인공지능 기반 설계 이론 및 사례 연구

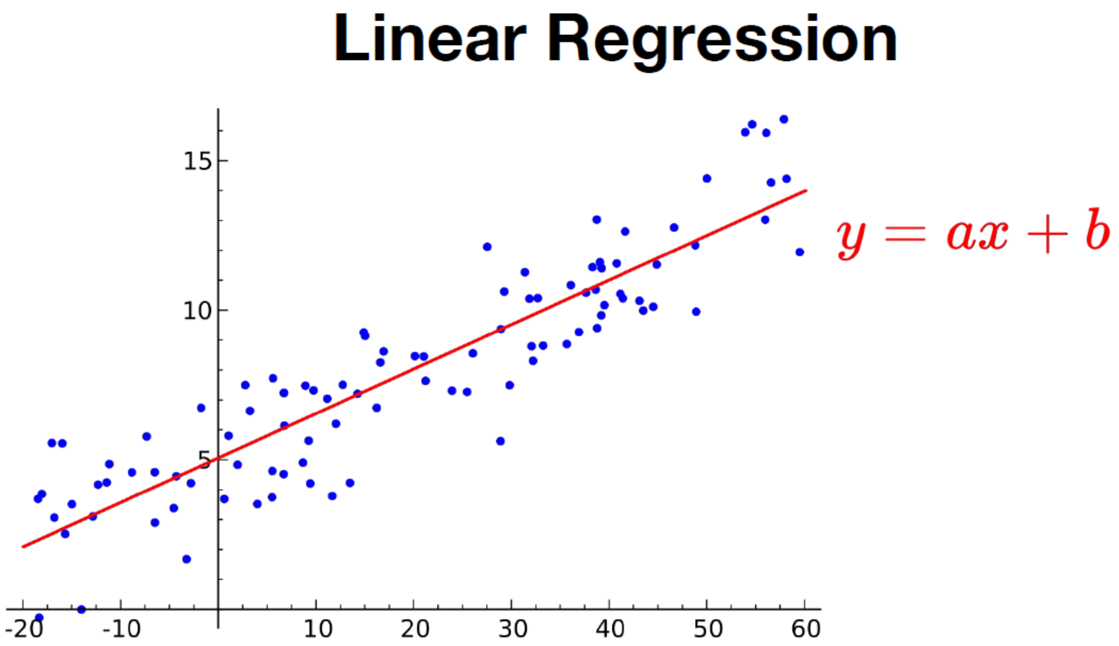
2차) 지도학습의 기초 (Regression)

