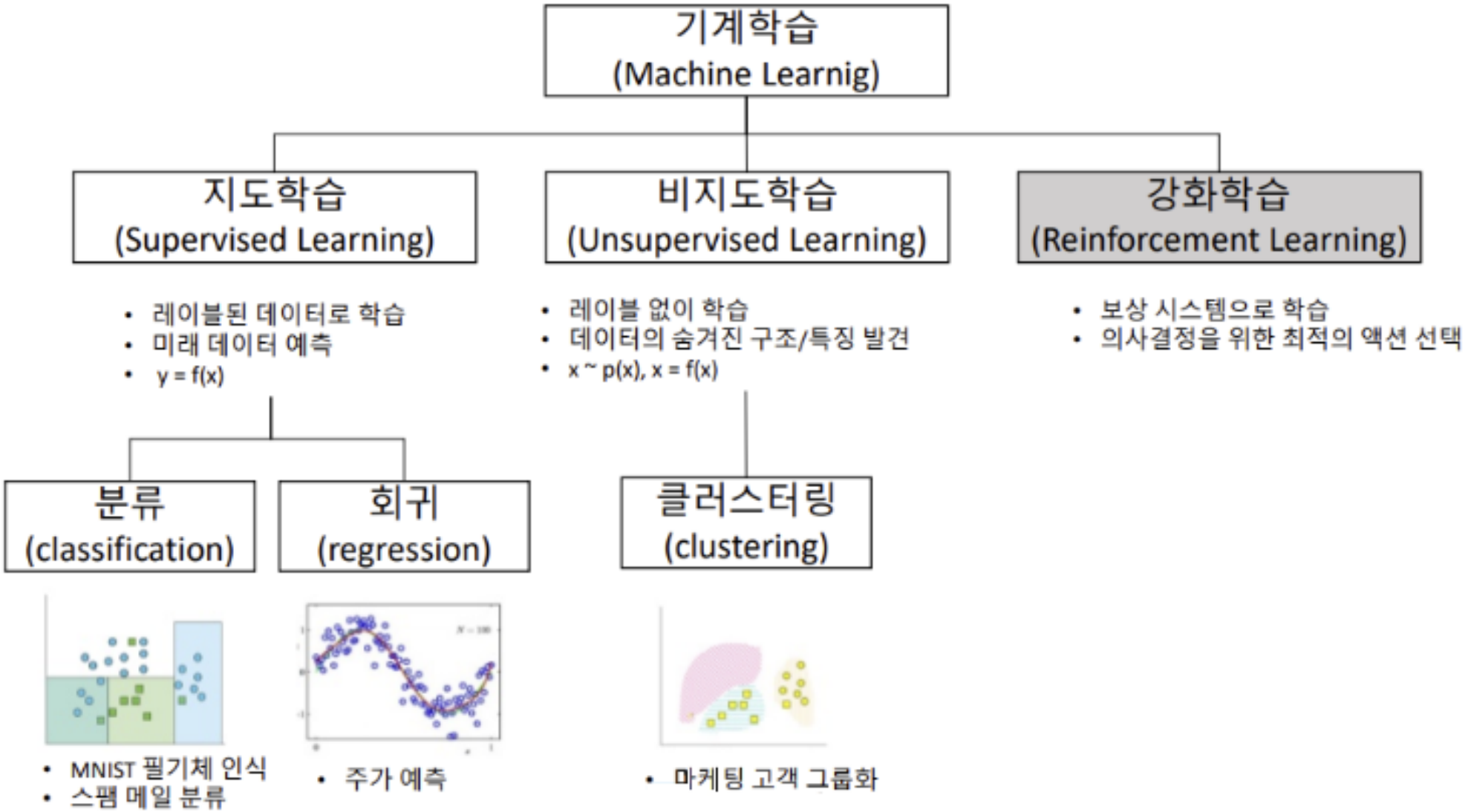
Regression

Stella Sangyoon Bae

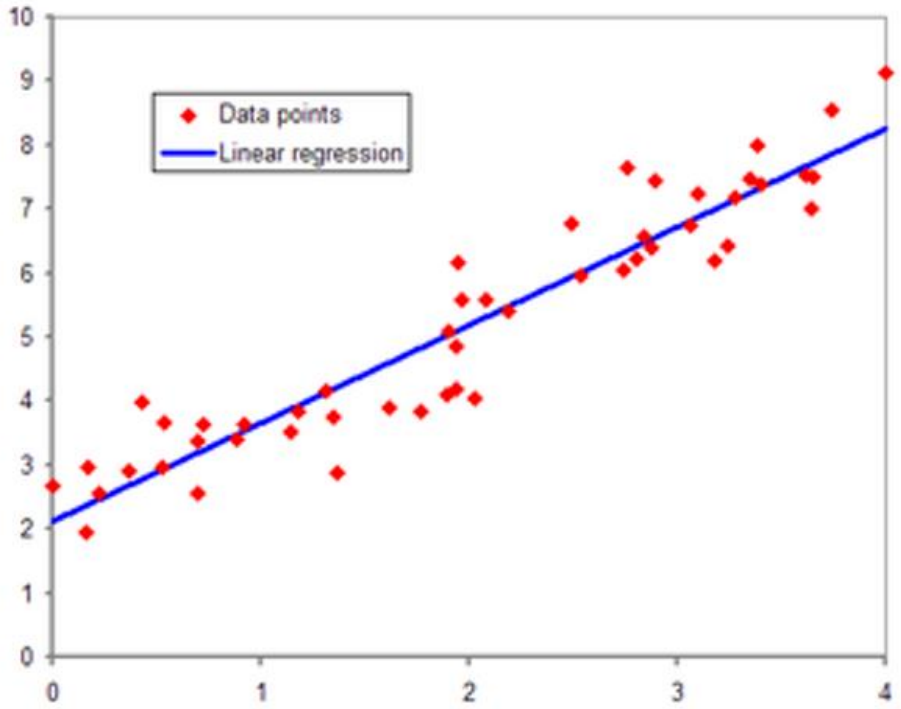
18th June 2023

1. 머신러닝 시스템의 분류



01 Simple linear regression

1-1 What is regression?



시험 공부하는 시간을 늘리면 늘릴 수록 성적이 잘 나옵니다. 이는 수학적으로 생각해보면 어떤 요인의 수치에 따라서 특정 요인의 수치가 영향을 받고 있다고 말할 수 있습니다.

