XGBoost Built-in Algorithm - Iris Classification Example

In [1]:

```
import numpy as np
import pandas as pd

# Define IAM role
import boto3
import re
import sagemaker
from sagemaker import get_execution_role

# SageMaker SDK Documentation: http://sagemaker.readthedocs.io/en/latest/estimators.html
```

Upload Data to S3

In [2]:

```
bucket_name = 'dwb-ml-sagemaker'
training_file_key = 'iris/iris_train.csv'
validation_file_key = 'iris/iris_validation.csv'

s3_model_output_location = r's3://{0}/iris/model'.format(bucket_name)
s3_training_file_location = r's3://{0}/{1}'.format(bucket_name,training_file_key)
s3_validation_file_location = r's3://{0}/{1}'.format(bucket_name,validation_file_key)
```

In [3]:

```
print(s3_model_output_location)
print(s3_training_file_location)
print(s3_validation_file_location)
```

```
s3://dwb-ml-sagemaker/iris/model
s3://dwb-ml-sagemaker/iris/iris_train.csv
s3://dwb-ml-sagemaker/iris/iris_validation.csv
```

In [4]:

```
# Write and Reading from S3 is just as easy
# files are referred as objects in S3.
# file name is referred as key name in S3
# Files stored in S3 are automatically replicated across 3 different availability zones
# in the region where the bucket was created.
# http://boto3.readthedocs.io/en/latest/guide/s3.html
def write_to_s3(filename, bucket, key):
    with open(filename,'rb') as f: # Read in binary mode
        return boto3.Session().resource('s3').Bucket(bucket).Object(key).upload_fileobj(f)
```

In [5]:

```
write_to_s3('iris_train.csv',bucket_name,training_file_key)
write_to_s3('iris_validation.csv',bucket_name,validation_file_key)
```

Training Algorithm Docker Image

AWS Maintains a separate image for every region and algorithm

In [6]:

```
# Use Spot Instance - Save up to 90% of training cost by using spot instances when compared to on-demand instances
# Reference: https://github.com/aws-samples/amazon-sagemaker-managed-spot-training/blob/main/xgboost_built_in_managed

# if you are still on two-month free-tier you can use the on-demand instance by setting:
#   use_spot_instances = False

# We will use spot for training
#DWB#use_spot_instances = True
use_spot_instances = False
#DWB# -^- still on the free tier on 2023-07-28

max_run = 3600 # in seconds
max_wait = 7200 if use_spot_instances else None # in seconds

job_name = 'xgboost-iris-v1'

checkpoint_s3_uri = None

if use_spot_instances:
    checkpoint_s3_uri = f's3://{bucket_name}/iris/checkpoints/{job_name}'

print (f'Checkpoint uri: {checkpoint_s3_uri}')
```

```
Checkpoint uri: None
```

In [7]:

```
sess = sagemaker.Session()
```

In [8]:

```
role = get_execution_role()
```

In [9]:

```
# This role contains the permissions needed to train, deploy models
# SageMaker Service is trusted to assume this role
print(role)
```

```
arn:aws:iam::030021571292:role/service-role/AmazonSageMaker-ExecutionRole-20230702T224355
```

In [10]:

```
# https://sagemaker.readthedocs.io/en/stable/api/utility/image_uris.html#sagemaker.image_uris.retrieve

# SDK 2 uses image_uris.retrieve the container image location

# Use XGBoost 1.2 version
container = sagemaker.image_uris.retrieve("xgboost",sess.boto_region_name,version="1.2-2")

print (f'Using XGBoost Container {container}')
```

```
Using XGBoost Container 683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.2-2
```

Build Model

In [11]:

```
# Configure the training job
# Specify type and number of instances to use
# S3 location where final artifacts needs to be stored

#   Reference: http://sagemaker.readthedocs.io/en/latest/estimators.html

# for managed spot training, specify the use_spot_instances flag, max_run, max_wait and checkpoint_s3_uri

# SDK 2.x version does not require train prefix for instance count and type
estimator = sagemaker.estimator.Estimator(
    container,
    role,
    instance_count=1,
    instance_type='ml.m5.xlarge',
    output_path=s3_model_output_location,
    sagemaker_session=sess,
    base_job_name = job_name,
    use_spot_instances=use_spot_instances,
    max_run=max_run,
    max_wait=max_wait,
    checkpoint_s3_uri=checkpoint_s3_uri)
```

