

# 7일차. 회귀 알고리즘의 이해와 실습

심선영 교수, 이주민 교수

# 강의 목표

- ❖ 회귀 모델의 개념과 유형을 이해한다.
- ❖ 선형 회귀 모델의 개념을 이해하고 실습을 통해 활용해 본다.
- ❖ 회귀 오차의 개념을 학습하고 실습해 본다.
- ❖ 회귀 트리 모델의 개념을 이해하고 실습을 통해 활용해 본다.
- ❖ 회귀 실습을 위한 Kaggle competition에 대해 알아보고 EDA를 수행한다.

# 강의 목차

## ❖ 회귀 모델

- 선형 회귀 - 단순, 다중
- 회귀 오차
- 회귀 트리

## ❖ 실습 competition 소개

- Bike Sharing Demand Competition 둘러보기
- Target, Features 확인
- 데이터 파일 확인
- EDA 실습

# 강의 스케줄

시간	목차		활동
0.5h	Overview		PPT 학습
1h	회귀 모델	선형 회귀	파이썬 실습
2h		회귀오차	파이썬 실습
1h		회귀 트리	파이썬 실습
1h	회귀실습 – Competition소개 및 EDA		PPT
2h			파이썬 실습
0.5h	Wrap-Up		학습 정리

# 회귀 모델

---

# 분류모델의 활용



스마트팜  
잡초만 검출하여 제초제 분무



생산라인  
불량품 검출



물류창고  
특정 상품 분류



의료  
X-Ray 영상 인식, 판독

# 회귀 분석

## ❖ 요인(변수)들 간의 인과관계 분석하여 결과를 예측하는 통계적 기법

- 원인이 되는 요인들을 찾아내어 결과와의 관계를 분석
- 발생한 사건이나 현상을 설명하기 위해 사용
- 해당 원인 요인을 이용하여 결과를 예측
  - 예) IQ와 성적, 흡연과 암 발생률 등

## ❖ 회귀 분석의 변수

- 종속변수(dependent variable) 혹은 반응변수(response variable): 다른 변수의 영향을 받는 변수  
→ Target
- 독립변수(independent variable) 혹은 설명 변수(explanatory variable): 다른 변수에 영향을 주는 변수  
→ Feature

# 회귀 모델의 유형

구분		회귀 모델
독립변수의 개수	한 개	단순 회귀 (Simple Regression)
	두 개 이상	다중 회귀 (Multiple Regression)
예측 함수의 형태	선형 함수	선형 회귀 (Linear Regression)
	비선형 함수	비선형 회귀 (Non-Linear Regression)
규제의 유무와 형태 (선형 회귀에서)	없음	일반 선형 회귀 (General Linear Regression)
	L1규제	라쏘 회귀 (Lasso Regression)
	L2규제	릿지 회귀 (Ridge Regression)
	L1과 L2규제 결합	엘라스틱 넷 회귀 (Elastic Net Regression)



# 단순 회귀(Simple Regression)

❖ 단순 선형 회귀 (simple linear regression)

$$Y = W_0 + W_1 X$$

$Y_i$  = 종속변수  
 $X_i$  = 독립변수  
 $W_0$  = 절편  
 $W_1$  = 기울기

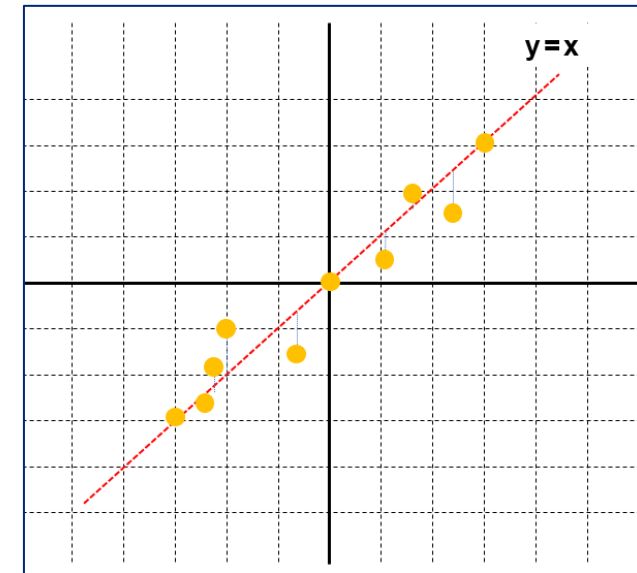
❖ 회귀 모델의 설명력:  $R^2$  (R Square)

