

# MLflow 실습 2

1. Example code 살펴보기
2. Example code 실행
3. MLflow 데이터 저장 방식

## 1. Example code 살펴보기

- [https://github.com/mlflow/mlflow/tree/master/examples/sklearn\\_elasticnet\\_diabetes](https://github.com/mlflow/mlflow/tree/master/examples/sklearn_elasticnet_diabetes)

```
# VM 혹은 linux 사용자
wget https://raw.githubusercontent.com/mlflow/mlflow/master/examples/sklearn_elasticnet_diabetes/linux/train_diabetes.py

# Mac 사용자
wget https://raw.githubusercontent.com/mlflow/mlflow/master/examples/sklearn_elasticnet_diabetes/osx/train_diabetes.py
```

- mlflow 에서 example 로 제공해주는 다양한 example 중 하나인 `train_diabetes.py`
  - scikit-learn 패키지에서 제공하는 diabetes(당뇨병) 진행도 예측용 데이터로 ElasticNet 모델을 학습하여, predict 한 뒤 그 evaluation metric 을 MLflow 에 기록하는 예제
  - 442 명의 당뇨병 환자를 대상으로, 나이, 성별, bmi 등의 10 개의 독립변수(X) 를 가지고 1년 뒤 당뇨병의 진행률 (y) 를 예측하는 문제
- 데이터에 대한 자세한 분석과 ElasticNet 에 대한 자세한 설명은 생략하겠습니다.
  - ElasticNet : Linear Regression + L1 Regularization + L2 Regularization
    - parameter
      - **alpha** : Regularization coefficient
      - **l1\_ratio** : L1 Regularization 과 L2 Regularization 의 비율
- 코드를 함께 살펴보겠습니다.
  - **mlflow 와 연관된 부분**에 주목해주세요.
    - `mlflow.log_param`
    - `mlflow.log_metric`
    - `mlflow.log_model`

- `mlflow.log_artifact`

## 2. Example code 실행

```
# mlflow ui 를 수행한 디렉토리와 같은 디렉토리로 이동
cd mlflow-tutorial




# example 코드를 실행 후 mlflow 에 기록되는 것 확인
python train_diabetes.py
```

- model 관련 meta 정보와 더불어 pkl 파일이 저장된 것을 확인

### ▼ Parameters

Name	Value
alpha	0.05
l1_ratio	0.05

### ▼ Metrics

Name	Value
mae 	66.31
r2 	0.066
rmse 	78.59

### ▶ Tags

### ▼ Artifacts

model


MLmodel

conda.yaml

model.pkl

requirements.txt

ElasticNet-paths.png

Full Path:./mlruns/0/197c1e2206af4f3c88e48f6653febe53/artifacts... 

Register Model

### MLflow Model

The code snippets below demonstrate how to make predictions using the logged model. You can also [register it to the model registry](#) to version control

#### Model schema

Input and output schema for your model. [Learn more](#)

Name	Type
No schema. See <a href="#">MLflow docs</a> for how to include input and output schema with your model.	

#### Make Predictions

Predict on a Spark DataFrame:

```
import mlflow
logged_model = 'runs:/197c1e2206af4f3c88e48f6653febe53/model'

# Load model as a Spark UDF.
loaded_model = mlflow.pyfunc.spark_udf(spark, model_uri=logged_model)

# Predict on a Spark DataFrame.
df.withColumn('predictions', loaded_model(*columns)).collect()
```

- parameters, metrics, artifacts

- 다양한 parameter 로 테스트 후 mlflow 확인

```
python train_diabetes.py 0.01 0.01
python train_diabetes.py 0.01 0.75
python train_diabetes.py 0.01 1.0
python train_diabetes.py 0.05 1.0
python train_diabetes.py 0.05 0.01
python train_diabetes.py 0.5 0.8
python train_diabetes.py 0.8 1.0
```

- 다음과 같은 화면이 출력되며, metrics 와 parameter 를 한 눈에 비교할 수 있습니다.

Default

Experiment ID: 0

Notes

Showing 7 matching runs

Refresh Compare Delete Download CSV Sort by All Columns Search Filter Clear

metrics.rmse < 1 and params.model = "tree"

	Start Time	Run Name	User	Source	Version	Models	Metrics			Parameters	
							mse	r2	rmse	alpha	t1_ratio
<input type="checkbox"/>	14 minutes ago	-	kjy	train_diabe	-	sklearn	54.7	0.338	66.17	0.7	1.0
<input type="checkbox"/>	14 minutes ago	-	kjy	train_diabe	-	sklearn	53.21	0.364	64.84	0.5	1.0
<input type="checkbox"/>	14 minutes ago	-	kjy	train_diabe	-	sklearn	51.26	0.396	63.2	0.05	1.0
<input type="checkbox"/>	14 minutes ago	-	kjy	train_diabe	-	sklearn	51.05	0.395	63.25	0.01	1.0
<input type="checkbox"/>	14 minutes ago	-	kjy	train_diabe	-	sklearn	53.76	0.355	65.29	0.01	0.75
<input type="checkbox"/>	14 minutes ago	-	kjy	train_diabe	-	sklearn	60.09	0.229	71.4	0.01	0.01
<input type="checkbox"/>	14 minutes ago	-	kjy	train_diabe	-	sklearn	66.31	0.066	78.59	0.05	0.05

Load more

### 3. MLflow 데이터 저장 방식

```
cd mlruns/0
ls
# 굉장히 많은 디렉토리가 생성되었습니다.
# (각각의 알 수 없는 폴더명은 mlflow 의 run-id 를 의미합니다.)

# 아무 디렉토리에나 들어가보겠습니다.
cd <특정 디렉토리>
ls

# artifacts, metrics, params, tag 와 같은 디렉토리가 있고 그 안에 실제 mlflow run 의 메타 정보가 저장된 것을 확인할 수 있습니다.
```