2020년 9월

강남우

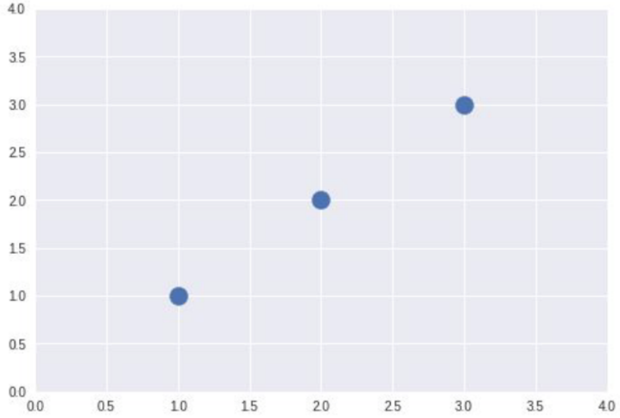
기계시스템학부
숙명여자대학교

Linear Regression



x	y
1	1
2	2
3	3

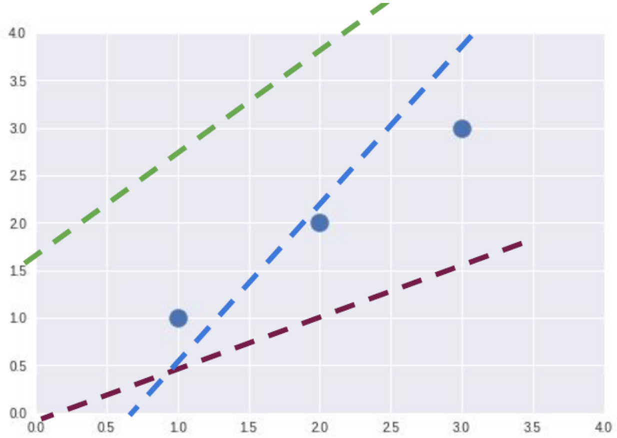
Data



$$H(x) = Wx + b$$

Hypothesis

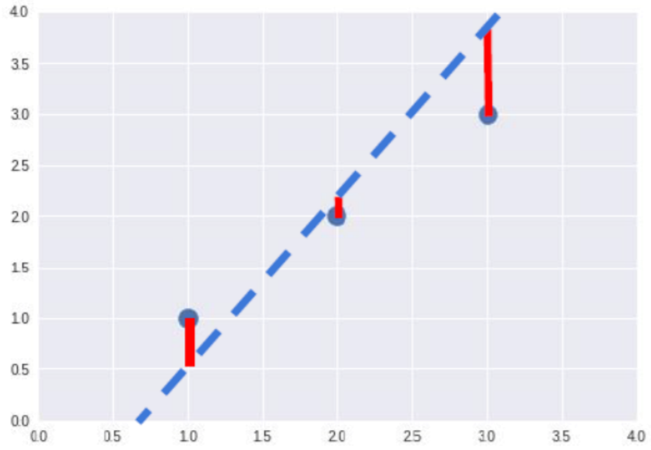
Which hypothesis is better?



Cost Function for Linear Regression

Cost: How fit the line to our (training) data

$$H(x) - y$$



$$\frac{(H(x_1)-y_1)^2+(H(x_2)-y_2)^2+(H(x_3)-y_3)^2}{3}$$

$$cost(W,b) = \frac{1}{m}\sum_{i=1}^{m}(H(x_i) - y_i)^2$$

Cost function

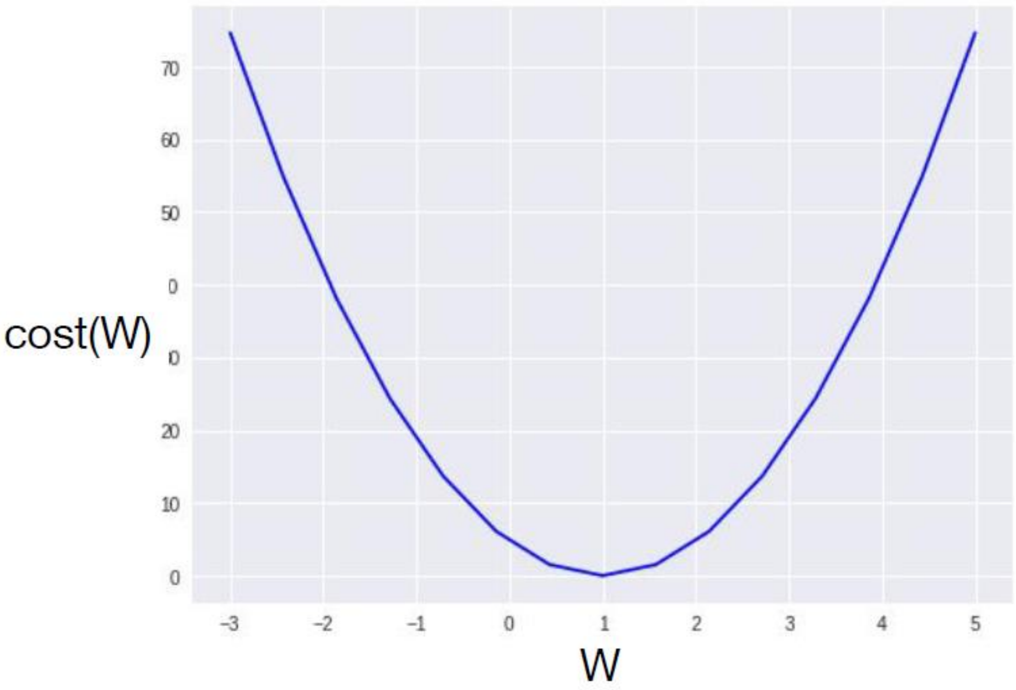
Mean Squared Error (MSE)

