

<http://bit.ly/sagemaker11>



# Amazon SageMaker Hands-on Guidebook

**6th November 2018**

강지양 SA / 김태현 SA



## Table of Contents

Lab 개요 .....	3
목표 .....	3
준비 조건 .....	3
LAB 실습 가이드.....	4
Module 1: Notebook Instance 생성하기 .....	5
Module 2: 실습용 코드 다운 받기.....	10
Module 3: Factorization Machine 을 이용한 영화 추천 서비스 Notebook .....	12
Module 4: TensorFlow 를 활용한 분산 학습 Notebook .....	14
Module 5: TensorFlow 를 활용한 분산 학습의 자동 모델 튜닝 Notebook.....	15
Module 6: 자동 모델 튜닝 결과 분석 Notebook.....	18
서비스 종료 가이드 .....	19

## Lab 개요

Amazon SageMaker 는 데이터 사이언티스트와 개발자들이 쉽고 빠르게 구성, 학습하고 어떤 규모로든 기계 학습된 모델을 배포할 수 있도록 해주는 관리형 서비스 입니다. 이 워크샵을 통해 SageMaker notebook instance 를 생성하고 샘플 Jupyter notebook 을 실습하면서 SageMaker 의 일부 기능을 알아보도록 합니다.

## 목표

- SageMaker 에 내장된 학습 기능을 사용하여 모델 훈련 Job 을 생성 합니다.
- SageMaker 의 endpoint 기능을 사용하여 생성된 모델이 예측에 사용될 수 있도록 endpoint 를 생성합니다.
- 머신 러닝이 정형 데이터(e.g. CSV 파일)와 비정형 데이터(e.g. 이미지)에 모두 적용 될수 있음을 확인 합니다.

## 준비 조건

- AWS 계정: AWS IAM, S3, SageMaker 자원을 생성할 수 있는 권한이 필요합니다.
- AWS Region: SageMaker 는 지원되는 region 은 <https://aws.amazon.com/about-aws/global-infrastructure/regional-product-services/> 에서 확인하실 수 있습니다. 이번 실습은 **Seoul (ap-northeast-2) region** 에서 실행 합니다.
- Browser: 최신 버전의 **Chrome, Firefox** 를 사용하세요.

※ 주의 사항: Notebook 안의 Cell 에서 코드 실행후 결과 값이 나오는 데는 수 초가 걸립니다. 훈련 Job 을 실행하는 경우 수 분이 걸릴 수도 있습니다. 실습 완료 후에는 아래 가이드에 따라 생성된 자원을 꼭 종료/삭제해 주세요.

## LAB 실습 가이드

실습은 총 6 개 모듈로 구성되어 있습니다. 1 번 완료후 2 번을 순서대로 진행하셔야 합니다. 3 번, 4 번, 5 번 모듈은 원하는 순서대로 진행하실 수 있습니다. 6 번 모듈은 반드시 5 번 모듈 완료 후에 실행하시기 바랍니다.

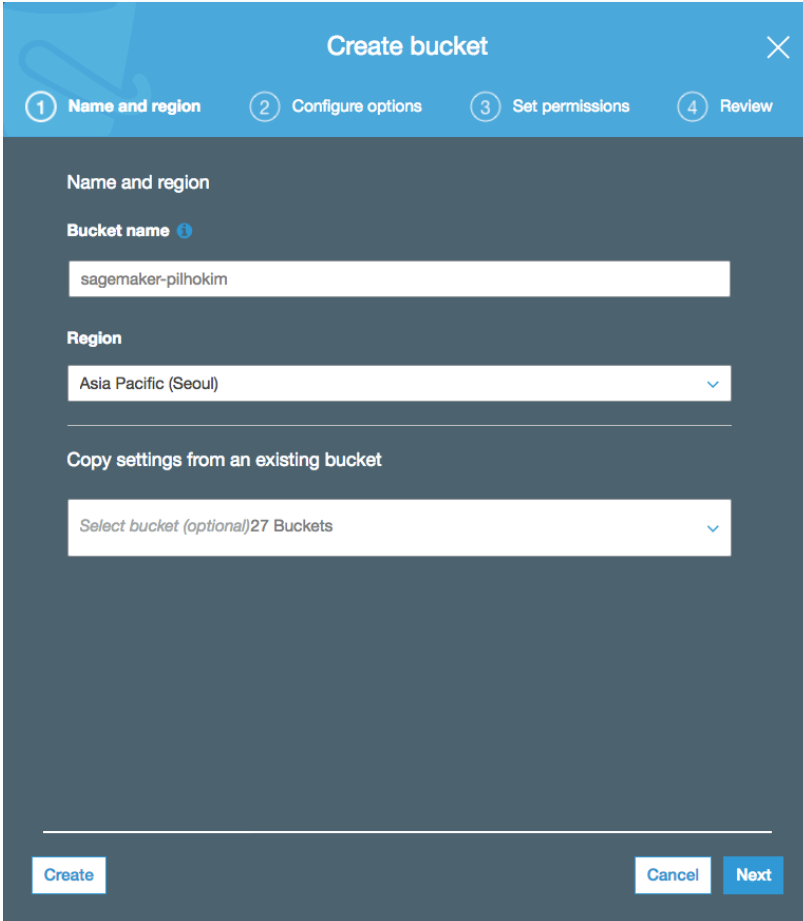
1. Notebook Instance 생성하기
2. 실습용 코드 다운받기
3. Factorization Machine 을 이용한 영화 추천 서비스 Notebook
4. TensorFlow 를 활용한 분산 학습 Notebook
5. TensorFlow 를 활용한 분산 학습의 자동 모델 튜닝 Notebook
6. 자동 모델 튜닝 결과 분석 Notebook

## Module 1: Notebook Instance 생성하기

### 1. S3 Bucket 생성하기

SageMaker 는 S3 를 데이터와 모델 저장소로 사용합니다. 여기서는 해당 목적으로 S3 Bucket 을 생성합니다.