조금 더 수학적인 표현을 써보면 어떤 변수의 값에 따라서 특정 변수의 값이 영향을 받고 있다고 볼 수 있습니다. 다른 변수의 값을 변하게 하는 변수를 x, 변수 x에 의해서 값이 종속적으로 변하는 변수 y라고 해봅시다.

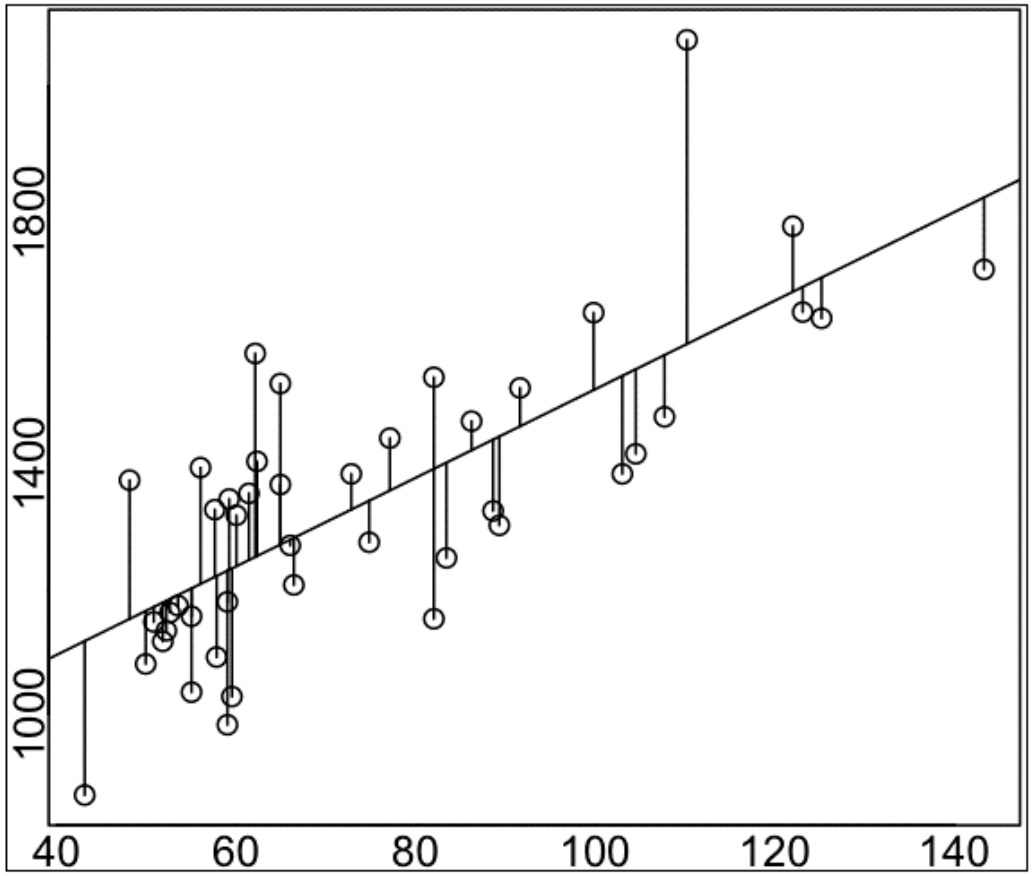
이때 변수 x의 값은 독립적으로 변할 수 있는 것에 반해, x값은 계속해서 y의 값에 의해서, 종속적으로 결정되므로 x를 독립 변수, y를 종속 변수라고도 합니다.