Goal: Minimize cost

$$\underset{W,b}{minimize}\; cost(W,b)$$

Cost Function for Linear Regression

What cost looks like?



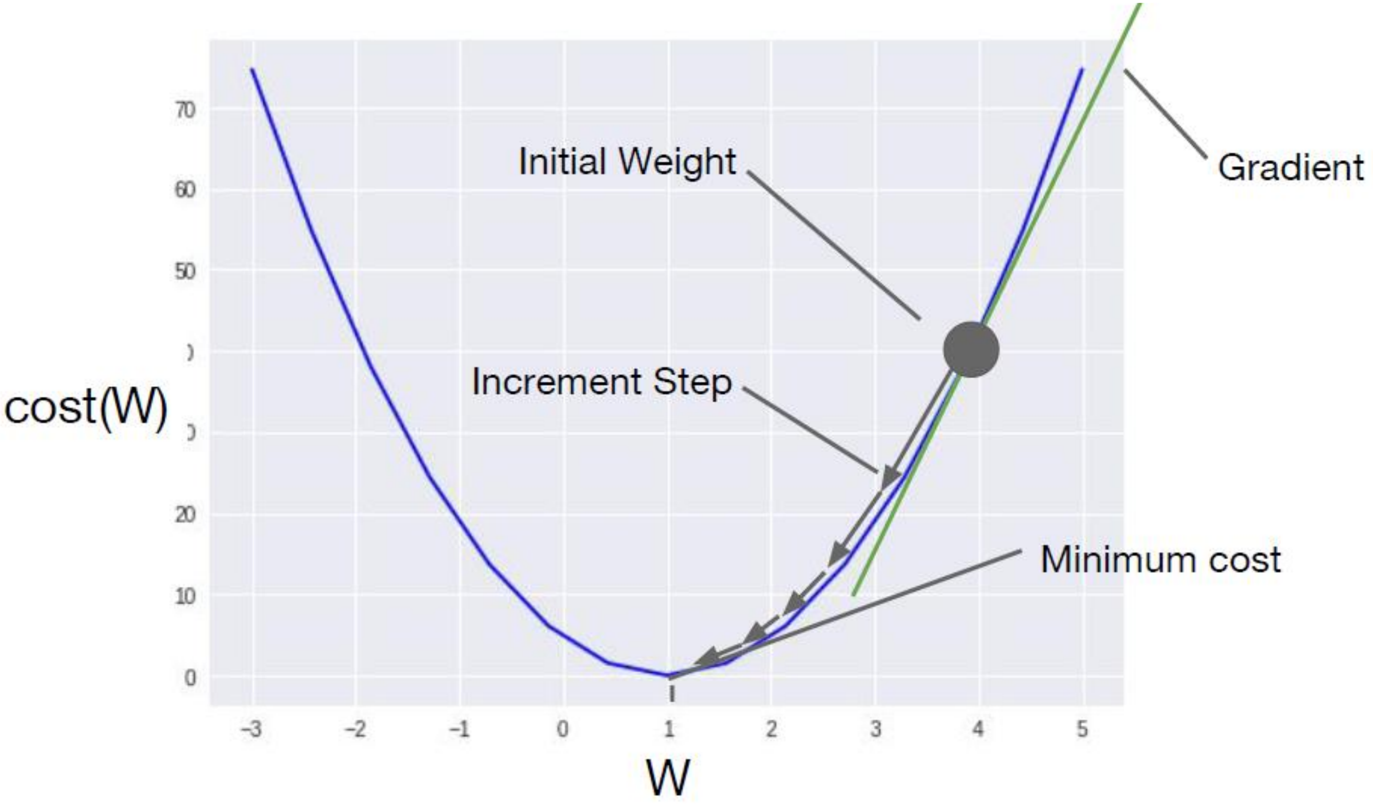
Simplified hypothesis

Hypothesis $H(x) = Wx$

Cost $cost(W) = \frac{1}{m} \sum_{i=1}^{m} (Wx_i - y_i)^2$

Optimization

Gradient descent algorithm



Cost function

$$cost(W) = \frac{1}{m}\sum_{i=1}^{m}(Wx_i - y_i)^2$$

$$cost(W) = \frac{1}{2m}\sum_{i=1}^{m}(Wx_i - y_i)^2$$

Updating W for minimizing cost

