

추가과제1. MIFlow 설치 및 실행

1. docker 프로그램 설치

> <https://www.docker.com/>

2. pyspark-notebook docker image를 다운로드 및 실행

```
> docker run -p 5000:5000 -v mydir/myfolder:/home/jovyan/work/ jupyter/pyspark-notebook:latest
```

* 여기서 mydir/myfolder는 code와 data 폴더(mlflow_example.zip)가 있는 경로여야함.

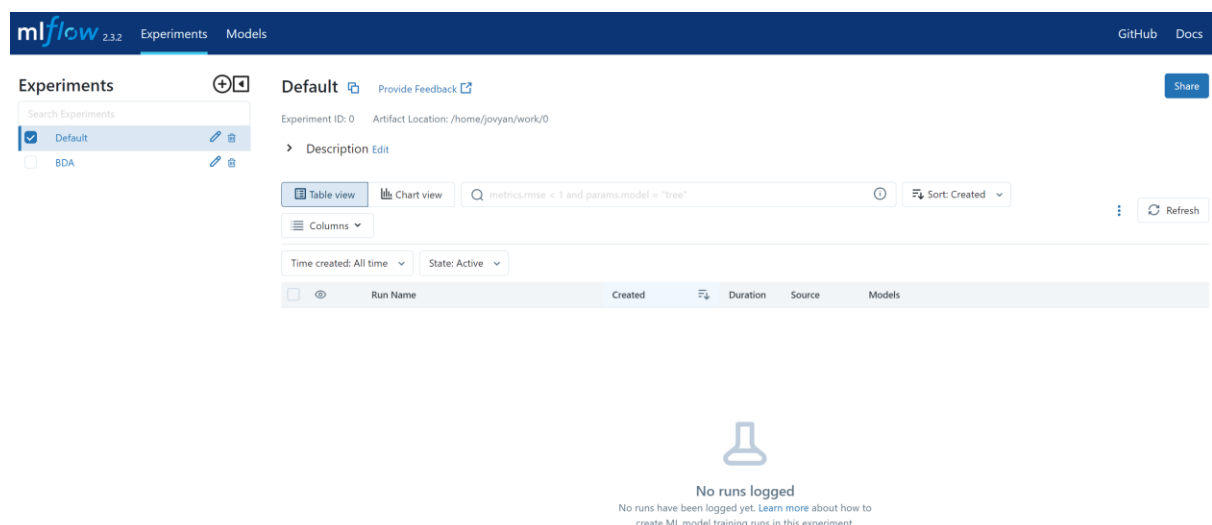
3. 해당 도커 터미널에서 다음 명령어를 통해 mlflow 설치

```
> conda install -c conda-forge mlflow
```

4. mlflow를 설치한뒤에 다음 명령어를 통해서 mlflow 서버를 실행

```
> mlflow server --backend-store-uri sqlite:///mlflow.db --default-artifact-root /home/jovyan/work -host 0.0.0.0 --port 5000 &
```

5. 웹브라우저에서 localhost:5000 입력하여 Mlflow가 정상적으로 뜨는지를 확인



6. terminal 창에서 다음 코드를 입력하여 MLFLOW 내 실행이 추적될 수 있도록 설정

> export MLFLOW_TRACKING_URI=http://localhost:5000

7. 다음 명령어를 수행하여 랜덤 포레스트 분류기를 여러 차례 실행(runs)

```
> spark-submit --master local[*] /home/jovyan/work/code/mlflow_example.py 32 5
> spark-submit --master local[*] /home/jovyan/work/code/mlflow_example.py 16 3
> spark-submit --master local[*] /home/jovyan/work/code/mlflow_example.py 16 5
> spark-submit --master local[*] /home/jovyan/work/code/mlflow_example.py 32 10
```

8. mlflow ui에 다시 접속하여 Experiments에 수행된 내용을 확인하고, 모델을 등록(Register Model)

< 모델 선택후, 우측 하단에 register model 선택 >

수행한 모델에는 ID값이 할당되어 있음 (runs:/id)