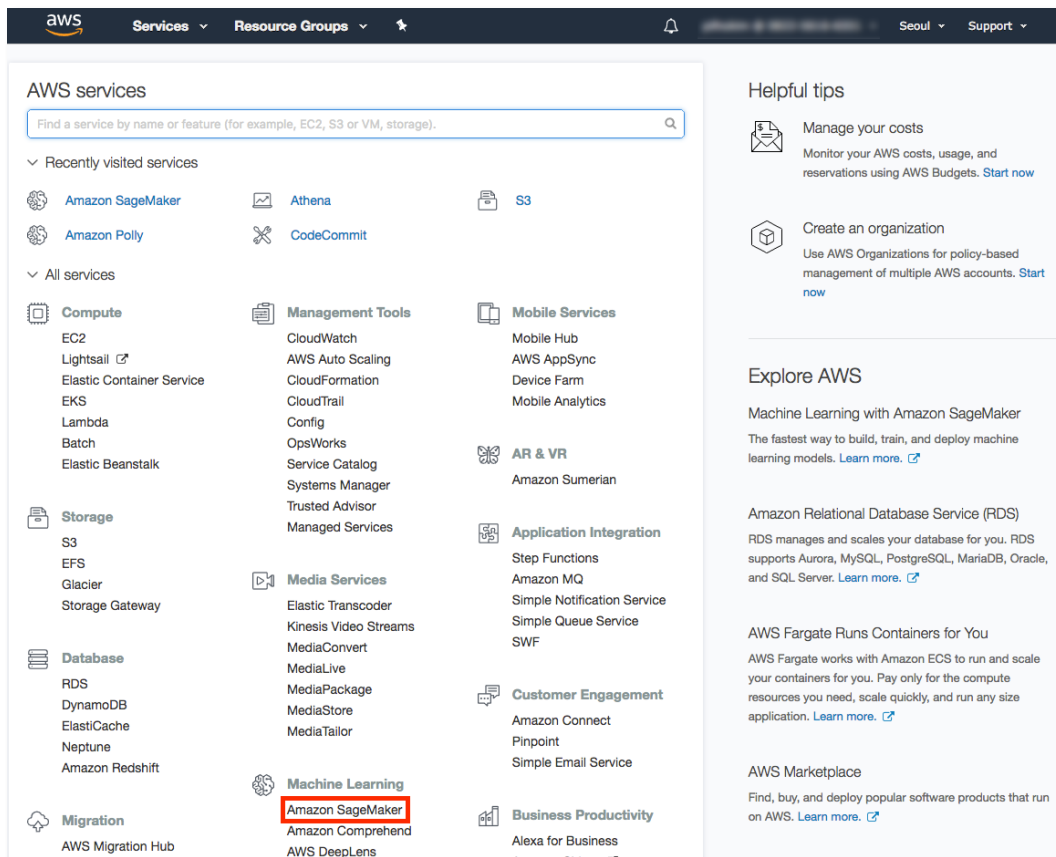
- 1) AWS 관리 콘솔 (<https://console.aws.amazon.com/>)에 Sign in 합니다.
- 2) AWS Services 리스트에서 S3 로 이동합니다.
- 3) "+ Create Bucket" 버튼을 선택합니다.
- 4) 아래 내용 설정 후 화면 왼쪽 아래 **Create** 클릭합니다.
  - Bucket name: sagemaker-{userid} [반드시 고유한 값 설정]
  - Region : Asia Pacific (Seoul)



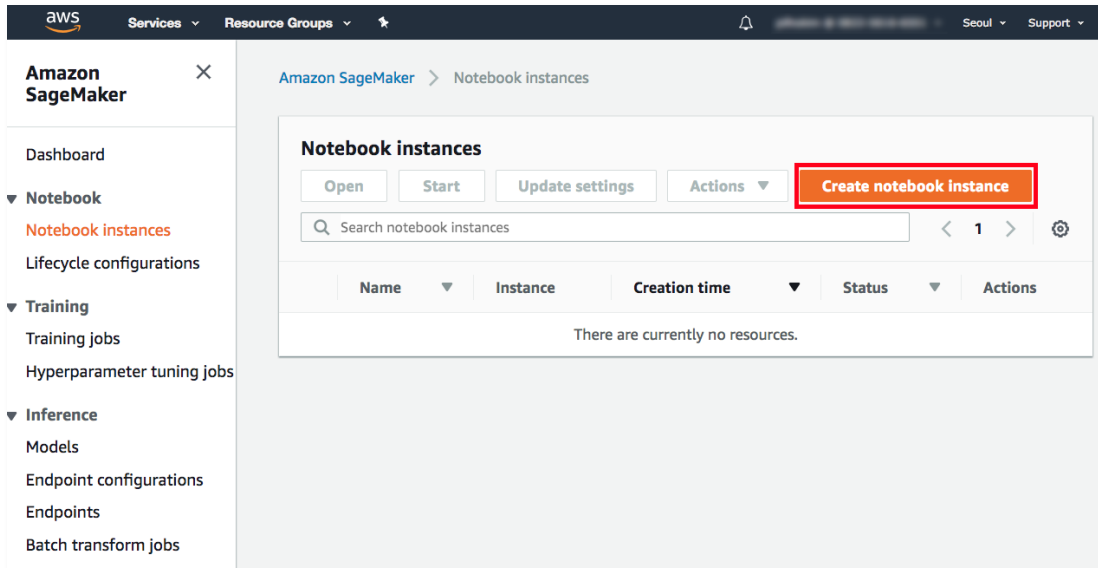
The screenshot shows the 'Create bucket' wizard in the AWS console. The wizard has four steps: 1. Name and region, 2. Configure options, 3. Set permissions, and 4. Review. The first step is currently active. In this step, the 'Bucket name' field contains 'sagemaker-pilhokim' and the 'Region' dropdown is set to 'Asia Pacific (Seoul)'. There is also a 'Copy settings from an existing bucket' section with a dropdown menu showing 'Select bucket (optional) 27 Buckets'. At the bottom of the wizard, there are three buttons: 'Create', 'Cancel', and 'Next'.