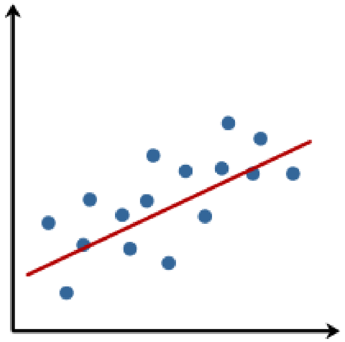
Learning rate

$$W := W - \alpha\frac{\partial}{\partial W}cost(W)$$

$$W := W - \alpha\frac{1}{m}\sum_{i=1}^{m}(W(x_i) - y_i)x_i$$

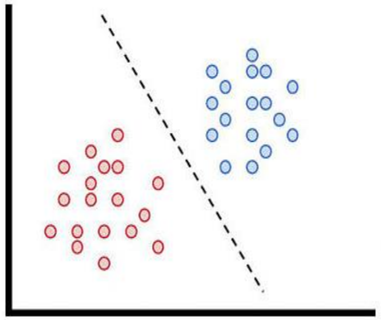
Logistic Regression

Linear



Regression

Logistic



Classification

Continuous

Time / Weight / Height

VS

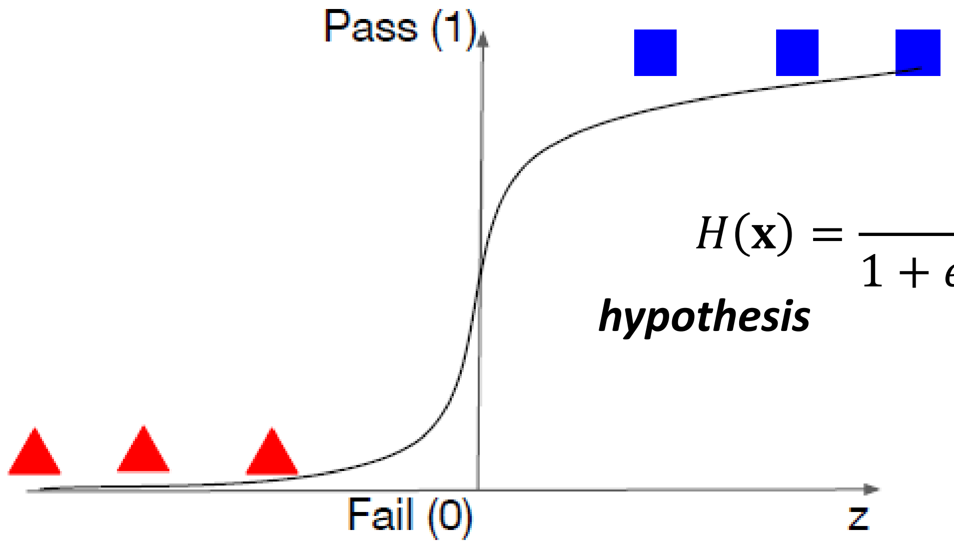
Discrete

What is Binary(Multi-class) Classification?
variable is either 0 or 1 (0:positive / 1:negative)