Linear regression은 **한 개 이상의 독립 변수 x와 y의 선형 관계**를 모델링합니다.
우리는 이 챕터에서 독립 변수 x가 1개인 simple linear regression에 대해 배워볼 겁니다.

$$y = wx + b$$

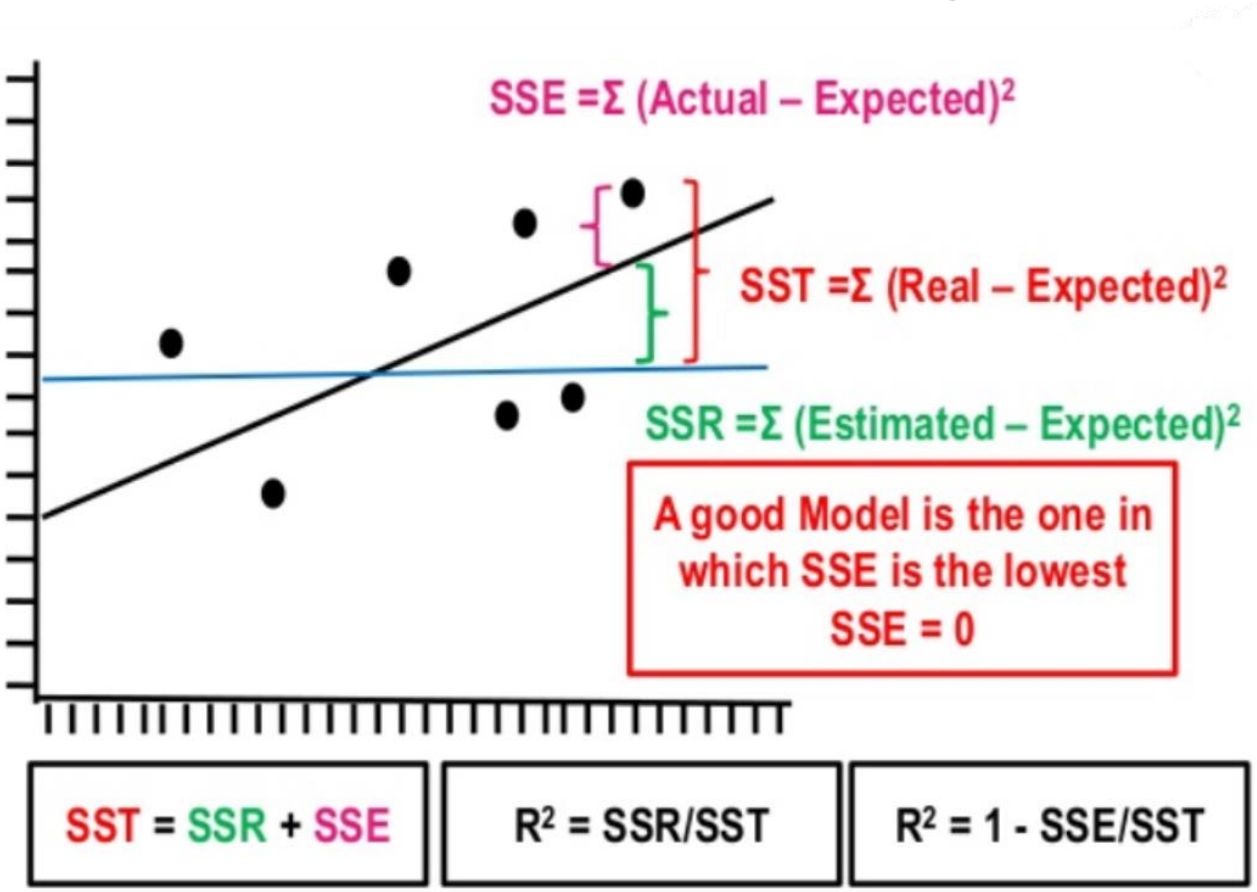
위는 simple linear regression의 식입니다.

1-2 Principles of Linear Regression



옆의 그림처럼 직선과 data point 간 거리인 residual error를 최소화하는 방향으로 직선이 만들어지게 됩니다.