## 2. Notebook instance 생성

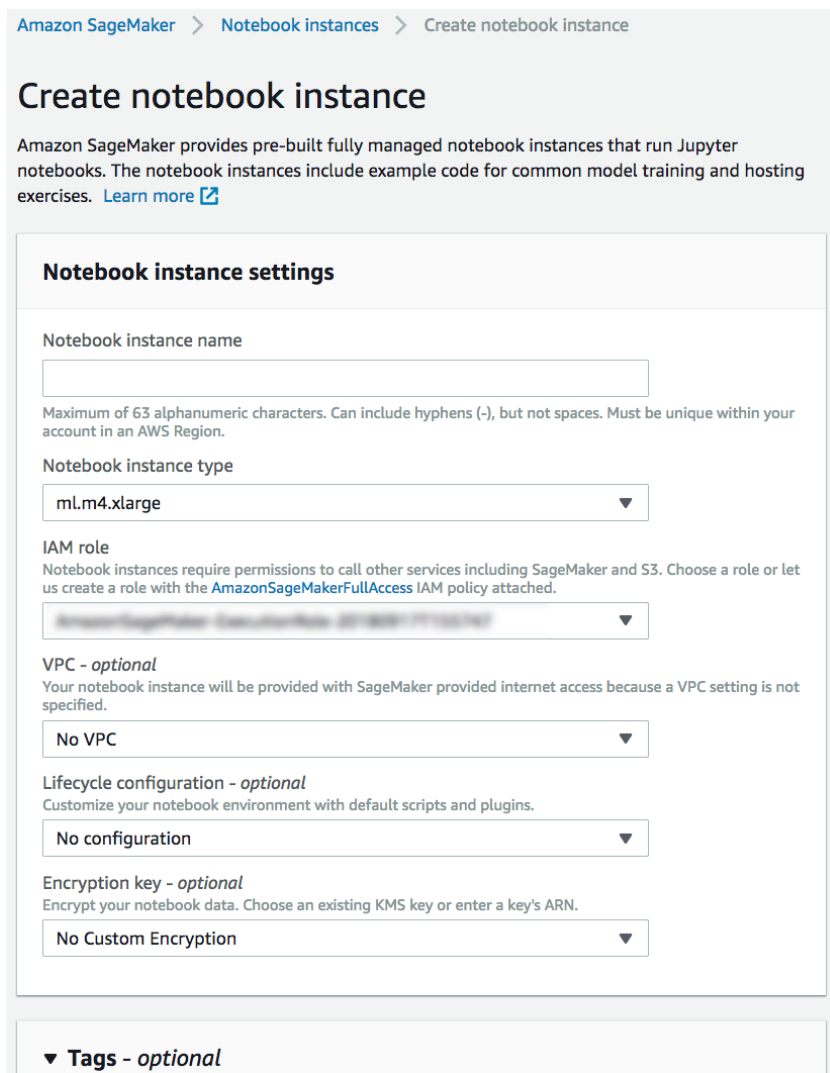
1) AWS 관리 콘솔에서 오른쪽 상단에서 Seoul Region 선택 후 AWS Services 리스트에서 Amazon SageMaker 서비스를 선택합니다.



2) 새로운 Notebook instance 를 생성하기 위해 왼쪽 패널 메뉴 중 **Notebook Instances** 선택 후 오른쪽 상단의 **Create notebook instance** 버튼을 클릭 합니다.



3) Notebook instance 이름으로 **[First Name]-[Last Name]-workshop** 으로 넣은 뒤 **ml.m4.xlarge** 인스턴스 타입을 선택 합니다.



4) IAM role 은 **Create a new role** 을 선택하고, 생성된 팝업 창에서는 **S3 buckets you specify – optional** 밑에 **Specific S3 Bucket** 을 선택 합니다. 그리고 텍스트 필드에 위에서 만든 S3 bucket 이름(예: sagemaker-xxxxx)을 선택 합니다. 이후 **Create role** 을 클릭합니다.

true"', and 'S3 bucket with a Bucket Policy allowing access to SageMaker'. At the bottom, there are 'Cancel' and 'Create role' buttons."/>

**Create an IAM role**

Passing an IAM role gives Amazon SageMaker permission to perform actions in other AWS services on your behalf. Creating a role here will grant permissions described by the [AmazonSageMakerFullAccess](#) IAM policy to the role you create.

The IAM role you create will provide access to:

- S3 buckets you specify - *optional*
  - Specific S3 buckets
 

Example: bucket-name-1, bucket-name-2, bi

Comma delimited. ARNs, "\*" and "/" are not supported.
  - Any S3 bucket
 

Allow users that have access to your notebook instance access to any bucket and its contents in your account.
  - None
- Any S3 bucket with "sagemaker" in the name
- Any S3 object with "sagemaker" in the name
- Any S3 object with the tag "sagemaker" and value "true" [See Object tagging](#)
- S3 bucket with a Bucket Policy allowing access to SageMaker [See S3 bucket policies](#)

5) 다시 Create Notebook instance 페이지로 돌아온 뒤 **Create notebook instance** 를 클릭합니다.

### 3. Notebook Instance 접근하기

1) 서버 상태가 **InService** 로 바뀔 때까지 기다립니다. 보통 5 분정도의 시간이 소요 됩니다.

