Competitions overview

WINNING A KAGGLE COMPETITION IN PYTHON

Yauhen Babakhin
Kaggle Grandmaster

Instructor

Yauhen Babakhin

- Master's Degree in Applied Data Analysis
- 5 years of working experience in Data Science
- Kaggle competitions Grandmaster
- Gold medals in both classic Machine Learning and Deep Learning competitions

다들 아시겠지만, Kaggle은 데이터 사이언스 및 기계학습 대회를 위한 웹 플랫폼입니다.

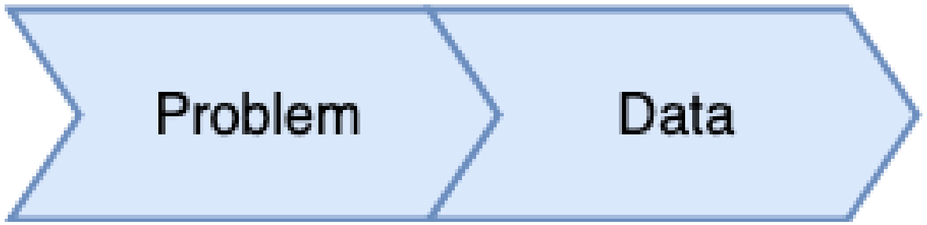
Kaggle benefits

1. Get practical experience on the real-world data
2. Develop portfolio projects
3. Meet a great Data Science community
4. Try new domain or model type
5. Keep up-to-date with the best performing methods

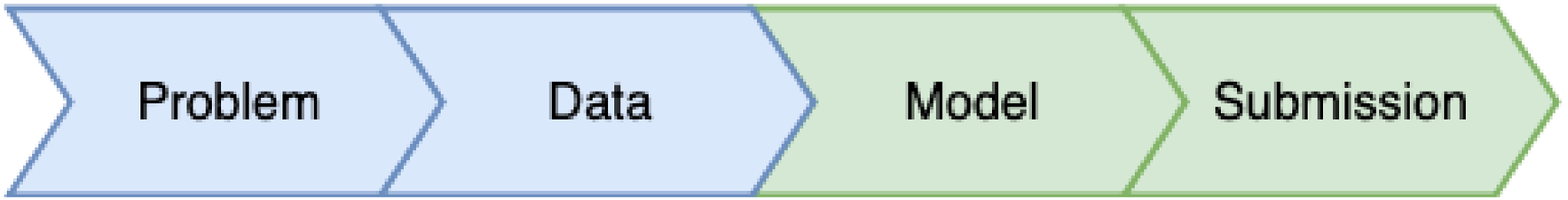
kaggle을 하면 실제 데이터 세트로 작업할 수 있고, 프로젝트를 만들고, 커뮤니티를 만나서 성장하고, 새로운 도메인 또는 모델 유형에 댕나 경험을 얻고, 기계학습 방법을 최신 상태로 유지할 수 있다는 점의 장점을 가지고 있습니다.

Competition process

먼저 Kaggle의 process를 보면 문제정의와 문제를 해결하기 위한 데이터를 제공합니다.

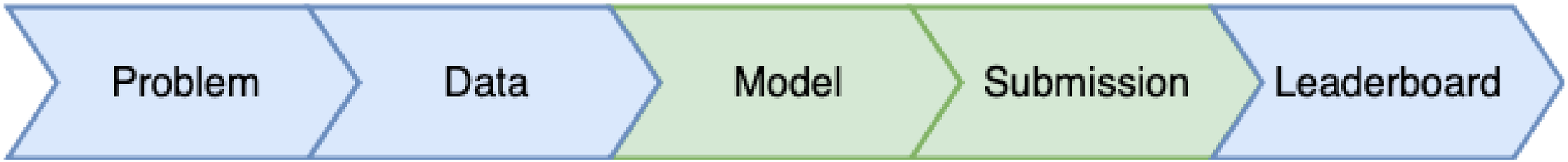


Competition process



그런 다음 모델을 개발하여 업로드합니다.