In [12]:

```
# Specify hyper parameters that appropriate for the training algorithm
# XGBoost Training Parameter Reference:
#   https://github.com/dmlc/xgboost/blob/master/doc/parameter.md
estimator.set_hyperparameters(max_depth=5,
                              objective="multi:softmax",
                              eval_metric="mlogloss",
                              num_class=3,
                              num_round=100,
                              early_stopping_rounds=10)
```

In [13]:

```
estimator.hyperparameters()
```

Out[13]:

```
{'max_depth': 5,
 'objective': 'multi:softmax',
 'eval_metric': 'mlogloss',
 'num_class': 3,
 'num_round': 100,
 'early_stopping_rounds': 10}
```

Specify Training Data Location and Optionally, Validation Data Location

In [14]:

```
# content type can be libsvm or csv for XGBoost
training_input_config = sagemaker.session.TrainingInput(
    s3_data=s3_training_file_location,
    content_type='csv',
    s3_data_type='S3Prefix')

validation_input_config = sagemaker.session.TrainingInput(
    s3_data=s3_validation_file_location,
    content_type='csv',
    s3_data_type='S3Prefix'
)

data_channels = {'train': training_input_config, 'validation': validation_input_config}
```

In [15]:

```
print(training_input_config.config)
print(validation_input_config.config)
```

```
{'DataSource': {'S3DataSource': {'S3DataType': 'S3Prefix', 'S3Uri': 's3://dwb-ml-sagemaker/iris/iris_train.csv', 'S3
DataDistributionType': 'FullyReplicated'}}, 'ContentType': 'csv'}
{'DataSource': {'S3DataSource': {'S3DataType': 'S3Prefix', 'S3Uri': 's3://dwb-ml-sagemaker/iris/iris_validation.cs
v', 'S3DataDistributionType': 'FullyReplicated'}}, 'ContentType': 'csv'}
```