- Exam : Pass or Fail
- Spam : Not Spam or Spam
- Face : Real or Fake
- Tumor : Not Malignant or Malignant

To start with machine learning, you must encode variable [0,1]

Logistic (Sigmoid) function



$$H(\mathbf{x}) = \frac{1}{1+e^{-\mathbf{W}^{\mathrm{T}}\mathbf{x}}} = \frac{1}{1+e^{-z}}$$

hypothesis

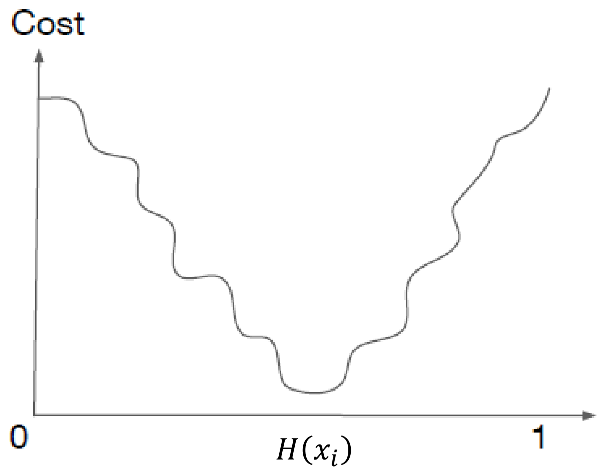
$$0 \le g(z) \le 1$$

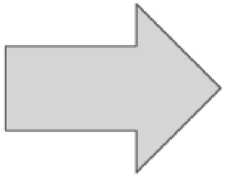
$$\mathbf{x} \longrightarrow \boxed{z = \mathbf{W}^{\mathrm{T}}\mathbf{x}} \longrightarrow \boxed{g(z) = \frac{1}{1+e^{-z}}} \longrightarrow \boxed{> 0.5} \longrightarrow \mathrm{y} \in \{0,1\}$$

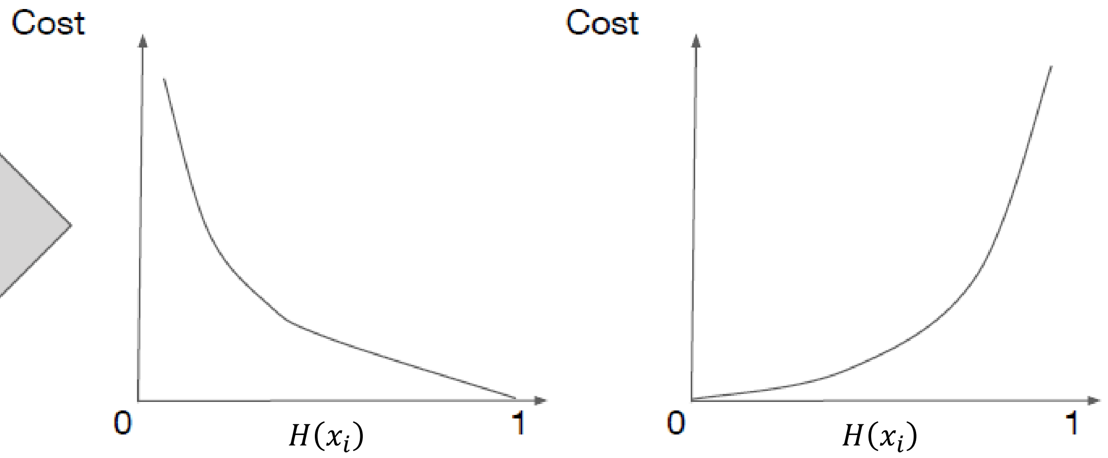
Linear function ***Logistic function (Sigmoid function)*** ***Decision Boundary***