Competition process



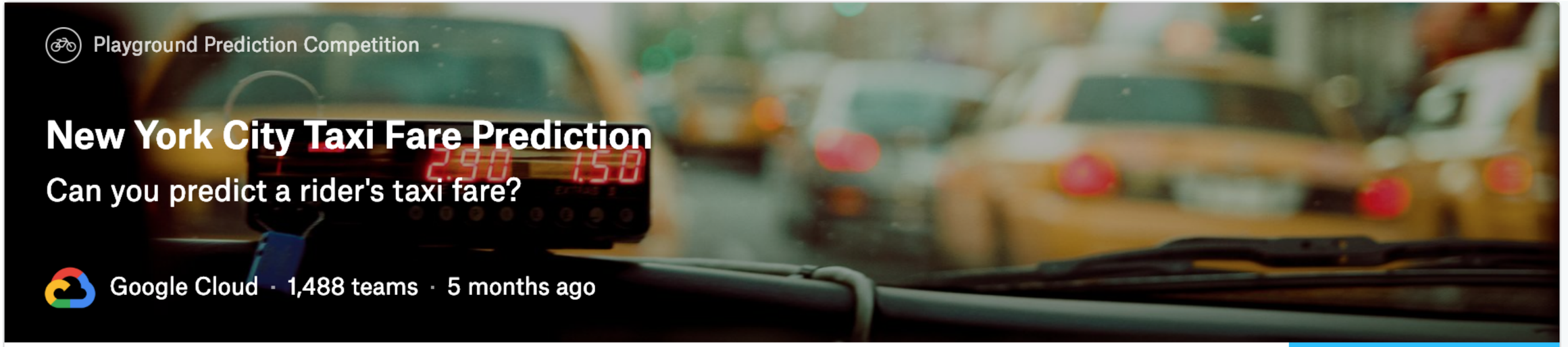
마지막으로, 내 모델이 남들보다 얼마나 좋은지는 리더보드를 통해 알 수 있습니다.

How to participate

1. Go to **http://kaggle.com** website and select the competition
2. Download the data
3. Start building the models!

참여방법은 kaggle 사이트에 들어가서 참여할 대회를 선택하고,
data를 다운로드하여 모델을 구축합니다.

New York city taxi fare prediction



Overview | Data | Kernels | Discussion | Leaderboard | Rules | Join Competition

Overview

예를 들어, 뉴욕시 택시 요금 예측이라는 과거의 kaggle대회를 예시로 할 것이고,
목표는 픽업 및 드롭 위치가 주어진 뉴욕시의 택시 승차 요금 금액을 예측하는 것입니다.

- **Description**
- **Evaluation**
- **Timeline**

In this playground competition, hosted in partnership with Google Cloud and Coursera, you are tasked with predicting the fare amount (inclusive of tolls) for a taxi ride in New York City given the pickup and dropoff locations. While you can get a basic estimate based on just the distance between the two points, this will result in an RMSE of $5-$8, depending on the model used (see the starter code for an example

Train and Test data

```
import pandas as pd

# Read train data
taxi_train = pd.read_csv('taxi_train.csv')
taxi_train.columns.to_list()
```

```
['key',
 'fare_amount',
 'pickup_datetime',
 'pickup_longitude',
 'pickup_latitude',
 'dropoff_longitude',
 'dropoff_latitude',
 'passenger_count']
```

```
# Read test data
taxi_test = pd.read_csv('taxi_test.csv')
taxi_test.columns.to_list()
```

```
['key',
 'pickup_datetime',
 'pickup_longitude',
 'pickup_latitude',
 'dropoff_longitude',
 'dropoff_latitude',
 'passenger_count']
```

데이터를 pandas로 load해보면 fare_amount가 예측하고자 하는 목표변수이기 때문에 test_set에서는 fare_amount를 제외한 column을 가지고 있습니다.