Amazon SageMaker > Notebook instances

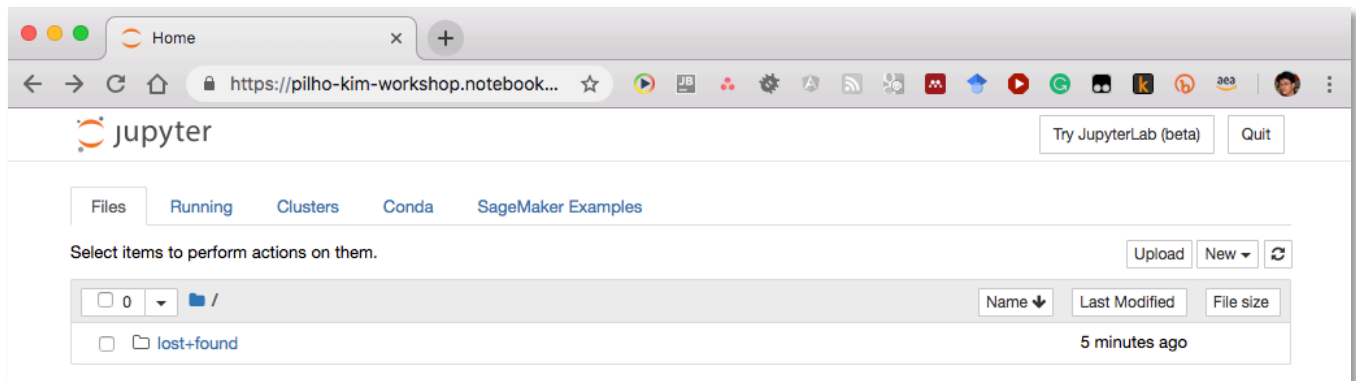
**Notebook instances**

Search notebook instances < 1 > ⚙

	Name	Instance	Creation time	Status	Actions
<input type="radio"/>	pilho-kim-workshop	ml.m4.xlarge	Sep 17, 2018 08:24 UTC	<input checked="" type="radio"/> InService	Open   Stop



2) Open 을 클릭하면 방금 생성한 notebook instance 의 Jupyter 홈페이지로 이동하게 됩니다.



## Module 2: 실습용 코드 다운 받기

SageMaker 의 Jupyter 노트북도 Linux 기반의 서버입니다. Jupyter 노트북에서 서버의 Terminal 을 바로 실행하는 기능을 제공하고 있습니다. Figure 1 와 같이 Terminal 을 선택합니다.

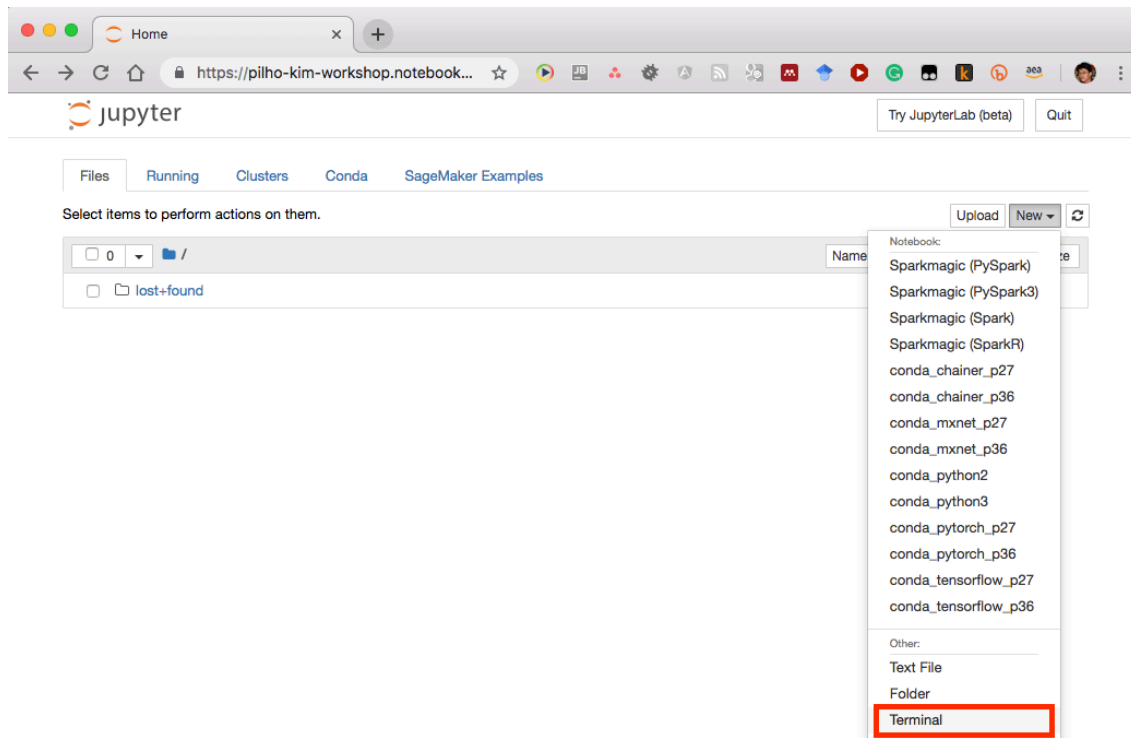


Figure 1. SageMaker 노트북 서버에 접속하기 위한 Terminal 실행 화면.

Figure 2 과 같이 터미널이 실행되면 아래의 명령어들을 입력해서 실행합니다.

```
cd SageMaker/
git clone https://github.com/aws-samples/aws-ai-ml-workshop-kr
```

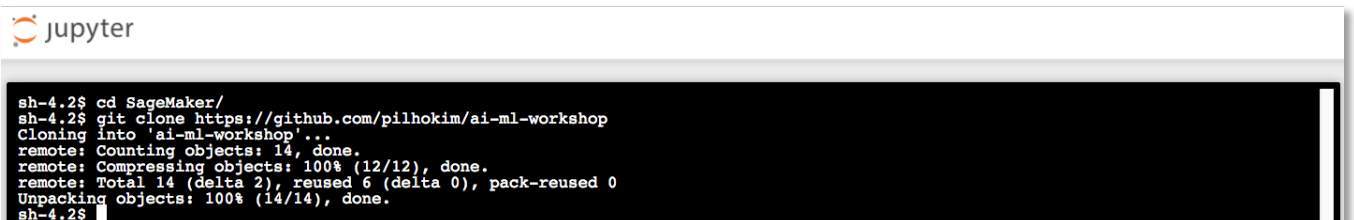


Figure 2. GitHub 에서 실습 코드 다운 받기.

코드를 다운 받고 난 후 Jupyter 노트북을 갱신 하면 (오른쪽 상단의 Refresh 아이콘을 클릭하세요) 새롭게 다운 받은 코드 폴더가 보입니다 (Figure 3).



Figure 3. 새로 다운 받은 폴더 생성 확인 화면.

폴더를 클릭해 `aws-ai-ml-workshop-kr/src/release/dev-day` 폴더를 찾습니다 (Figure 4). 이 폴더에서 다음 단계의 실습을 진행합니다.