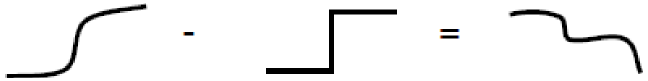
Cost Function for Logistic Regression

A Convex Logistic Regression Cost Function









$$cost(W,b) = \frac{1}{m}\sum_{i=1}^{m}(H(x_i) - y_i)^2$$

Mean Squared Error (MSE)

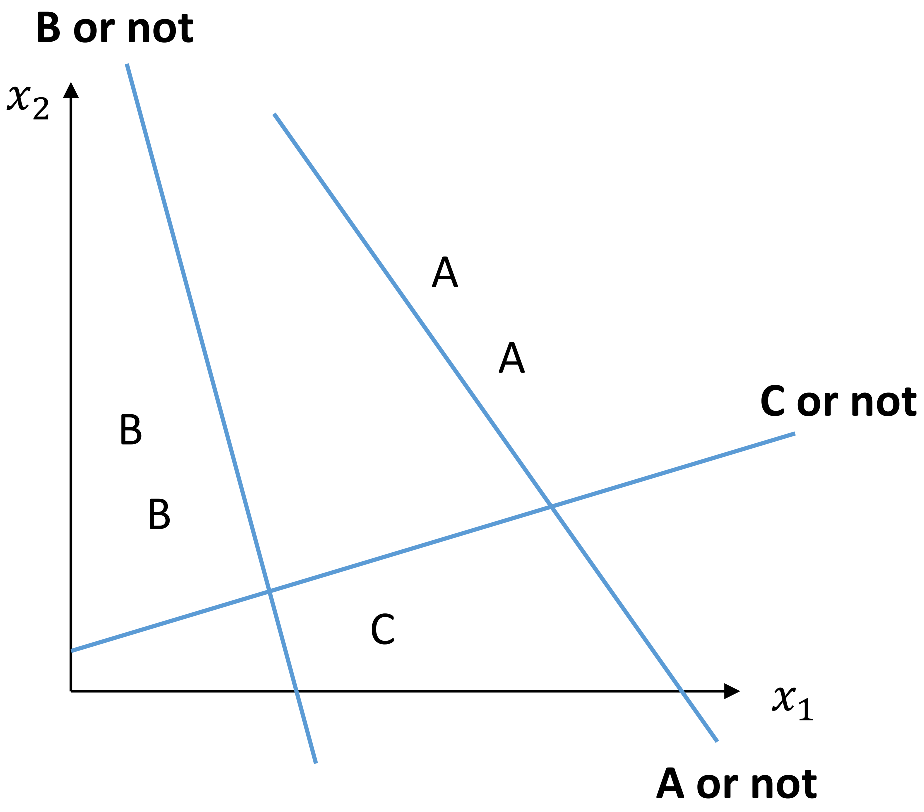
$$\begin{array}{ll} -\log(H(x_i)) & if\ y_i = 1 \\ -\log(1 - H(x_i)) & if\ y_i = 0 \end{array}$$

$$cost(W,b) = \frac{1}{m}\sum_{i=1}^{m}[-y_i \log\big(H(x_i)\big) - (1 - y_i)\log(1 - H(x_i))]$$

Cross-entropy

Multinomial Logistic Regression

Multinomial Classification



sigmoid

$[w_{A1}\ w_{A2}]\begin{bmatrix}x_1\\x_2\end{bmatrix} = [w_{A1}x_1 + w_{A2}x_2]$ ➡ sigmoid ➡ **A or not**

$[w_{B1}\ w_{B2}]\begin{bmatrix}x_1\\x_2\end{bmatrix} = [w_{B1}x_1 + w_{B2}x_2]$ ➡ sigmoid ➡ **B or not**