Sample submission

```
# Read sample submission
taxi_sample_sub = pd.read_csv('taxi_sample_submission.csv')
taxi_sample_sub.head()
```

```
                            key     fare_amount
0     2015-01-27 13:08:24.0000002     11.35
1     2015-01-27 13:08:24.0000003     11.35
2     2011-10-08 11:53:44.0000002     11.35
3     2012-12-01 21:12:12.0000002     11.35
4     2012-12-01 21:12:12.0000003     11.35
```

모든 kaggle대회는 sample 제출 파일을 보여주는데, 이 형식에 따라 제출하면 됩니다.
예상대로 fare_amount와 key 두개를 예측하면 됩니다.

Let's practice!

WINNING A KAGGLE COMPETITION IN PYTHON

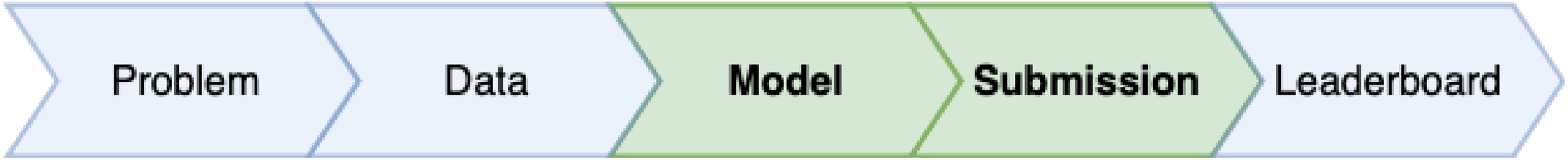
Prepare your first submission

WINNING A KAGGLE COMPETITION IN PYTHON

Yauhen Babakhin
Kaggle Grandmaster

datacamp

What is submission



제출은 일반적으로 test예측이 포함된 .csv파일로 제출합니다.

New York city taxi fare prediction

```
# Read train data
taxi_train = pd.read_csv('taxi_train.csv')
taxi_train.columns.to_list()
```

택시 요금 예측 문를 다시 다루면,
to_list를 이용해서 columns을 list로 볼 수 있고,

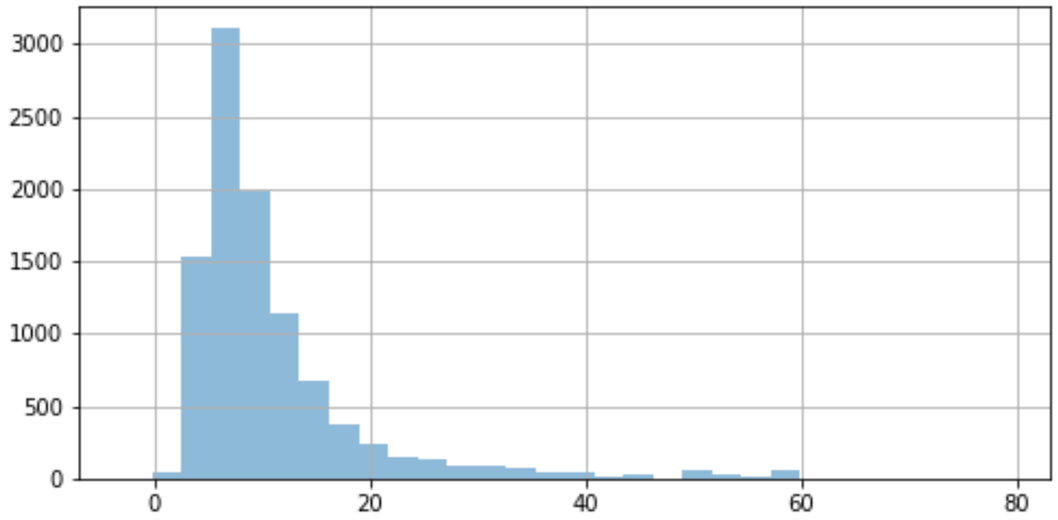
```
['key',
 'fare_amount',
 'pickup_datetime',
 'pickup_longitude',
 'pickup_latitude',
 'dropoff_longitude',
 'dropoff_latitude',
 'passenger_count']
```