1-3 Statistical analysis of regression



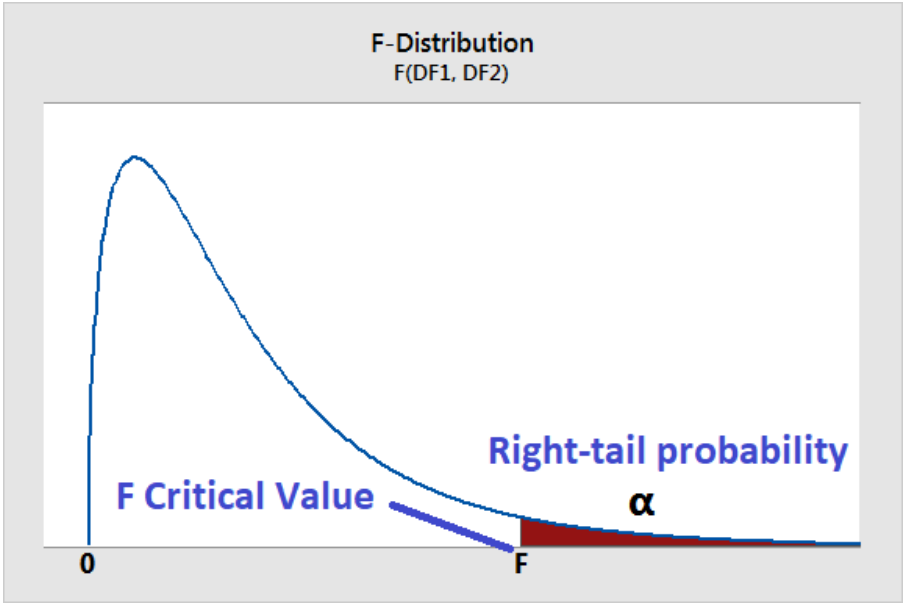
SST (전체 variance) = SSR (regression line에 의해 설명되는 variance) + SSE (regression line에 의해 설명되지 않는 variance)

1-3 Statistical analysis of regression

Variance analysis!

H0 : β = 0 (regression line이 유의하지 않다.)

요인	자유도	제곱합	평균제곱	F-통계량
회귀	1	SSR	MSR=SSR/1	F0=MSR/MSE
잔차	n-2	SSE	MSE=SSE/(n-2)	
총합	n-1	SST		



F-statistics 값이 클수록, 즉 p-value가 작을 수록

regression line이 전체 데이터를 더 잘 설명한다는 뜻입니다.

H0을 reject 하게 되죠.

1-3 Statistical analysis of regression

Variance analysis!

H0 : β = 0 (regression line이 유의하지 않다.)

요인	자유도	제곱합	평균제곱	F-통계량
회귀	1	SSR	MSR=SSR/1	F0=MSR/MSE
잔차	n-2	SSE	MSE=SSE/(n-2)	
총합	n-1	SST		

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

“Coefficient of determination” 이라고 부릅니다.

1에 가까울 수록 regression line이 유의하고,

0에 가까울 수록 regression line이 유의하지 않습니다.

1-3 Statistical analysis of regression

[회귀분석 결과 해석 방법 - SDEM]

유의성 검증 (Significance)

- 내가 설정한 독립변수가 유의하게 나왔나?
- 계수와 변수의 P-value를 통해 확인

방향성 확인 (Direction)

- 계수의 방향성(+ -)이 나의 가설과 맞는지 확인

효과의 크기 (Effect Size)

- X가 Y에 얼마나 큰 영향력을 주는지 확인하는 과정
- 유의해도 너무 작으면 의미가 없을 수도 있음

모델 적합성 (Model Fitting)

- 모델이 너무 주어진 데이터에만 매몰되어 있지 않나(Overfitting)
- 모델이 주어진 데이터를 적게 설명하는 것아닌가 (Underfitting)

실제로 파이썬 코드를 실행하면서 결과를 해석해봅시다!

https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/data

데이터는 여기에 있습니다.

여기서 train.csv를 다운로드 받아서 불러오도록 합시다.

1-3 Statistical analysis of regression

	coef	std err	t	P>|t|	[0.025	0.975]
const	-2.53e+06	1.16e+05	-21.858	0.000	-2.76e+06	-2.3e+06
YearBuilt	1375.3735	58.717	23.424	0.000	1260.194	1490.553

Omnibus:	761.903	Durbin-Watson:	1.984
Prob(Omnibus):	0.000	Jarque-Bera (JB):	6856.947
Skew:	2.264	Prob(JB):	0.00
Kurtosis:	12.603	Cond. No.	1.29e+05

OLS Regression Results

Dep. Variable:	SalePrice	R-squared:	0.273
Model:	OLS	Adj. R-squared:	0.273
Method:	Least Squares	F-statistic:	548.7
Date:	Thu, 08 Jun 2023	Prob (F-statistic):	2.99e-103
Time:	22:14:38	Log-Likelihood:	-18311.
No. Observations:	1460	AIC:	3.663e+04
Df Residuals:	1458	BIC:	3.664e+04
Df Model:	1		
Covariance Type:	nonrobust		

02 Multiple linear regression

2-1 Multiple linear regression?