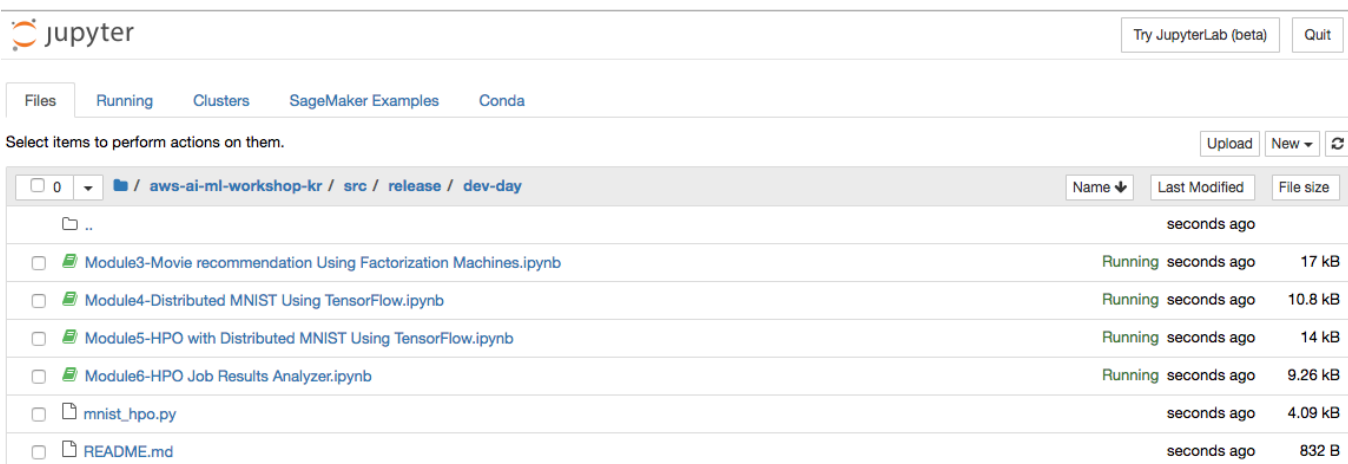


Figure 4. 실습 폴더 확인 화면.

### Module 3: Factorization Machine 을 이용한 영화 추천 서비스 Notebook

이 모듈에서는 Jupyter notebook 예제를 통해 어떻게 아마존이 제공하는 알고리즘을 SageMaker 에서 사용할 수 있는지 알아 봅니다. 특히 SageMaker 버전의 Factorization Machines 알고리즘을 사용하게 되는데, Factorization Machines 는 2010 년에 소개된 Supervised Machine Learning 알고리즘으로 행렬 인수분해(Matrix Factorization)과 SVM(Support Vector Machine) 개념을 결합한 알고리즘입니다. Factorization Machines 알고리즘은 분류 또는 회귀에 사용할 수 있으며 선형 회귀와 같은 알고리즘보다는 고차원 희소 데이터 셋을 처리하는데 훨씬 효율적인 알고리즘 입니다. Amazon SageMaker 의 Factorization Machines 알고리즘은 확장성이 높고 배포된 인스턴스에 걸쳐 교육이 가능하고 희소 및 밀집 데이터 세트에 대해 CPU 인스턴스를 사용한 교육 및 추론을 권장합니다.

이 모듈에서는 Factorization Machines 에 대한 소개와 이를 이용한 영화 추천 서비스를 만드는 과정입니다. 본 실습을 통해:

- Factorization Machines 알고리즘을 위한 데이터 준비 과정
- Factorization Machines 모델 훈련 및 배포

을 다루게 됩니다.

실습을 위해서는 현재 설치되어 있는 SageMaker 의 Jupyter 노트북의 예제들 중 아래의 디렉토리에 위한 Jupyter 노트북을 실행하시면 됩니다.

**/aws-ai-ml-workshop-kr/src/release/dev-day/Module3-Movie recommendation Using Factorization Machines.ipynb**

모듈 실행중 아래 코드를 만나면 모듈 1, 2 에서 사용한 s3 버킷의 이름(예: sagemaker-xxxxx)을 넣고 실행합니다.

```
In [ ]: import sagemaker
import sagemaker.amazon.common as smac
from sagemaker import get_execution_role
from sagemaker.predictor import json_deserializer

import boto3, csv, io, json
import numpy as np
from scipy.sparse import lil_matrix
from collections import defaultdict

bucket = '<put your S3 bucket name>' # replace with an existing bucket if needed
prefix = 'sagemaker/rm-movieLens'
```



※ 이 모델을 훈련하는데는 약 10 분에서 15 분이 소요됩니다.

## Module 4: TensorFlow 를 활용한 분산 학습 Notebook

---

이 모듈에서는 [MNIST Database](#) 에서 손으로 쓴 숫자의 이미지 데이터를 활용하여 SageMaker 에서 어떻게 분산 학습을 실행하는지 배우게 됩니다. 이 모듈은 [TensorFlow MNIST Example](#) 에 기반한 Convolutional Neural Network model 을 활용합니다.

이 모듈에서는 데이터 전처리 작업과 모델 학습은 물론, SageMaker 의 hosted endpoint 생성과 훈련된 모델을 endpoint 에 실제로 적용 하기 위해 어떻게 Jupyter notebook 과 SageMaker Python SDK 를 사용하는지를 차례로 실습할 수 있습니다.

생성된 모델은 실제로 사용자가 그려 넣은 숫자가 무엇인지 예측합니다. 이 예제에서는 TensorFlow 를 사용해 자신의 코드를 가져와 실행하는 것 뿐만 아니라, SageMaker 에서 모델 학습을 위해 여러 대의 인스턴스 클러스터를 얼마나 쉽게 생성할 수 있는지 보여 줍니다.

실습을 위해서는 현재 설치되어 있는 SageMaker 의 Jupyter 노트북의 예제들 중 아래의 디렉토리에 위한 Jupyter 노트북을 실행하시면 됩니다.

```
/aws-ai-ml-workshop-kr/src/release/dev-day/Module4-Distributed MNIST  
Using TensorFlow.ipynb
```

※ 이 모델을 훈련하는데는 약 15 분에서 20 분 정도가 소요됩니다.