■ 종속변수의 분산(변동성) 가운데 회귀모델에 의해 설명되는 비율

■  $R^2 = \frac{\text{예측값의 분산}}{\text{실제값의 분산}}$

■ 예) 개인소득- 종속변수, 경력- 독립변수

● R-제곱 = 0.83: 개인소득 분산의 83%가 경력에 의해 설명된다는 의미



# 다중 회귀(Multiple Regression)

❖ 다중 선형 회귀 (multiple linear regression)

- 결과값(Y)을 예측하기 위해 두 개 이상의 독립변수(feature)를 사용

$$Y = W_0 + W_1 X_1 + W_2 X_2 + \dots + W_n X_n$$

# 통계학의 회귀 VS 머신러닝의 회귀

❖ 가설: OO독립변수는 OO 종속변수에 영향을 미칠 것이다 (연관이 있을 것이다)

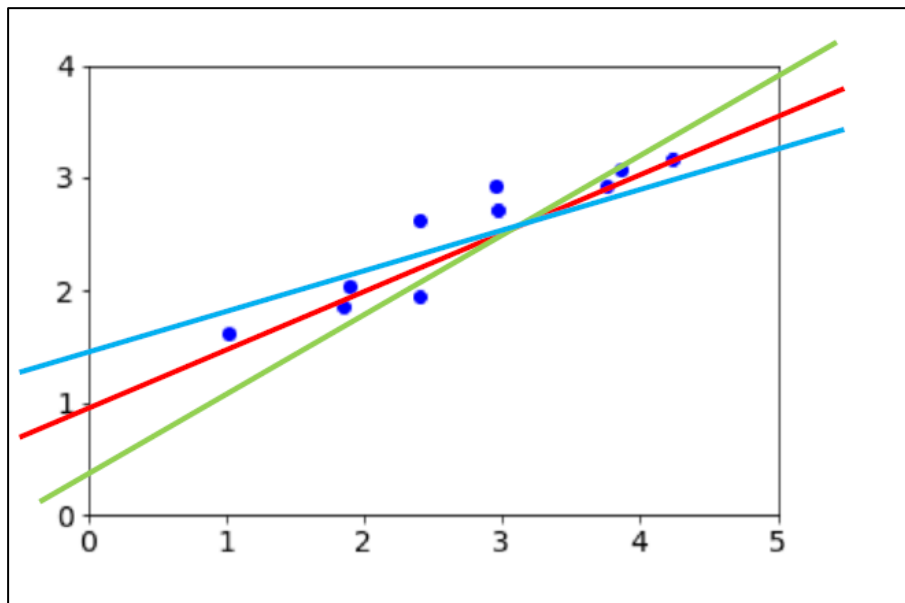
## ❖ 통계학

- 정교한 수학 방법론으로 소량의 데이터에서 의미를 찾으려고 노력
- 데이터에 관한 다양한 탐색에 중점 – 인간의 전문성이 중요
- 탐색이 마무리되고 가설 검정이 끝난 후 회귀모델이 만들어지면 더는 바뀌지 않는 상태로 예측에 사용 (탐색과정이 중요!)