집의 매매 가격은 단순히 집의 평수가 크다고 결정되는 게 아니라 집의 층의 수, 방의 개수, 지하철 역과의 거리와도 영향이 있습니다. 이러한 다수의 요소를 가지고 집의 매매 가격을 예측해보고 싶습니다.

y는 여전히 1개이지만 이제 x는 1개가 아니라 여러 개가 되었습니다.

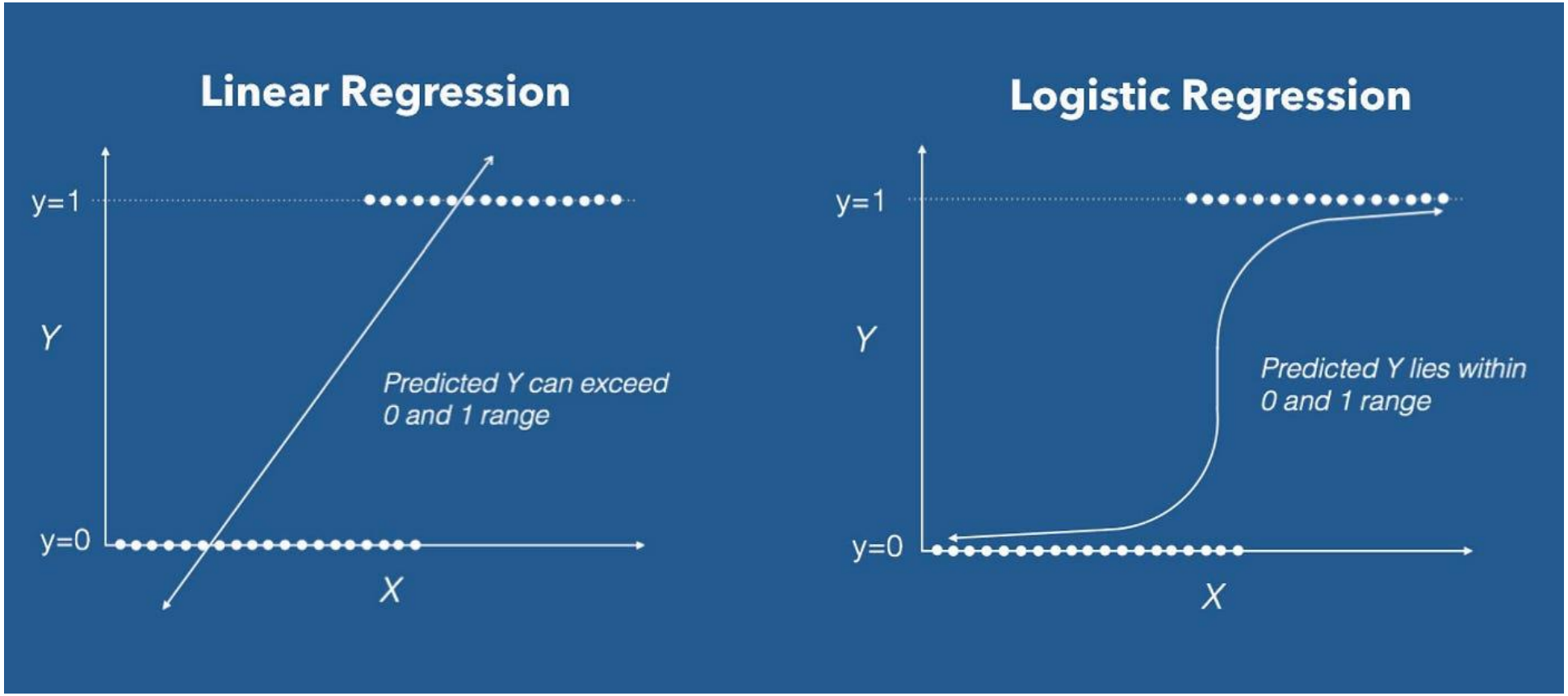
$$y = w_1x_1 + w_2x_2 + \dots w_nx_n + b$$

위는 multiple linear regression의 식입니다.

변수를 조금만 추가하면 됩니다!
실습으로 확인해볼까요?

03 Logistic regression

3-1 Logistic regression?



두 그룹을 분리시키는 선이 직선이 아니라 '곡선' 인 모델입니다. Scikit-learn으로 쉽게 구현할 수 있습니다.

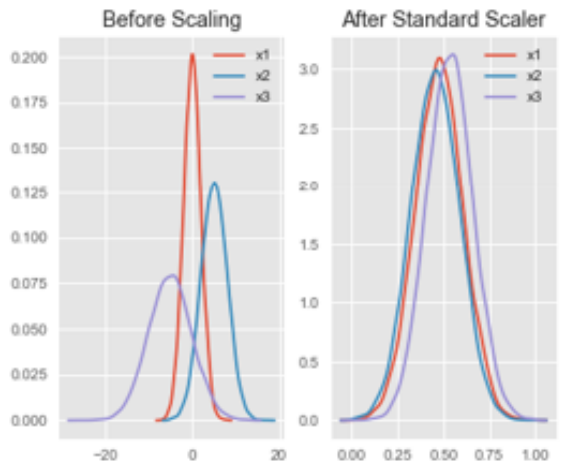
그런데 scikit-learn으로 구현하기 전에 우리는 'normalization'을 포함한 'preprocessing'을 해야 합니다.