Problem type

```
import matplotlib.pyplot as plt

# Plot a histogram
taxi_train.fare_amount.hist(bins=30, alpha=0.5)
plt.show()
```

히스토그램을 통해서 문제 유형이 회귀인지 분류인지 다른것인지 결정합니다.
연속변수인것을 확인할 수 있고, 회귀문제를 다룰 예정입니다.



Build a model

```
from sklearn.linear_model import LinearRegression


# Create a LinearRegression object
lr = LinearRegression()
```

선형회귀를 통해서 모델을 fit 시켜줍니다.

```
# Fit the model on the train data
lr.fit(X=taxi_train[['pickup_longitude', 'pickup_latitude', 'dropoff_longitude',
                     'dropoff_latitude', 'passenger_count']],
       y=taxi_train['fare_amount'])
```

Predict on test set

```
# Select features
features = ['pickup_longitude', 'pickup_latitude',
            'dropoff_longitude', 'dropoff_latitude',
            'passenger_count']
```

```
# Make predictions on the test data
taxi_test['fare_amount'] = lr.predict(taxi_test[features])
```

새로운 예측 결과는 fare_amount 열에 저장됩니다.

Prepare submission

key와 fare_amount 열로 구성된 파일을 저장하여 업로드합니다.

```
# Read a sample submission file
taxi_sample_sub = pd.read_csv('taxi_sample_submission.csv')
taxi_sample_sub.head(1)
```

```
                            key     fare_amount
0     2015-01-27 13:08:24.0000002     11.35
```

```
# Prepare a submission file
taxi_submission = taxi_test[['key', 'fare_amount']]

# Save the submission file as .csv
taxi_submission.to_csv('first_sub.csv', index=False)
```

Let's practice!