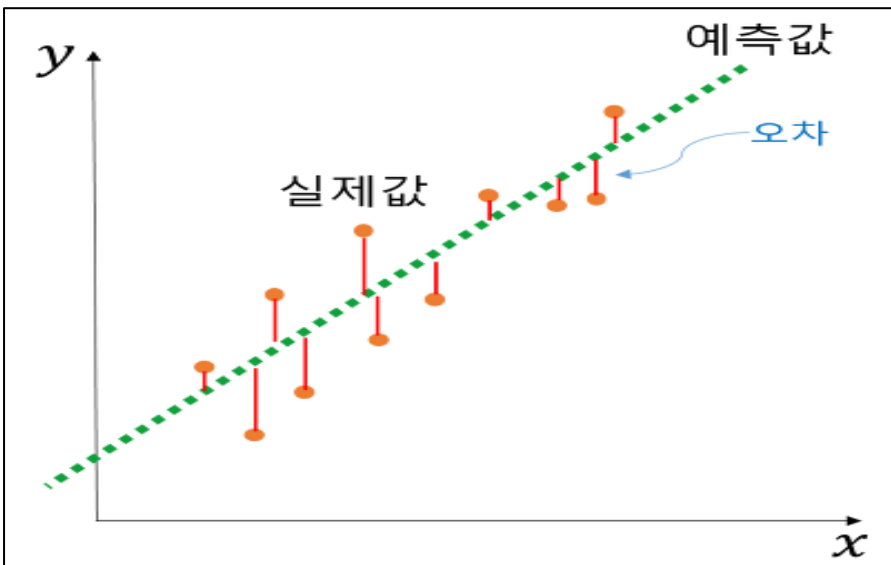
## ❖ 머신러닝

- 데이터가 많아지고 머신러닝이 시작되면서 탐색보다는 예측의 정확성에 중점
- 방대한 데이터를 기반으로 어떤 모델이 적합한지 다양하게 테스트해 보고 그 가운데 예측력이 가장 뛰어난 모델을 선정 – 데이터 자체의 패턴을 활용한 기계 학습
- 해당 데이터만을 수없이 반복 학습하다 보면 **과대적합** 문제 → **규제** 회귀를 사용

# 회귀 오차



여러 회귀 모델 중 **오차가 가장 적은 모델**을 찾음



## ❖ 오차의 계산 방법 (회귀 모델 성능 평가 지표)

■ 음의 오차와 양의 오차가 상쇄되지 않도록 절대값 또는 제곱하여 합산

### ① Mean Absolute Error (MAE)

$$\frac{\sum_{\text{all examples}} |\text{predicted} - \text{actual}|}{N}$$

$$\frac{|(2-1)| + |(4-3)| + |(6-6)| + |(8-9)| + |(10-12)|}{5} = \frac{5}{5} = 1$$

$\hat{Y}$	$Y$
2	1
4	3
6	6
8	9
10	12

### ② Mean Square Error (MSE)

$$\frac{\sum_{\text{all examples}} |\text{predicted} - \text{actual}|^2}{N}$$

$$\frac{|(2-1)|^2 + |(4-3)|^2 + |(6-6)|^2 + |(8-9)|^2 + |(10-12)|^2}{5} = \frac{7}{5} = 1.4$$

### ③ Root Mean Square Error (RMSE)

$$\sqrt{\frac{\sum_{\text{all examples}} |\text{predicted} - \text{actual}|^2}{N}}$$

$$\sqrt{\frac{|(2-1)|^2 + |(4-3)|^2 + |(6-6)|^2 + |(8-9)|^2 + |(10-12)|^2}{5}} = \sqrt{\frac{7}{5}} \approx 1.18$$

### ④

$$R^2 = \frac{\text{예측값의 분산}}{\text{실제값의 분산}}$$

# 회귀 오차

❖ 일반적으로 오차의 절대값보다는 제곱값을 많이 사용함 (MSE)

- 미분을 통해 최소값 찾기가 용이하기 때문

(Y: 실제값,  $\hat{Y}$ : 예측값)

❖  $error^2 = (Y - \hat{Y})^2 = (Y - (W_0 + W_1 X))^2$

$$\hat{Y} = W_0 + W_1 X$$

- X에 대한 이차식 → 실수의 제곱은 음이 될 수 없으므로 이 값의 최소값은 0

❖ 결국, 이 값이 최소(0)가 되도록 하는  $W_0, W_1$ 을 구하는 것이 목표!