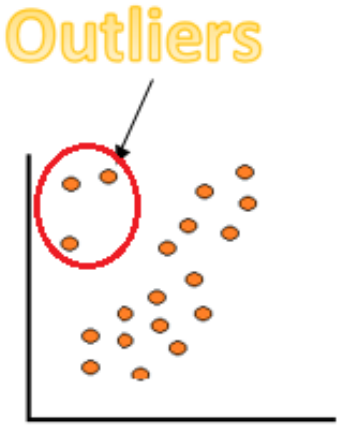
Data Preprocessing

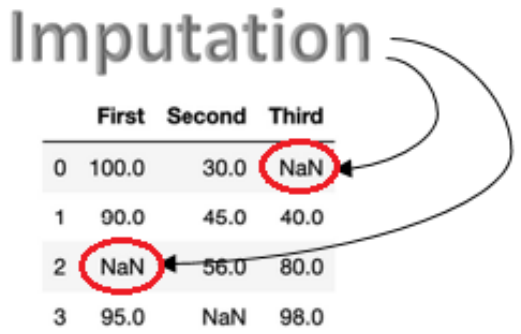
Stella Sangyoon Bae

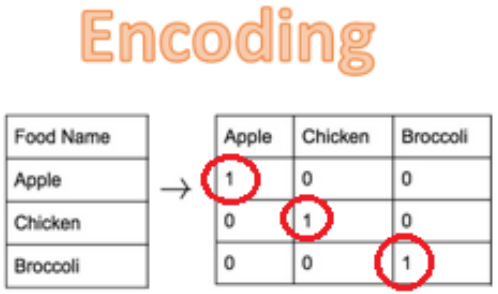
18th June 2023

01 Overview









이거 말고도 더 있어요!

Data Preprocessing

머신러닝 모델의 성능과 정확도를 높이기 위한 step입니다.

이 과정이 선행되지 않으면 모델의 성능이 떨어지거나 아예 실행조차 되지 않을 수 있습니다.

02 Encoding

Name	Species
Coco	Dog
Happy	Cat

하나의 category에
하나의 정수를 할당해주는
encoding

Name	Species
Coco	0
Happy	1

id	color
1	red
2	blue
3	green
4	blue

One Hot Encoding

id	color_red	color_blue	color_green
1	1	0	0
2	0	1	0
3	0	0	1
4	0	1	0

03 Imputation

여러 가지 이유들로 (처음부터 없었음, 소실됨 등)
사라진 데이터 값을 채워 넣는 과정입니다.

1. mean, median 값으로 채워 넣기

	col1	col2	col3	col4	col5
0	2	5.0	3.0	6	NaN
1	9	NaN	9.0	0	7.0
2	19	17.0	NaN	9	NaN

mean()

	col1	col2	col3	col4	col5
0	2.0	5.0	3.0	6.0	7.0
1	9.0	11.0	9.0	0.0	7.0
2	19.0	17.0	6.0	9.0	7.0