WINNING A KAGGLE COMPETITION IN PYTHON

Public vs Private leaderboard

WINNING A KAGGLE COMPETITION IN PYTHON

Yauhen Babakhin
Kaggle Grandmaster

datacamp

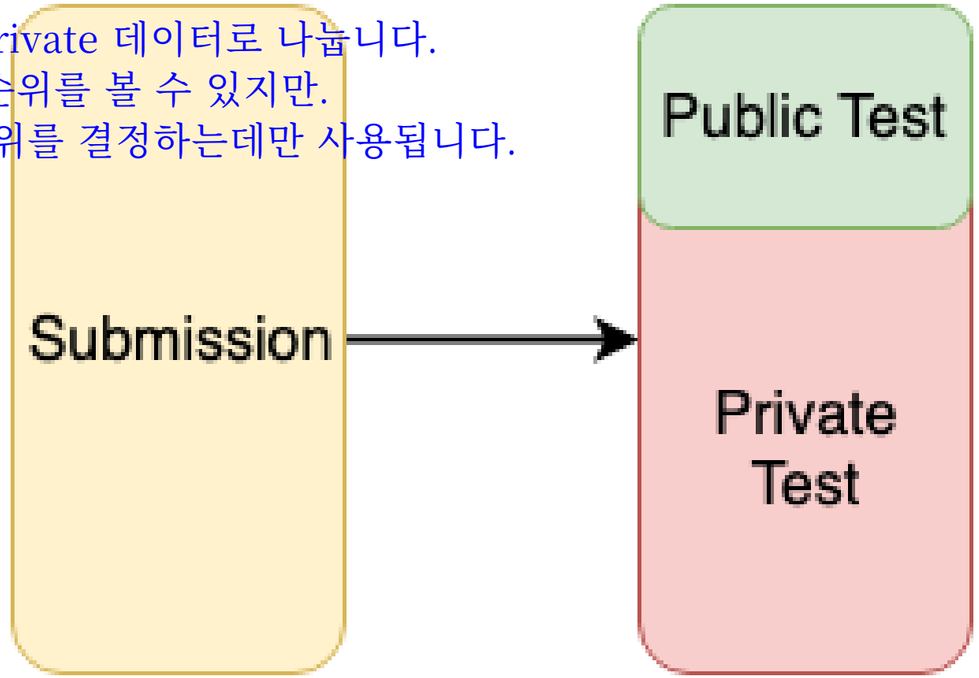
Competition metric

Evaluation metric	Type of problem
Area Under the ROC (AUC)	Classification
F1 Score (F1)	Classification
Mean Log Loss (LogLoss)	Classification
Mean Absolute Error (MAE)	Regression
Mean Squared Error (MSE)	Regression
Mean Average Precision at K (MAPK, MAP@K)	Ranking