The screenshot displays the MLflow web interface. At the top, there's a navigation bar with 'mlflow 2.3.2', 'Experiments', and 'Models' tabs. Below this, the 'Experiments' section shows an experiment named 'resilient-colt-646'. The experiment details include: Run ID: 1c408f504de148209598df7be97f31ba, Date: 2023-05-23 16:45:45, Source: mlflow_example.py, User: jovyan, Duration: 19.6s, and Status: FINISHED. The 'Lifecycle Stage' is 'active'. On the left, there's a sidebar with 'Artifacts' expanded, showing a folder 'spark-model' containing files: sparkml, MLmodel, conda.yaml, python_env.yaml, and requirements.txt. The main content area shows the 'Full Path' of the artifact and a 'Register Model' button. Below this, the 'MLflow Model' section provides instructions on how to use the model. It includes a 'Model schema' table with columns 'Name' and 'Type', and a 'Make Predictions' code snippet showing how to load the model and make predictions.

resilient-colt-646

Run ID: 1c408f504de148209598df7be97f31ba Date: 2023-05-23 16:45:45 Source: mlflow_example.py

User: jovyan Duration: 19.6s Status: FINISHED

Lifecycle Stage: active

> Description Edit

> Parameters (2)

> Metrics (1)

> Tags

▼ Artifacts

Full Path: /home/jovyan/work/1/1c408f504de148209598df7be97f31ba/artifacts/spark-model

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 MLflow docs for how to	

Make Predictions

```
import mlflow
logged_model = 'runs:/1c408f504de148209598df7be97f31ba/spark-model'

# Load model
loaded_model = mlflow.spark.load_model(logged_model)
```

<div> <div>mlflow2.3.2</div> <div>Experiments</div> <div>Models</div> </div> <div> GitHub Docs </div>					
Registered Models					
<div> <div>Share and manage machine learning models. Learn more</div> <div> <div>Create Model</div> <div> <div>Search by model names or tags</div> <div>Search</div> <div>Clear</div> </div> </div> </div>					
Name	Latest Version	Staging	Production	Last Modified	Tags
BDA	Version 1	-	-	2023-05-23 16:49:10	-

7. terminal에서, 모델 폴더(아래 빨간색 부분)를 확인하여 다음 명령어를 실행

```
> spark-submit --master local[*] predict_spark.py
/home/jovyan/work/1/1c408f504de148209598df7be97f31ba/artifcats/spark-model
```

```
23/05/23 07:53:56 INFO CodeGenerator: Code generated in 9.91281 ms
+-----+-----+-----+-----+
|rawPrediction|probability|label|features|
+-----+-----+-----+-----+
|[18.36831020301444,1.6316897969855622]| [0.9184155101507219,0.0815844898492781]| 0 |[19.0,626.0,15.0,117.0,1.0,-1.0,0.0]|
|[18.36831020301444,1.6316897969855622]| [0.9184155101507219,0.0815844898492781]| 0 |[19.0,5368.0,4.0,77.0,6.0,-1.0,0.0]|
|[15.143686586236463,4.856313413763536]| [0.7571843293118231,0.24281567068817683]| 1 |[20.0,76.0,18.0,639.0,2.0,-1.0,0.0]|
|[18.36831020301444,1.6316897969855622]| [0.9184155101507219,0.0815844898492781]| 0 |[20.0,67.0,19.0,387.0,1.0,-1.0,0.0]|
|[18.36831020301444,1.6316897969855622]| [0.9184155101507219,0.0815844898492781]| 0 |[20.0,130.0,4.0,75.0,3.0,-1.0,0.0]|
+-----+-----+-----+-----+
only showing top 5 rows
```

과제 제출 내용:

0. 학번 / 이름

1. (mlflow_example.py) Mlflow 적어도 5개의 runs를 수행한 결과화면 (5개를 초과/미만해서 안되며, 삭제해야함)

2. (predict_spark.py) 7의 그림과 같이 예측된 값에 대해서 출력화면

주의사항:

- 프로그램 오류는 개별적으로 해결해야 함.
- 타 학생의 스크린샷을 사용하는 경우, 이유를 불문하고 0 점 처리함