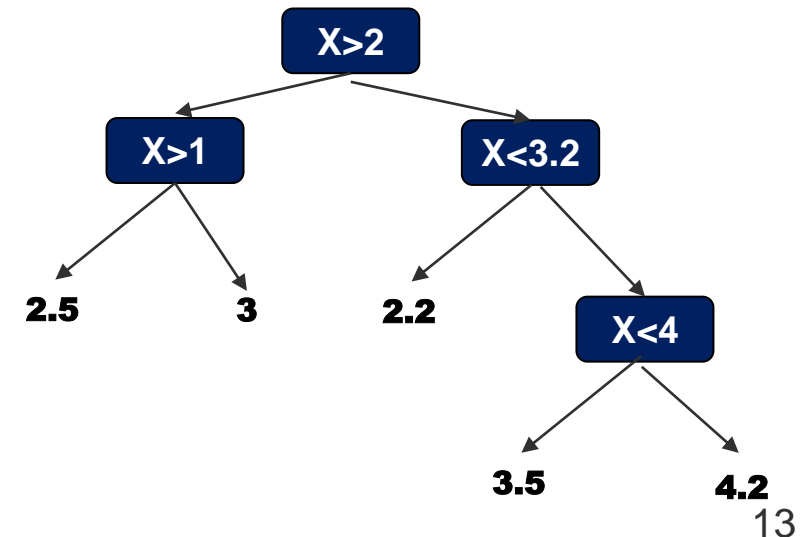
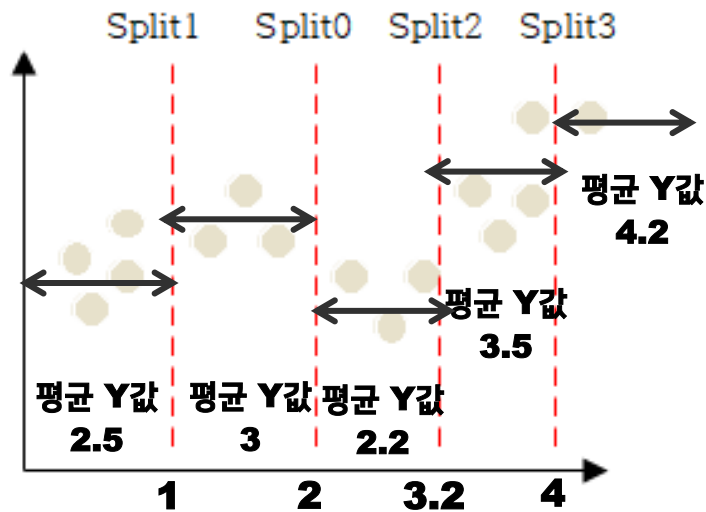
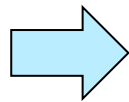
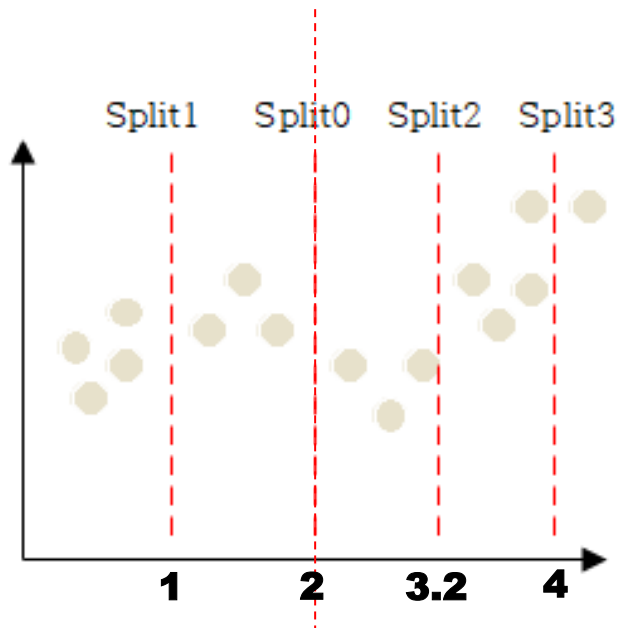
- 통계학: 수학적 방법으로 회귀 계수인  $W_0, W_1$ 을 계산
- 머신러닝: 주어진 데이터를 통해 오차 함수를 학습하여 회귀 계수인  $W_0, W_1$ 을 계산