각 대회는 순위를 매기는데 사용되는 메트릭을 지정하는데,
여기서 목표는 모델이 더 나은 메트릭 값을 달성하독 하는것입니다.
classification에서는 주로 AUC, F1 score, Mean Log Loss를 사용하고 회귀문제에서는 MAE, MSE 등 사용하는것을 볼 수 있습니다.

Test split

test set의 모든 case를 예측해야하지만,
kaggle에서는 test data를 public data와 private 데이터로 나눕니다.
public 테스트 케이스는 대회 중에 결과와 순위를 볼 수 있지만.
private test case는 대회가 끝나면 최종순위를 결정하는데만 사용됩니다.



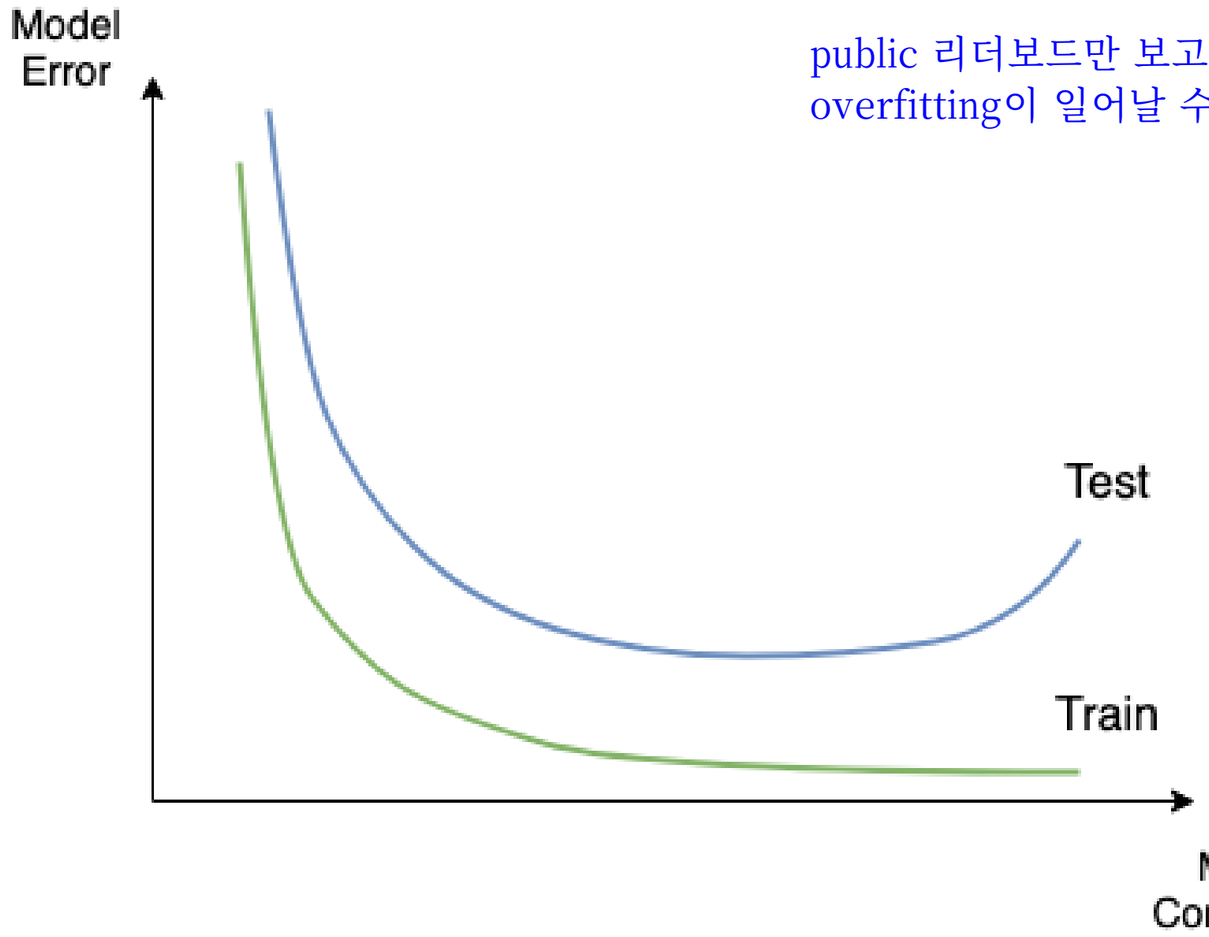
Leaderboards

```
# Write a submission file to the disk
submission[['id', 'target']].to_csv('submission_1.csv', index=False)
```