## Module 5: TensorFlow 를 활용한 분산 학습의 자동 모델 튜닝 Notebook

이 모듈에서는 앞 모듈의 MNIST 이미지 분류 예제를 기반으로 SageMaker 의 자동 모델 튜닝 기능을 실습합니다. 이 기능은 기계 학습 알고리즘의 최적의 하이퍼파라미터(Hyperparameter) 값을 베이지안 최적화 기법을 통해 찾아줍니다.

실습을 위해서 현재 설치되어 있는 SageMaker 의 Jupyter 노트북의 예제들 중 아래의 디렉토리에 위한 Jupyter 노트북을 실행하시면 됩니다.

```
/aws-ai-ml-workshop-kr/src/release/dev-day/Module5-HPO with Distributed MNIST Using TensorFlow.ipynb
```

이 모듈에서는 MNIST 이미지 분류 예제의 하이퍼파라미터 중에서 learning rate 값을 자동으로 튜닝하며, 효과적인 탐색을 위해 최대값과 최소값을 아래 그림과 같이 설정합니다.

```
In [8]: hyperparameter_ranges = {'learning_rate': ContinuousParameter(0.001, 0.02)}
```

베이지안 최적화 기법은 하이퍼파라미터를 변경하면서 미리 지정된 숫자만큼 실험을 반복하는 특징이 있습니다. 이번 모듈에서는 병렬로 3 개의 학습을 3 번, 즉 총 9 번의 실험을 시도하도록 아래와 같이 설정합니다.

```
In [10]: tuner = HyperparameterTuner(estimator,
                                     objective_metric_name,
                                     hyperparameter_ranges,
                                     metric_definitions,
                                     max_jobs=9,
                                     max_parallel_jobs=3,
                                     objective_type=objective_type)
```

하이퍼파라미터 튜닝 작업은 아래와 같은 코드로 실행하며, 실행하면 각 하이퍼파라미터 값에 대한 개별 학습이 백그라운드에서 시작됩니다.

```
In [12]: boto3.client('sagemaker').describe_hyper_parameter_tuning_job(
         HyperParameterTuningJobName=tuner.latest_tuning_job.job_name)['HyperParameterTuningJobStatus']
Out[12]: u'InProgress'
```

이 때, SageMaker 의 콘솔에서 새로운 하이퍼파라미터 튜닝 작업 (Hyperparameter tuning jobs)이 생성된 것을 확인할 수 있습니다. 다음 모듈을 위해 이 작업의 이름을 메모해 놓습니다.



**Hyperparameter tuning jobs** Add/Edit tags Create hyperparameter tuning job

Search hyperparameter tuning jobs

Name	Status	Training completed/total	Creation time	Duration
sagemaker-tensorflow-181015-0010	InProgress	6 / 9	Oct 15, 2018 00:10 UTC	11 minutes
sagemaker-tensorflow-181014-2316	Completed	9 / 9	Oct 14, 2018 23:16 UTC	13 minutes
sagemaker-tensorflow-181013-0658	Completed	9 / 9	Oct 13, 2018 06:58 UTC	14 minutes
sagemaker-tensorflow-181013-0655	Stopped	0 / 3	Oct 13, 2018 06:55 UTC	3 minutes

실험이 모두 끝나면 하이퍼파라미터 튜닝 작업의 이름을 클릭해 튜닝 결과를 확인합니다. 아래 그림에서는 learning\_rate 가 0.004928838215245632 가 최적의 값이며 이때의 loss 값은 0.0642523318529129 인 것을 확인할 수 있습니다.

Amazon SageMaker > Hyperparameter tuning jobs > sagemaker-tensorflow-181015-0010

### sagemaker-tensorflow-181015-0010

Stop tuning job

**Hyperparameter tuning job summary**

Name sagemaker-tensorflow-181015-0010	Status Completed	Approx. total training duration 13 minute(s)	Role ARN <a href="#">arn:aws:iam::637338777613:role/service-role/AmazonSageMaker-ExecutionRole-20180102T080922</a>
ARN arn:aws:sagemaker:us-east-1:637338777613:hyperparameter-tuning-job/sagemaker-tensorflow-181015-0010	Creation time Oct 15, 2018 00:10 UTC	Last modified time Oct 15, 2018 00:23 UTC	

Best training job | Training jobs | Job configuration | Hyperparameter configuration | Tags

**Best training job summary** Create model

Name <a href="#">sagemaker-tensorflow-181015-0010-008-e631a4b3</a>	Status Completed	Objective metric loss	Value 0.0642523318529129
---	---------------------	--------------------------	-----------------------------

**Best training job hyperparameters**

Search

Name	Type	Value
_tuning_objective_metric	FreeText	loss
checkpoint_path	FreeText	"s3://sagemaker-us-east-1-637338777613/DEMO-hpo-tensorflow-2018-10-15-00-10-15-934/checkpoints"
evaluation_steps	FreeText	100
learning_rate	Continuous	0.004928838215245632





※ 이 모델을 훈련하는데는 약 20 분이 소요됩니다.

## Module 6: 자동 모델 튜닝 결과 분석 Notebook

이 모듈에서는 앞에서 실행한 하이퍼파라미터 튜닝 작업의 결과를 해석하는 과정을 실습합니다. BokehJS와 pandas 라이브러리를 사용해 튜닝 결과를 Jupyter 노트북에서 테이블과 그래프 형태로 시각화해볼 수 있습니다.

실습을 위해서 현재 설치되어 있는 SageMaker 의 Jupyter 노트북의 예제들 중 아래의 디렉토리에 위한 Jupyter 노트북을 실행하시면 됩니다.

```
/aws-ai-ml-workshop-kr/src/release/dev-day/Module6-HPO Job Results Analyzer.ipynb
```

모듈의 첫 부분에서 아래의 코드를 만나면 앞 모듈에서 실행된 **하이퍼파라미터 튜닝 작업 (Hyperparameter tuning jobs)**의 이름을 따옴표 안에 넣습니다.

```
In [ ]: import boto3
import sagemaker
import os

region = boto3.Session().region_name
sage_client = boto3.Session().client('sagemaker')
tuning_job_name = 'YOUR-HYPERPARAMETER-TUNING-JOB-NAME'
```

실행 결과로 나오는 두 개의 그래프에서, 탐색된 하이퍼파라미터 값의 변화에 따른 loss 함수 값의 변화를 해석해 보시기 바랍니다.