Train the model

In [16]:

```
# XGBoost supports "train", "validation" channels
# Reference: Supported channels by algorithm
#   https://docs.aws.amazon.com/sagemaker/latest/dg/sagemaker-algo-docker-registry-paths.html
estimator.fit(data_channels)
```

```
INFO:sagemaker:Creating training-job with name: xgboost-iris-v1-2023-07-28-19-17-43-719
```

```
2023-07-28 19:17:43 Starting - Starting the training job...
2023-07-28 19:18:02 Starting - Preparing the instances for training......
2023-07-28 19:19:01 Downloading - Downloading input data...
2023-07-28 19:19:26 Training - Downloading the training image...
2023-07-28 19:20:12 Uploading - Uploading generated training model.[2023-07-28 19:20:07.975 ip-10-2-205-51.ec2.internal:7 INFO utils.py:27] RULE_JOB_STOP_SIGNAL_FILENAME: None
[2023-07-28:19:20:07:INFO] Imported framework sagemaker_xgboost_container.training
[2023-07-28:19:20:07:INFO] Failed to parse hyperparameter eval_metric value mlogloss to Json.
Returning the value itself
[2023-07-28:19:20:07:INFO] Failed to parse hyperparameter objective value multi:softmax to Json.
Returning the value itself
[2023-07-28:19:20:08:INFO] No GPUs detected (normal if no gpus installed)
[2023-07-28:19:20:08:INFO] Running XGBoost Sagemaker in algorithm mode
[2023-07-28:19:20:08:INFO] Determined delimiter of CSV input is ','
[2023-07-28:19:20:08:INFO] Determined delimiter of CSV input is ','
[2023-07-28:19:20:08:INFO] Determined delimiter of CSV input is ','
[2023-07-28:19:20:08:INFO] Determined delimiter of CSV input is ','
[2023-07-28:19:20:08:INFO] Single node training.
[2023-07-28:19:20:08:INFO] Train matrix has 105 rows and 4 columns
[2023-07-28:19:20:08:INFO] Validation matrix has 45 rows
[2023-07-28 19:20:08.049 ip-10-2-205-51.ec2.internal:7 INFO json_config.py:91] Creating hook from json_config at /opt/ml/input/config/debughookconfig.json.
[2023-07-28 19:20:08.050 ip-10-2-205-51.ec2.internal:7 INFO hook.py:201] tensorboard_dir has not been set for the hook. SMDebug will not be exporting tensorboard summaries.
[2023-07-28 19:20:08.050 ip-10-2-205-51.ec2.internal:7 INFO profiler_config_parser.py:102] User has disabled profiler.
[2023-07-28 19:20:08.051 ip-10-2-205-51.ec2.internal:7 INFO hook.py:255] Saving to /opt/ml/output/tensors
[2023-07-28 19:20:08.051 ip-10-2-205-51.ec2.internal:7 INFO state_store.py:77] The checkpoint config file /opt/ml/input/config/checkpointconfig.json does not exist.
[2023-07-28:19:20:08:INFO] Debug hook created from config
[0]#011train-mlogloss:0.73876#011validation-mlogloss:0.74994
[2023-07-28 19:20:08.053 ip-10-2-205-51.ec2.internal:7 INFO hook.py:423] Monitoring the collections: metrics
[2023-07-28 19:20:08.055 ip-10-2-205-51.ec2.internal:7 INFO hook.py:486] Hook is writing from the hook with pid: 7
[1]#011train-mlogloss:0.52787#011validation-mlogloss:0.55401
[2]#011train-mlogloss:0.38960#011validation-mlogloss:0.42612
[3]#011train-mlogloss:0.29429#011validation-mlogloss:0.34328
[4]#011train-mlogloss:0.22736#011validation-mlogloss:0.29000
[5]#011train-mlogloss:0.17920#011validation-mlogloss:0.24961
[6]#011train-mlogloss:0.14403#011validation-mlogloss:0.22234
[7]#011train-mlogloss:0.11664#011validation-mlogloss:0.20338
[8]#011train-mlogloss:0.09668#011validation-mlogloss:0.18999
[9]#011train-mlogloss:0.08128#011validation-mlogloss:0.18190
```

```
[10]#011train-mlogloss:0.06783#011validation-mlogloss:0.17996
[11]#011train-mlogloss:0.05794#011validation-mlogloss:0.18029
[12]#011train-mlogloss:0.05011#011validation-mlogloss:0.18306
[13]#011train-mlogloss:0.04428#011validation-mlogloss:0.18471
[14]#011train-mlogloss:0.03993#011validation-mlogloss:0.18693
[15]#011train-mlogloss:0.03615#011validation-mlogloss:0.18553
[16]#011train-mlogloss:0.03310#011validation-mlogloss:0.18571
[17]#011train-mlogloss:0.03065#011validation-mlogloss:0.18615
[18]#011train-mlogloss:0.02874#011validation-mlogloss:0.18930
[19]#011train-mlogloss:0.02739#011validation-mlogloss:0.18989
[20]#011train-mlogloss:0.02639#011validation-mlogloss:0.19251

2023-07-28 19:20:23 Completed - Training job completed
Training seconds: 83
Billable seconds: 83
```

Deploy Model

In [17]:

```
# Ref: http://sagemaker.readthedocs.io/en/latest/estimators.html
predictor = estimator.deploy(initial_instance_count=1,
                             instance_type='ml.m5.xlarge',
                             endpoint_name = job_name)
```

```
INFO:sagemaker:Creating model with name: xgboost-iris-v1-2023-07-28-19-33-37-255
INFO:sagemaker:Creating endpoint-config with name xgboost-iris-v1
INFO:sagemaker:Creating endpoint with name xgboost-iris-v1
----!
```

Run Predictions

In [18]:

```
# SDK 2.0 serializers
from sagemaker.serializers import CSVSerializer
predictor.serializer = CSVSerializer()
```

In [19]:

```
predictor.predict([[4.8,3.4,1.6,0.2],[4.8,3.4,1.6,0.2],[5.8,2.7,4.1,1.0]])
```

Out[19]:

```
b'0.0\n0.0\n1.0\n'
```

In [20]:

```
#DWB#  Good enough for me, but there's no Endpoint-deletion Code.
#DWB#+ I will put some in, here.
#DWB#+ As Chandra wrote with the previous such code
# Delete Endpoint to prevent unnecessary charges
#DWB# Not yet - one more notebook #predictor.delete_endpoint()
```

In [ ]:

```
```