Submission	Public LB MSE	Private LB MSE
submission_1.csv	2.895	?

보통 하루에 약 5개의 제출할 수 있게 제한되어 있고,
public 파트의 리더보만 확인할 수 있습니다.

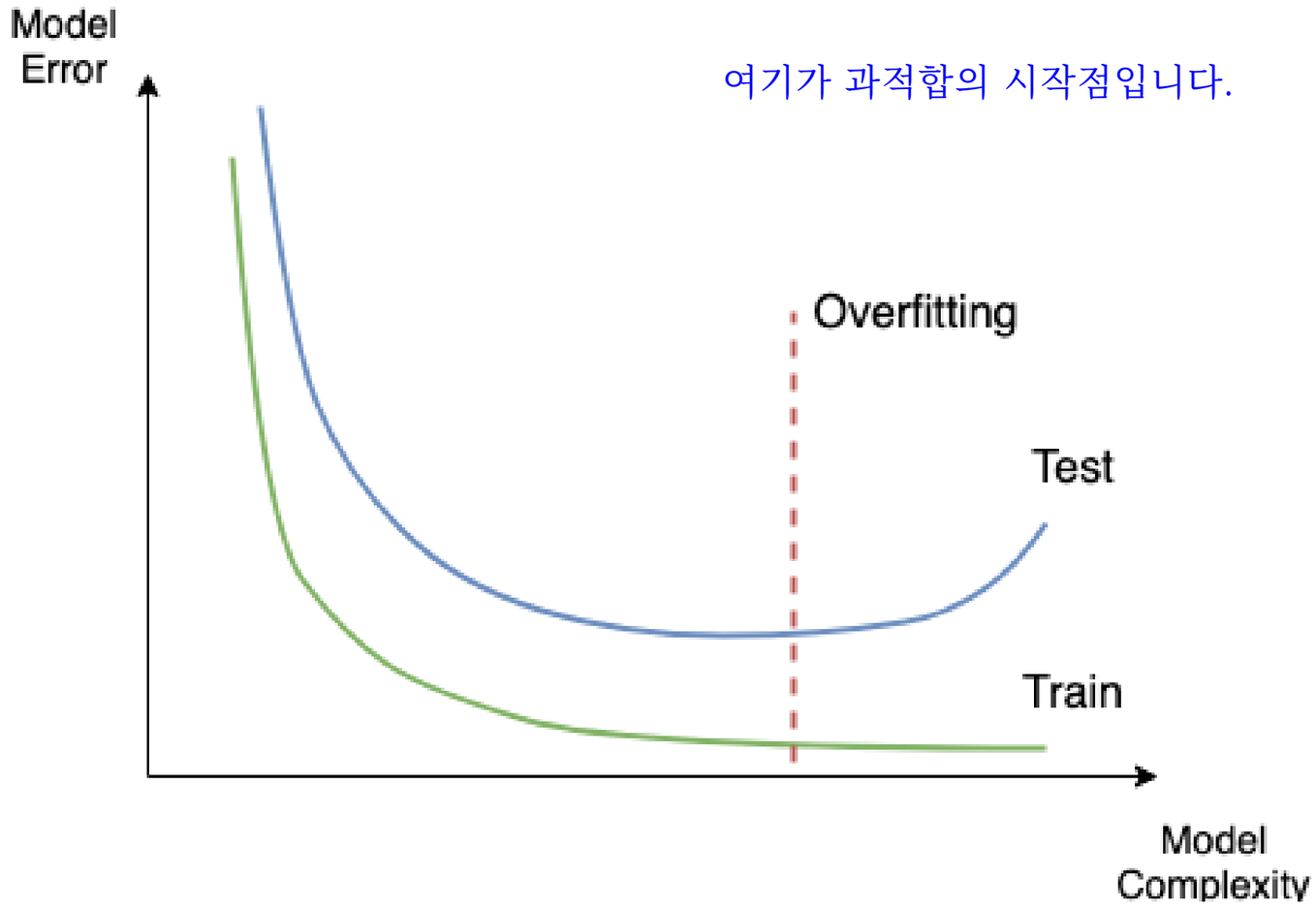
Overfitting



public 리더보드만 보고 모델을 고치다보면 overfitting이 일어날 수 있습니다.