- 오차제곱의 합 → 손실함수 or 비용함수

- 다수의 feature를 사용하는 경우 (=다수의 독립변수) →  $W_0, W_1, W_2, \dots, W_n$  → 비용함수에서 계산해야 할 회귀 계수(파라미터)가 많아짐 → 통계학의 수학적 접근보다 머신러닝의 학습적 접근이 더 유리

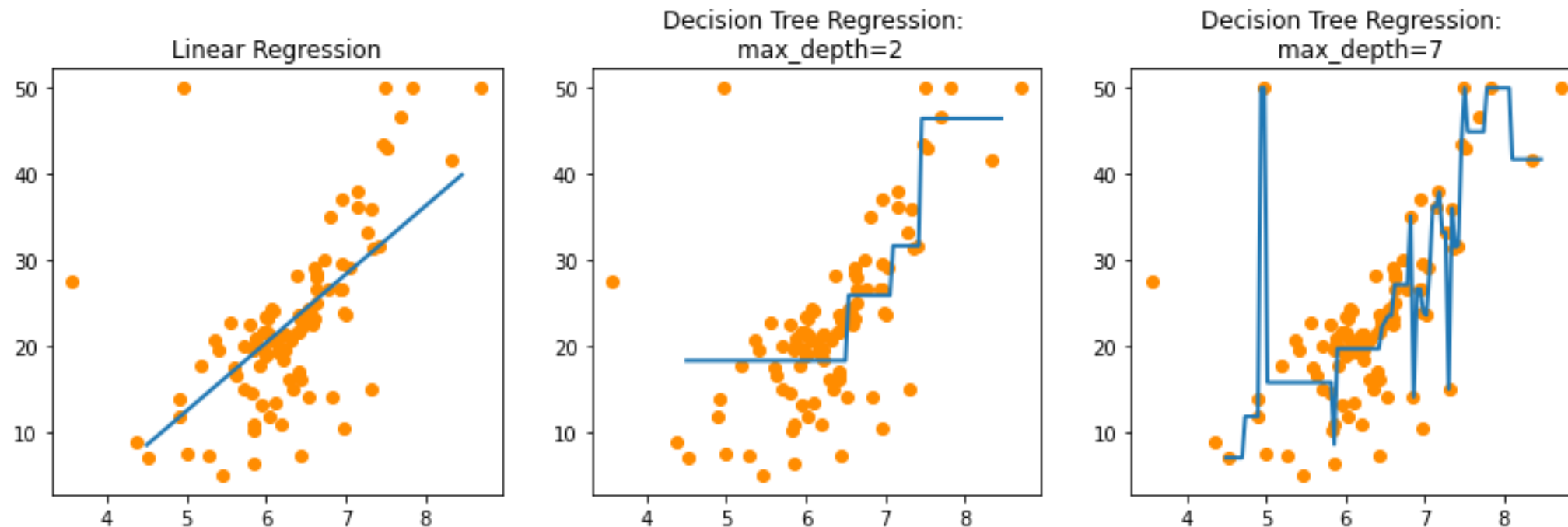
# 회귀 트리

- ❖ 회귀 계수를 구하는 **회귀 함수** 대신 **트리**를 사용하는 하는 회귀 모델
- ❖ 분류를 위한 결정 트리 방식과 유사
  - 데이터 셋을  $X$ 값의 균일도를 반영한 지니 계수에 따라 분할
  - 말단 노드별 소속된 데이터의 평균값을 구해서 리프노드별 최종 결정 값으로 사용!
    - 결정 트리: 말단 노드에 속한 특정 클래스의 레이블을 결정
    - 회귀 트리: 말단 노드에 속하는 데이터의 평균값으로 최종 예측



# 회귀 트리

- 선형회귀는 직선으로 예측 회귀선을 표현
- 회귀트리의 경우 분할되는 노드에 따라 가지를 만들면서 계단 형식의 회귀선이 생성



## ■ 사이킷런 Estimator Class

알고리즘	분류 Estimator Class	회귀 Estimator Class
의사결정나무 (Decision Tree)	DecisionTreeClassifier	DecisionTreeRegressor