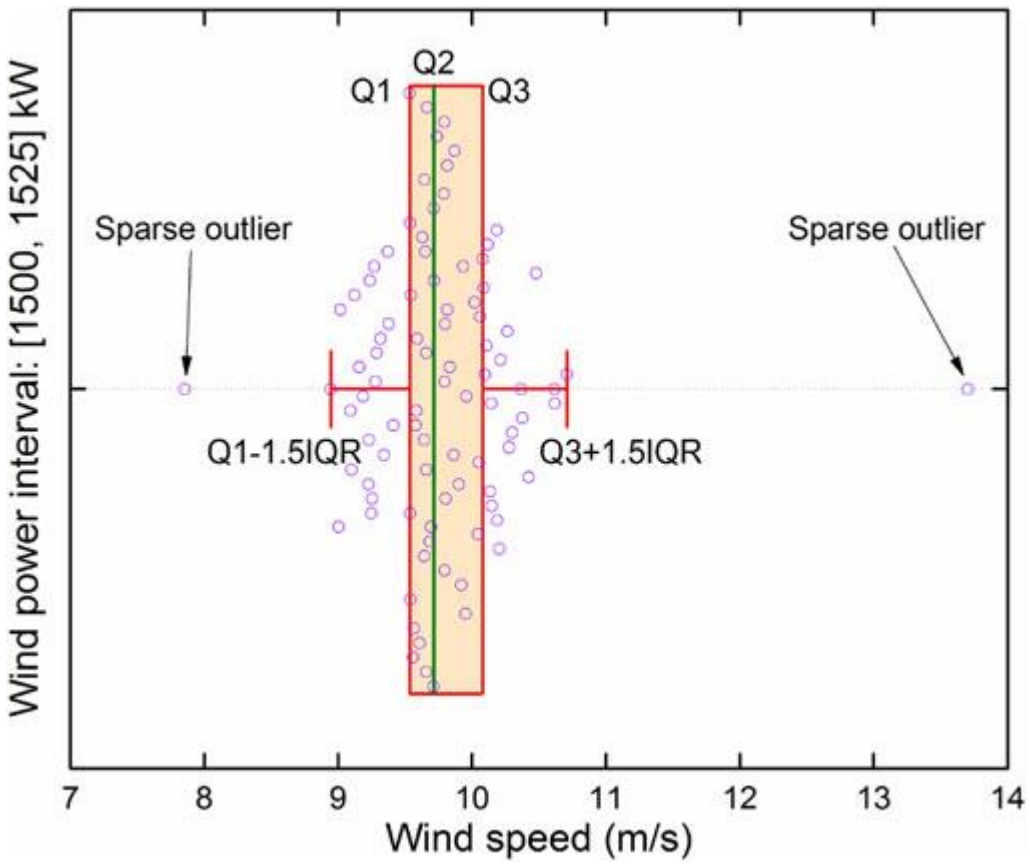
2. constant로 채워 넣기

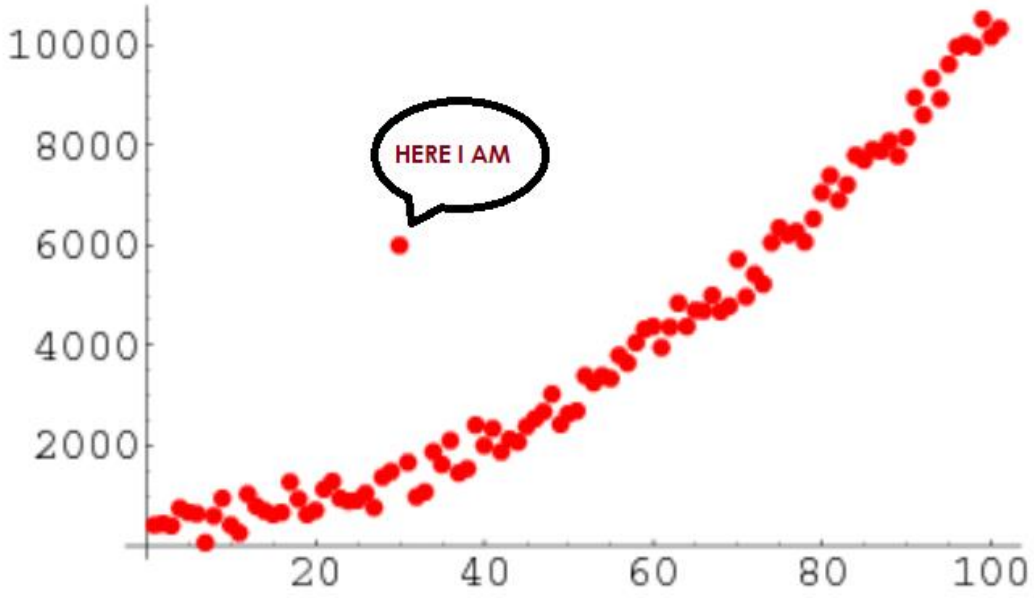
	col1	col2	col3	col4	col5
0	2	5.0	3.0	6	NaN
1	9	NaN	9.0	0	7.0
2	19	17.0	NaN	9	NaN

df.fillna(0)

	col1	col2	col3	col4	col5
0	2	5.0	3.0	6	0.0
1	9	0.0	9.0	0	7.0
2	19	17.0	0.0	9	0.0

04 Removing Outlier





Outlier는 다른 observed value와 크게 다른 data point입니다.