※ 이 모듈의 실습에는 약 5 분이 소요됩니다.

## 서비스 종료 가이드

워크 샵 이후 발생 되는 비용을 방지하기 위해서 아래의 단계에 따라 모두 종료/삭제 해 주세요. 비용이 발생하더라도 실습하신 Internet-facing App 을 유지하고 싶으신 경우에는 아래의 Notebook instance 의 경우만 처리하시면 됩니다.

- **Notebook instance:**

1) 만약 향후 사용을 위해 인스턴스를 저장하고 싶다면 **stop** 을 하시면 됩니다. 이 경우 스토리지 비용은 발생합니다. 향후 다시 재가동 하시려면 Start button 을 클릭하면 됩니다.

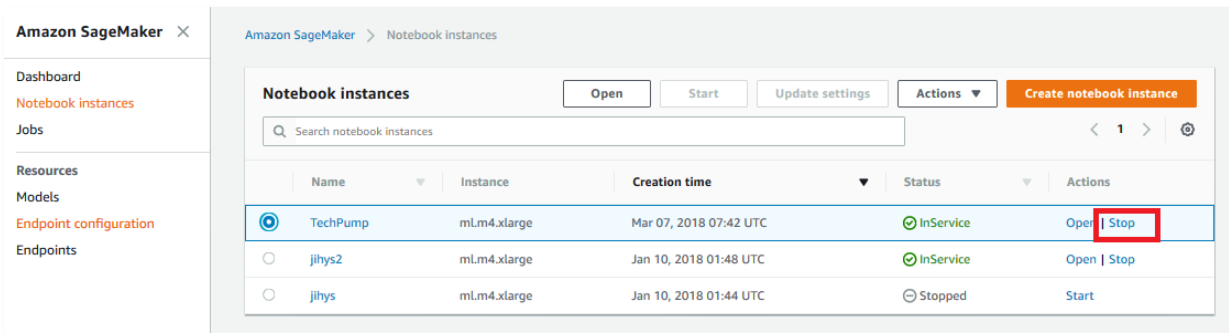


Figure 4. SageMaker 노트북 인스턴스 중단 화면.

2) 삭제를 할 경우는 **stop** 되어 있는 해당 notebook instance 를 선택하고 **Action** Dropdown 메뉴에서 **Delete** 선택 하시면 됩니다.

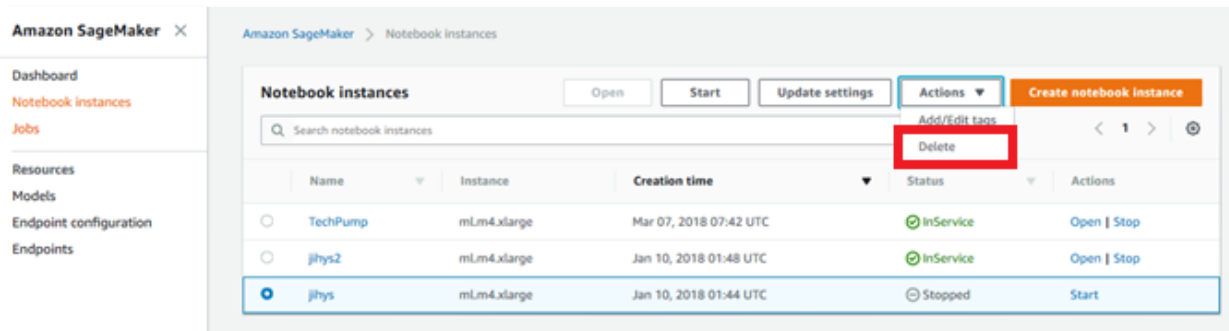


Figure 5. SageMaker 노트북 인스턴스 삭제 화면.

- **SageMaker Endpoints:**

훈련된 모델을 실제 예측 업무를 위해 배포된 한대 이상으로 구성된 클러스터입니다. Notebook 안에서 명령어로 삭제하거나 SageMaker console 에서 삭제 하실 수 있습니다. 삭제



하시기 위해서는 왼쪽 패널의 Endpoints 를 선택 하신 후 해당 endpoints 들 옆에 radio button 을 클릭 하신 후 Action Dropdown 메뉴에서 Delete 선택 하시면 됩니다.

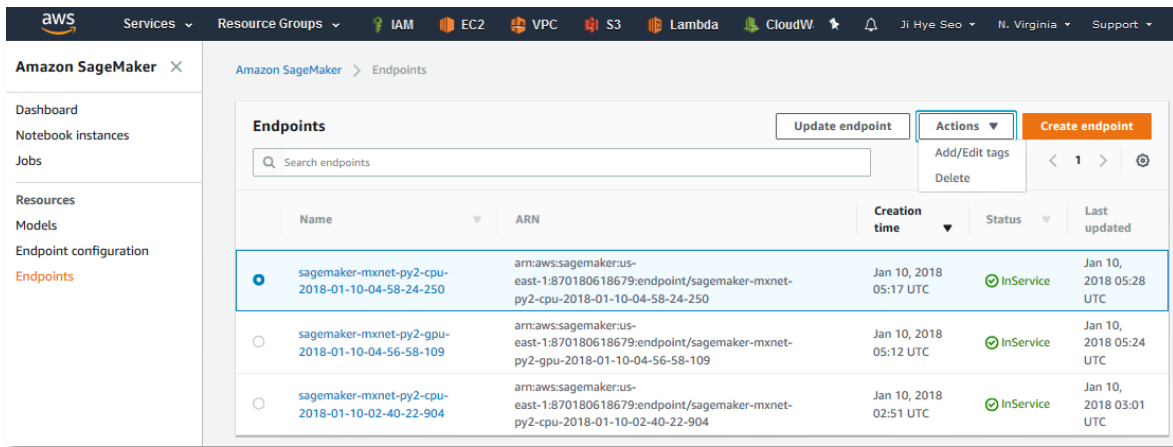


Figure 6. SageMaker Endpoint 삭제 화면.