# 회귀 트리

- 분류(classification)모델에서 설명한 tree기반의 모든 알고리즘은 회귀에서도 사용됨
  - DecisionTree, RandomForest, GBM 등
- Tree의 생성이 CART(Classification And Regression Trees) 알고리즘에 기반하고 있기 때문
- 사이킷런 **Estimator Class**

알고리즘	분류 Estimator Class	회귀 Estimator Class
Decision Tree	DecisionTreeClassifier	DecisionTreeRegressor
Random Forest	RandomForestClassifier	RandomForestRegressor
Gradient Boosting	GradientBoostingClassifier	GradientBoostingRegressor
XGBoost	XGBClassifier	XGBRegressor
LightGBM	LGBMClassifier	LGBMRegressor



# 회귀 실습을 위한 Kaggle competition 소개

---

# Bike Sharing Demand Competition 둘러보기

❖ 캐글 home에서 “Bike Sharing Demand”로 검색

❖ 검색 결과 중 “Competition”으로 표시된 “Bike Sharing Demand”선택

The screenshot shows the Kaggle search results for 'bike sharing demand'. On the left, there are filters for Dataset Size, Dataset File Types, Dataset License, Notebook Language, and Competition Evaluation Algorithm. The main results list several notebooks, with the 'Bike Sharing Demand' competition entry highlighted by a red dashed box.

**Filters:**

- Dataset Size:** small (11)
- Dataset File Types:** csv (9), ipynb (3), txt (1)
- Dataset License:** Other (8), Commercial (3)
- Notebook Language:** Python (2,026), R (1,448), Julia (1)
- Competition Evaluation Algorithm:** RMSLE (10), RMSE (9), AUC (2)

**Search Results:**

- Bike1** (Notebook) by seonyoungs. Private, 6 days ago, 3m to run, Python. 0 upvotes. Code snippet: `/input/bike-sharing-demand/train.csv') train.head() train.info() datetime의 데이터타입(Dtype)이 object라는 것은`
- BIKE SHARING DEMAND [ RMSLE:: 0.3194]** (Notebook) by Raj Mehrotra. 3 years ago, 2m to run, Python. 230 upvotes.
- BIKE SHARING DEMAND [ RMSLE:: 0.3194]** (Competition) with a bicycle icon. Playground. 7 years ago, 3242 teams.
- EDA & Ensemble Model (Top 10 Percentile)** (Notebook) by Vivek Srinivasan. 5 years ago, 22m to run, Python. 528 upvotes.
- Linear Regression Model - Bike Sharing Demand** (Notebook)

# Bike Sharing Demand Competition 둘러보기


❖ <https://www.kaggle.com/competitions/bike-sharing-demand>

- 워싱턴 D.C 소재의 자전거 대여 스타트업인 Capital Bikeshare의 데이터를 활용
- 특정 시간대에 얼마나 많은 사람들이 자전거를 대여하는지 예측하는 것

Playground Prediction Competition

## Bike Sharing Demand

Forecast use of a city bikeshare system

 Kaggle · 3,242 teams · 7 years ago

Overview

Data

Code

Discussion

Leaderboard

Rules

Team

My Submissions

Late Submission

...

Overview





Description

Evaluation

Get started on this competition through [Kaggle Scripts](#)

Bike sharing systems are a means of renting bicycles where the process of obtaining membership, rental, and bike return is automated via a network of kiosk locations throughout a city. Using these systems, people are able rent a bike from a one location and return it to a different place on an as-needed basis. Currently, there are over 500 bike-sharing programs around the world.

The data generated by these systems makes them attractive for researchers because the duration of travel, departure location, arrival location, and time elapsed is explicitly recorded. Bike sharing systems therefore function as a sensor network, which can be used for studying mobility in a city. In this competition, participants are asked to combine historical usage patterns with weather data in order to forecast bike rental demand in the Capital Bikeshare program in Washington, D.C.



