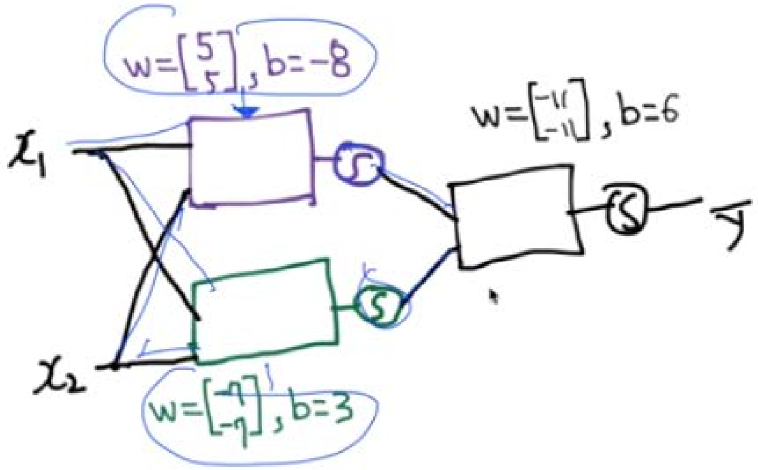
$[1\ 1]\begin{bmatrix}5\\5\end{bmatrix}-8=5+5-8=2, \text{ sigmoid}(2)=1$

$[1\ 1]\begin{bmatrix}-7\\-7\end{bmatrix}+3=-7+-7+3=-11, \text{ sigmoid}(-11)=0$

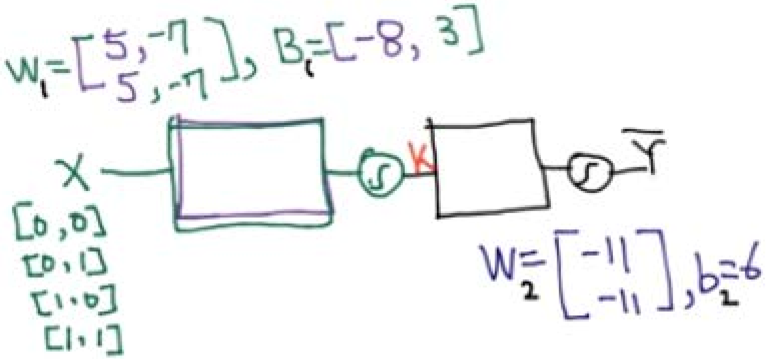
$[1\ 0]\begin{bmatrix}-11\\-11\end{bmatrix}+6=-11+0+6=-5$

$\text{sigmoid}(-5)=0$

x_1	x_2	y_1	y_2	$\bar{y}$	XOR
0	0	0	1	0	0
0	1	0	0	1	1
1	0	0	0	1	1
1	1	1	0	0	0



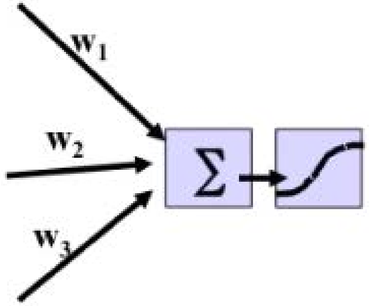
NN

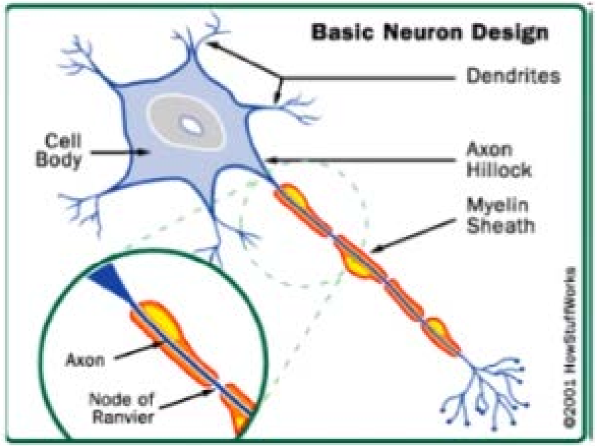


$$K(x) = \mathrm{sigmoid}(XW_1 + B_1)$$

$$\bar{Y} = H(X) = \mathrm{sigmoid}(K(x)W_2 + b_2)$$

```
# NN
K = tf.sigmoid(tf.matmul(X, W1) + b1)
hypothesis = tf.sigmoid(tf.matmul(K, W2) + b2
```





ler “How stuff works: the brain”