$[w_{C1}\ w_{C2}]\begin{bmatrix}x_1\\x_2\end{bmatrix} = [w_{C1}x_1 + w_{C2}x_2]$ ➡ sigmoid ➡ **C or not**

⬇

$$\begin{bmatrix}w_{A1} & w_{A2}\\ w_{B1} & w_{B2}\\ w_{C1} & w_{C2}\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix} = \begin{bmatrix}w_{A1}x_1 + w_{A2}x_2\\ w_{B1}x_1 + w_{B2}x_2\\ w_{C1}x_1 + w_{C2}x_2\end{bmatrix}$$

Multinomial Logistic Regression

$$S = f(x_i; W)$$

$$\begin{bmatrix} w_{A1} & w_{A2} \\ w_{B1} & w_{B2} \\ w_{C1} & w_{C2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} w_{A1}x_1 + w_{A2}x_2 \\ w_{B1}x_1 + w_{B2}x_2 \\ w_{C1}x_1 + w_{C2}x_2 \end{bmatrix} = \overset{S}{\begin{bmatrix} 2.0 \\ 1.0 \\ 0.1 \end{bmatrix}}$$

$$\boxed{P(Y = k | X = x_1) = \frac{e^{s_k}}{\sum_j e^{s_j}}}$$

Softmax

One-hot encoding

$$\Rightarrow \begin{bmatrix} 0.7 \\ 0.2 \\ 0.1 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \begin{matrix} \textbf{A} \\ \text{B} \\ \text{C} \end{matrix}$$

Probabilities

$$\frac{e^2}{e^2 + e^1 + e^{0.1}} = 0.7$$

Cost Function for Multinomial Logistic Regression

Cross entropy for multi-class

$$cost = H(p,q) = -\sum_i p_i \log(q_i)$$

Prediction

\# of classes True

Cross entropy for binary class

$$where\ p \in \{y, 1-y\}\ and\ q \in \{\hat{y}, 1-\hat{y}\}$$

$$cost = H(p,q) = -\sum_i p_i \log(q_i) = -y_i \log(\hat{y}_i) - (1-y_i)\log(1-\hat{y}_i)$$

Regression Metrics

RMSE (Root Mean Squared Error)

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

- 예측하려는 값의 크기에 의존적임

MAPE (Mean Absolute Percentage Error)

$$\mathrm{M} = \frac{100}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right|$$

- 예측하려는 값의 크기에 의존적이지 않음
  - 예측하려는 값이 1이상이어야 함

MSE (Mean Squared Error)

MAE (Mean Absolute Error)

Confusion Matrix for Classification

Confusion Matrix

n=165	Predicted: Negative	Predicted: Positive	
Actual: Negative	TN = 50	FP = 10	60
Actual: Positive	FN = 5	TP = 100	105
	55	110	

- **true positives (TP):** These are cases in which we predicted yes (they have the disease), and they do have the disease.
- **true negatives (TN):** We predicted no, and they don't have the disease.
- **false positives (FP):** We predicted yes, but they don't actually have the disease. (Also known as a "Type I error.")
- **false negatives (FN):** We predicted no, but they actually do have the disease. (Also known as a "Type II error.")

Confusion Matrix for Classification

Confusion Matrix

n=165	Predicted: Negative	Predicted: Positive	
Actual: Negative	TN = 50	FP = 10	60
Actual: Positive	FN = 5	TP = 100	105
	55	110	

성능지표

- **Accuracy** (실제 이상/정상인지 맞게 예측한 비율)

  = (TP+TN)/(TP+FN+FP+TN) = 90.9%
- **Precision** (이상으로 예측한 것중에 실제 이상인 샘플의 비율)

  = TP/(TP+FP) = 90.9%
- **Recall** (실제 이상 샘플중에 이상으로 예측한 비율)

  = TP/(TP+FN) = 95.20%

What Questions Do You Have?

nwkang@sm.ac.kr

www.smartdesignlab.org