# Bike Sharing Demand Competition 둘러보기

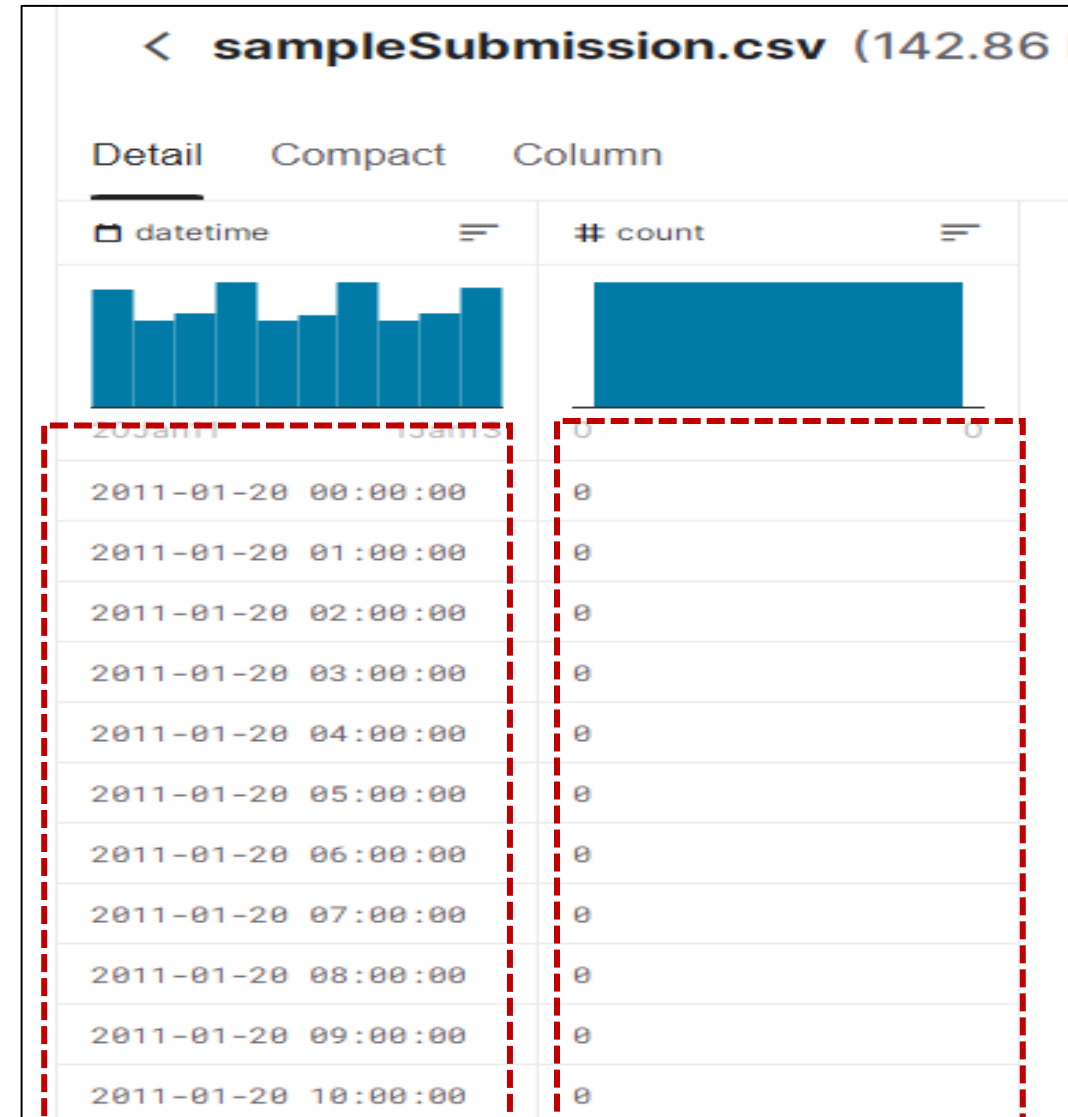
❖ <https://www.kaggle.com/competitions/bike-sharing-demand>

❖ 시간대별 자전거 대여량 예측


→ 회귀 모델

❖ 어떤 feature들이 자전거 대여량에 영향을 미칠까?

- 하루 중 시간대?
- 요일? 근무일 여부?
- 날씨?
- 계절 (월)?