- Lambda instance: 생성하신 Lambda instance 를 삭제합니다.

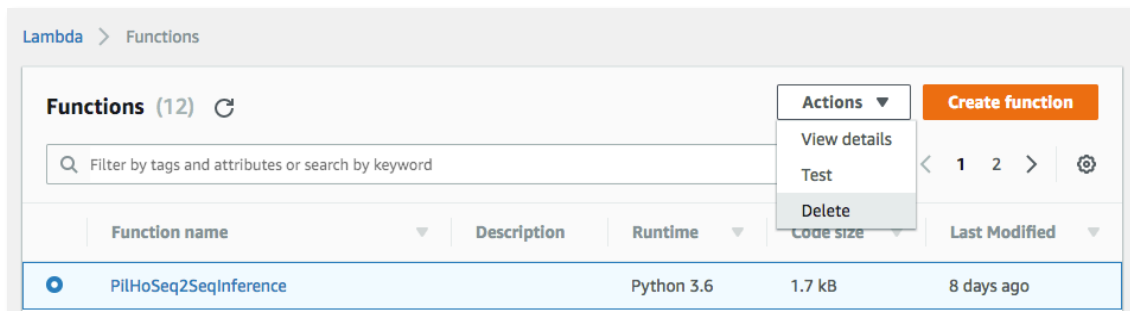


Figure 7. Lambda 인스턴스 삭제 화면.

- Amazon API Gateway instance: 생성하신 Gateway instance 를 삭제합니다.

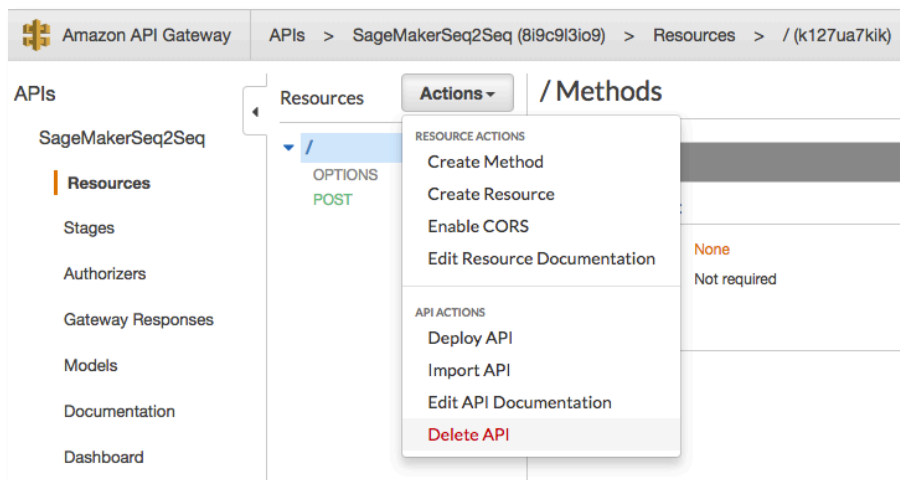


Figure 8. API Gateway 삭제 화면.

- Amazon S3 buckets: 생성하신 S3 Bucket (SageMaker 용, Public Internet 용)들을 모두 삭제합니다.

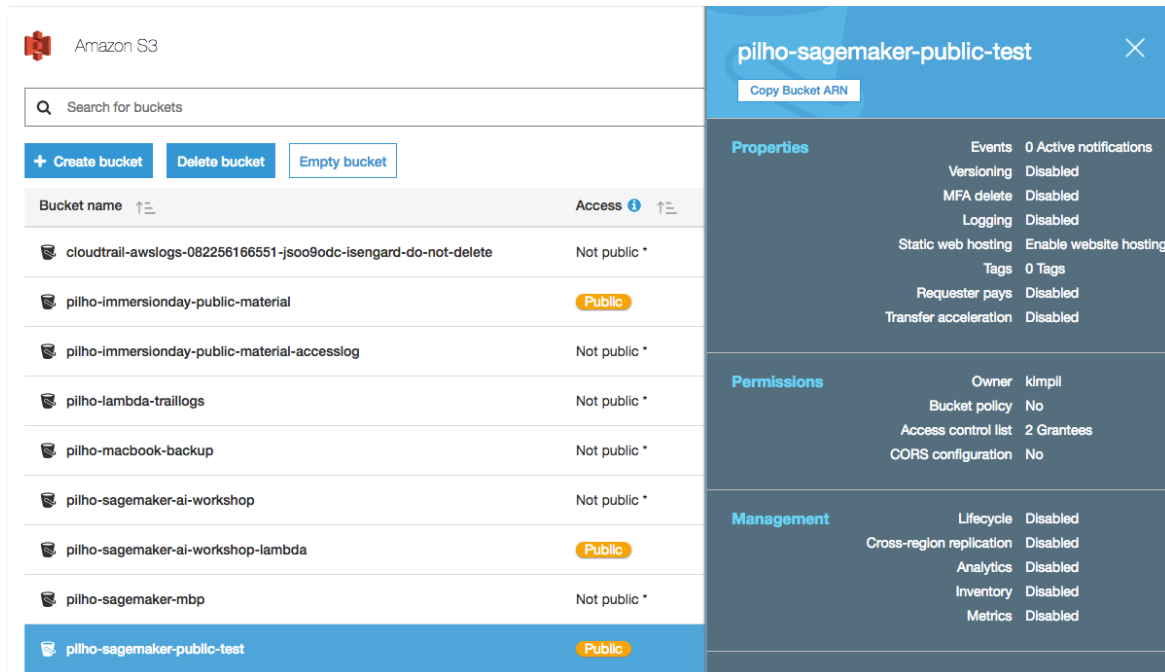


Figure 9. S3 버킷 삭제 화면.

이상으로 본 핸즈온 세션의 모든 과정을 마무리 하셨습니다. 수고하셨습니다.