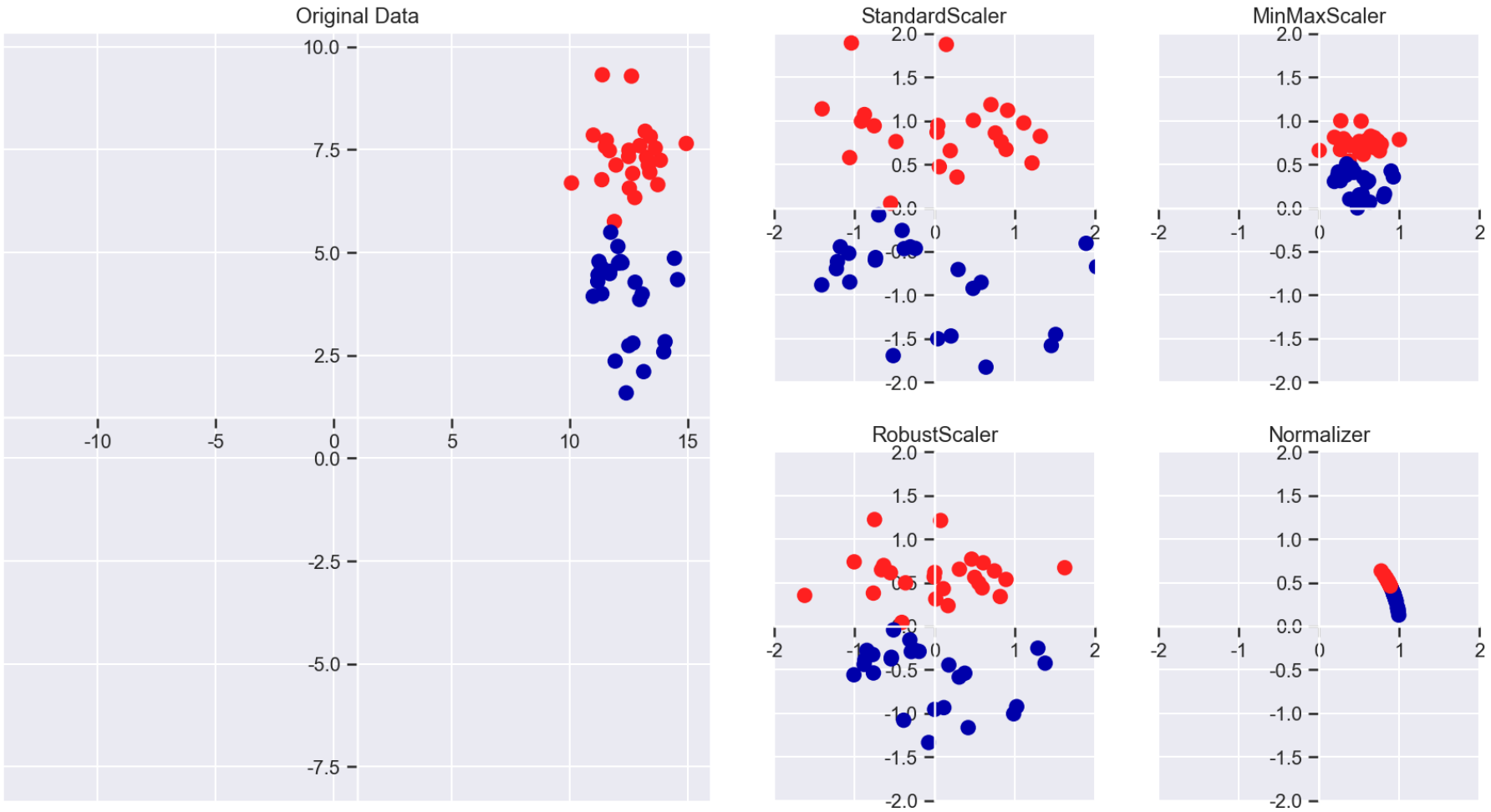
원인으로는 측정의 가변성, 새로운 데이터의 표시로 인한 것일 수 있습니다.

05 Correlation Analysis

```
Survived    1.000000
Sex         1.000000
Fare        0.203837
Parch       0.159120
SibSp       0.099943
Age         0.021962
Embarked   -0.037432
Pclass     -0.108615
Name: Survived, dtype: float64
```

크게 관련이 없는 column은 빼는 게 좋습니다.

06 Scaling



머신러닝 알고리즘은 입력 숫자들의 스케일이 많이 다르면 잘 작동하지 않습니다.

따라서, 숫자들의 범위 (스케일)를 일정하게 맞춰주는 scaling 작업이 필요합니다.

06 Scaling

(1) StandardScaler (skewness가 심할 때!)

각 feature의 평균을 0, 분산을 1로 변경합니다. 모든 특성들이 같은 스케일을 갖게 됩니다.

(2) RobustScaler

모든 특성들이 같은 크기를 갖는다는 점에서 StandardScaler와 비슷하지만, 평균과 분산 대신 median과 quartile을 사용합니다.

RobustScaler는 outlier에 영향을 받지 않습니다.

(3) MinMaxScaler (정규분포를 따를 때!)

모든 feature가 0과 1사이에 위치하게 만듭니다.

데이터가 2차원 셋일 경우, 모든 데이터는 x축의 0과 1 사이에, y축의 0과 1사이에 위치하게 됩니다.

(4) Normalizer

StandardScaler, RobustScaler, MinMaxScaler가 각 columns의 통계치를 이용한다면 Normalizer는 row마다 각각 정규화 합니다.

Normalizer는 유클리드 거리가 1이 되도록 데이터를 조정합니다. (유클리드 거리는 두 점 사이의 거리를 계산할 때 쓰는 방법)