# Data Fields확인

 Kaggle · 3,242 teams · 7 years ago

[Overview](#) [Data](#) [Code](#) [Discussion](#) [Leaderboard](#) [Rules](#) [Team](#) [My Submissions](#) [Late Submission](#) [...](#)

## Data Description

[See, fork, and run a random forest benchmark model through Kaggle Scripts](#)

You are provided hourly rental data spanning two years. For this competition, the training set is comprised of the first 19 days of each month, while the test set is the 20th to the end of the month. You must predict the total count of bikes rented during each hour covered by the test set, using only information available prior to the rental period.

### Data Fields

- datetime - hourly date + timestamp
- season - 1 = spring, 2 = summer, 3 = fall, 4 = winter
- holiday - whether the day is considered a holiday
- workingday - whether the day is neither a weekend nor holiday
- weather - 1: Clear, Few clouds, Partly cloudy, Partly cloudy  
2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist  
3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds  
4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
- temp - temperature in Celsius
- atemp - "feels like" temperature in Celsius
- humidity - relative humidity
- windspeed - wind speed
- casual - number of non-registered user rentals initiated
- registered - number of registered user rentals initiated
- count - number of total rentals

# Target, Features 확인

❖ Target: **count** (총대여 수량)

❖ Features

■ **count** = casual + registered

■ **datetime ~ windspeed**

datetime - hourly date + timestamp (년-월-일-시-분-초 형태)

season – 1 = spring, 2 = summer, 3 = fall, 4 = winter

holiday – whether the day is considered a holiday (공휴일이면 1, 아니면 0)

workingday - whether the day is neither a weekend nor holiday (근무일이면 1, 아니면 0)

weather (날씨)

1: Clear, Few clouds, Partly cloudy, Partly cloudy

2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist

3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds

4: Heavy Rain + Ice Pellets + Thunderstorm + Mist, Snow + Fog

temp - temperature in Celsius (온도)

atemp - "feels like" temperature in Celsius (체감온도)

humidity - relative humidity (습도)

windspeed - wind speed (풍속)

casual - number of non-registered user rentals initiated (비회원의 대여량)

registered - number of registered user rentals initiated (회원의 대여량)

count - number of total rentals (총 대여량)

# 데이터 파일 다운로드

❖ 총 3개의 CSV

❖ 필요 시 다운로드

(캐글 노트북 작업을 위해서는  
다운받지 않아도 됨)

The screenshot shows the Kaggle Data Explorer interface for the 'test.csv' file (323.86 kB). The interface includes a top navigation bar with tabs: Overview, Data (highlighted with a red dashed box), Code, Discussion, Leaderboard, Rules, Team, My Submissions, and a Late Submission button. Below the navigation bar, the 'Data Explorer' section shows a list of files: sampleSubmission.csv, test.csv (highlighted with a red dashed box), and train.csv. The 'Summary' section at the bottom left shows '3 files' and '23 columns', with a 'Download All' button (highlighted with a red dashed box). The main content area displays a table view of the 'test.csv' data, with columns: datetime, # season, # holiday, and # workingday. The table shows data for the year 2011, starting from 2011-01-20 00:00:00. The 'datetime' column is highlighted with a red dashed box. The table also includes bar charts for each column, with a tooltip for the '# workingday' chart showing a count of 4,453 for the value 1.

datetime	# season	# holiday	# workingday
2011-01-20 00:00:00	1	0	1
2011-01-20 01:00:00	1	0	1
2011-01-20 02:00:00	1	0	1
2011-01-20 03:00:00	1	0	1
2011-01-20 04:00:00	1	0	1
2011-01-20 05:00:00	1	0	1
2011-01-20 06:00:00	1	0	1
2011-01-20 07:00:00	1	0	1
2011-01-20 08:00:00	1	0	1
2011-01-20 09:00:00	1	0	1
2011-01-20 10:00:00	1	0	1
2011-01-20 11:00:00	1	0	1
2011-01-20 12:00:00	1	0	1
2011-01-20 13:00:00	1	0	1
2011-01-20 14:00:00	1	0	1

❖ Day7. Wrap-up 설문

- <https://forms.gle/vH8XEUvbfcWZFbNv5>