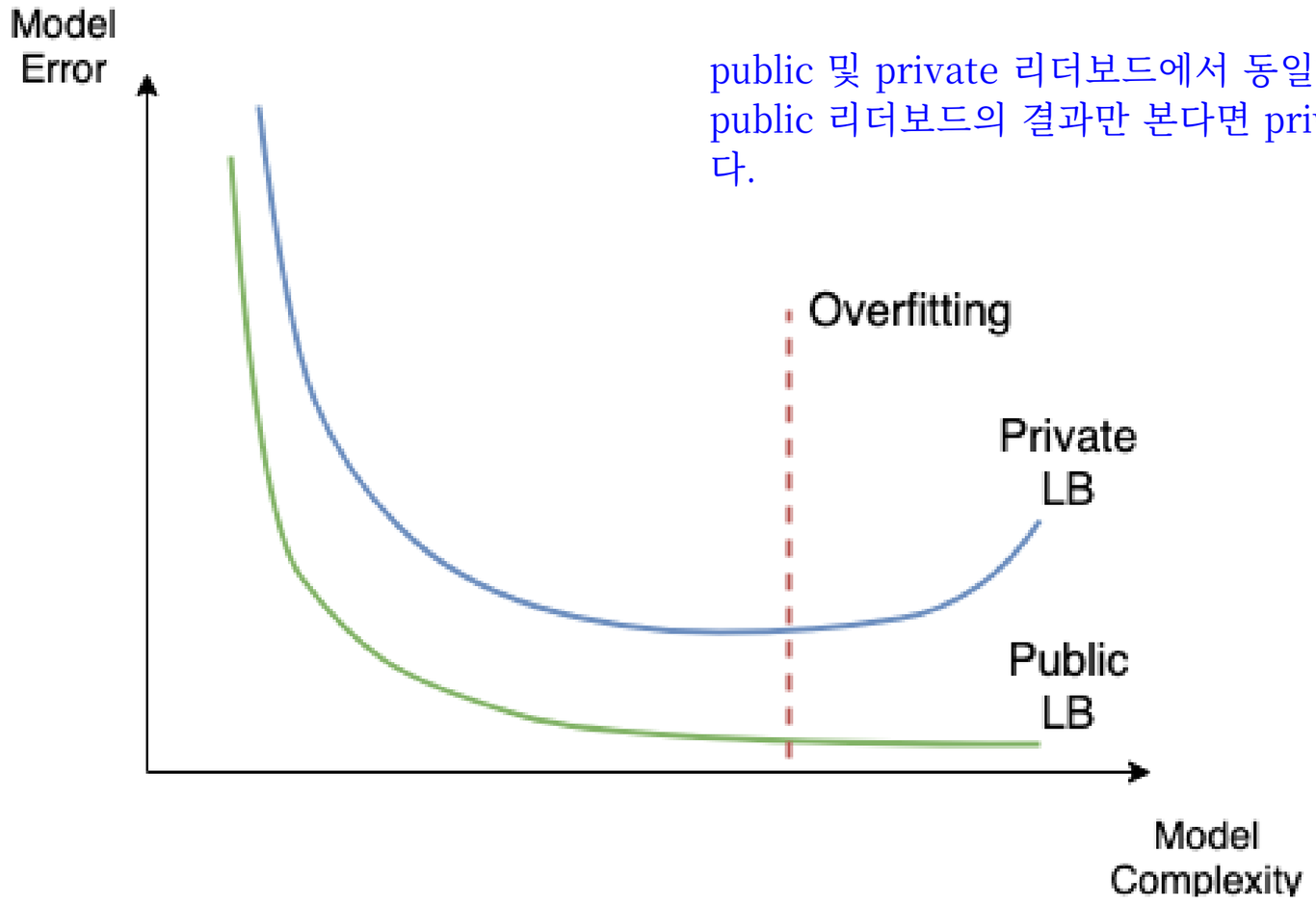
우리가 머신 러닝 모델을 개발하고 기차와 테스트 데이터 모두에서 오류율을 측정한다고 가정하자. 모델의 복잡성은 증가하지만, 열차 오류는 일반적으로 감소합니다. 이는 모델이 기차 데이터를 매우 잘 학습하기 때문에 매우 적은 오류로도 훌륭한 성능을 발휘하기 때문입니다. 그러나 테스트 오류는 어느 시점에서 증가할 수 있습니다.

Overfitting



여기가 과적합의 시작점입니다.

Overfitting



public 및 private 리더보드에서 동일한 일이 생길 수 있고,
public 리더보드의 결과만 본다면 private에서 overfitting될 수 있습니다.

Public vs Private leaderboard shake-up

#	△pub	Team Name
1	—	Kyle Boone
2	▲ 2	Mike & Silogram
3	▼ 1	Major Tom
4	▼ 1	AhmetErdem
5	—	SKZ Lost in Translation
6	▲ 2	Stefan Stefanov
7	▲ 3	hklee
8	▼ 1	rapids.ai
9	▼ 3	Three Musketeers
10	▲ 3	J&J

#	△pub	Team Name
1	▲ 1484	gmobaz
2	▲ 414	RHINODAVEB
3	▲ 1784	Jayden Tan
4	▲ 1599	mchahhou
5	▲ 2753	R.elsharawy
6	▲ 1132	DDgg
7	▲ 772	Maverix
8	▲ 15	dil-bert
9	▲ 213	zr17
10	▲ 1211	KG123

공개 리더보드와 개인 리더보드 순위 간의 차이를 '쉐이크업'이라고 하고,
public 리더보드에서는 kyle boone이 1등이였지만,
이 대회 우승자는 public 리더보드에서 1485등이었던 사람이 됐습니다.

Let's practice!

WINNING A KAGGLE COMPETITION IN PYTHON