Assignment 4

Following the first couple steps of the lab

In [ ]:
```
# Import necessary libraries - Running the lab
import os
import boto3
import sagemaker
import pandas as pd
import numpy as np
import re
from time import sleep

# Libraies related to model group
import time
from sagemaker import get_execution_role, session
from sagemaker.model_card import (
    ModelCard,
    ModelOverview,
    ObjectiveFunction,
    Function,
    TrainingDetails,
    IntendedUses,
    BusinessDetails,
    EvaluationJob,
    AdditionalInformation,
    Metric,
    MetricGroup,
    ModelCardStatusEnum,
    ObjectiveFunctionEnum,
    FacetEnum,
    RiskRatingEnum,
    MetricTypeEnum,
    EvaluationMetricTypeEnum,
)
```

In [ ]:
```
# Helper functions
def get_csv_output_from_s3(s3uri, batch_file):
```

```
    file_name = "{}.out".format(batch_file)
    match = re.match("s3://([^/]+)/(.*)", "{}/{}".format(s3uri, file_name))
    output_bucket, output_prefix = match.group(1), match.group(2)
    s3.download_file(output_bucket, output_prefix, file_name)
    return pd.read_csv(file_name, sep=",", header=None)
```

In [ ]:
```
# Define the variables that we need - will be using the same code in the lab demo for the dataset
role = sagemaker.get_execution_role()
sess = sagemaker.Session()
region = sess.boto_region_name

bucket = sess.default_bucket()
prefix = "DEMO-breast-cancer-prediction-xgboost-highlevel"
```

Getting the data

In [ ]:
```
# Initialize s3
s3 = boto3.client("s3")

filename = "wdbc.csv"
s3.download_file(
    f"sagemaker-example-files-prod-{region}", "datasets/tabular/breast_cancer/wdbc.csv", filename
)
data = pd.read_csv(filename, header=None)

# specify columns extracted from wbdc.names
data.columns = [
    "id",
    "diagnosis",
    "radius_mean",
    "texture_mean",
    "perimeter_mean",
    "area_mean",
    "smoothness_mean",
    "compactness_mean",
    "concavity_mean",
    "concave points_mean",
    "symmetry_mean",
    "fractal_dimension_mean",
    "radius_se",
```

```
    "texture_se",
    "perimeter_se",
    "area_se",
    "smoothness_se",
    "compactness_se",
    "concavity_se",
    "concave points_se",
    "symmetry_se",
    "fractal_dimension_se",
    "radius_worst",
    "texture_worst",
    "perimeter_worst",
    "area_worst",
    "smoothness_worst",
    "compactness_worst",
    "concavity_worst",
    "concave points_worst",
    "symmetry_worst",
    "fractal_dimension_worst",
]

# save the data
# data.to_csv("data/assignment_data.csv", sep=",", index=False)

data.sample(8)
```

Out[ ]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	con
522	91789	B	11.260	19.83	71.30	388.1	0.08511	0.04413	
314	894047	B	8.597	18.60	54.09	221.2	0.10740	0.05847	
63	859196	B	9.173	13.86	59.20	260.9	0.07721	0.08751	
553	924342	B	9.333	21.94	59.01	264.0	0.09240	0.05605	
133	867387	B	15.710	13.93	102.00	761.7	0.09462	0.09462	
269	8910720	B	10.710	20.39	69.50	344.9	0.10820	0.12890	
519	917080	B	12.750	16.70	82.51	493.8	0.11250	0.11170	
490	91376701	B	12.250	22.44	78.18	466.5	0.08192	0.05200	

8 rows × 32 columns

In [ ]:
```
# Replace M/B Diagnosis with bool values
data["diagnosis"] = data["diagnosis"].apply(lambda x: ((x == "M")) + 0)
data.sample(8)
```

Out[ ]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	con
557	925236	0	9.423	27.88	59.26	271.3	0.08123	0.04971	
11	84610002	1	15.780	17.89	103.60	781.0	0.09710	0.12920	
473	9113846	0	12.270	29.97	77.42	465.4	0.07699	0.03398	
433	908445	1	18.820	21.97	123.70	1110.0	0.10180	0.13890	
121	86517	1	18.660	17.12	121.40	1077.0	0.10540	0.11000	
241	883539	0	12.420	15.04	78.61	476.5	0.07926	0.03393	
139	868871	0	11.280	13.39	73.00	384.8	0.11640	0.11360	
262	888570	1	17.290	22.13	114.40	947.8	0.08999	0.12730	

8 rows × 32 columns

In [ ]:
```
# data split in three sets, training, validation and batch inference
rand_split = np.random.rand(len(data))
train_list = rand_split < 0.8
val_list = (rand_split >= 0.8) & (rand_split < 0.9)
batch_list = rand_split >= 0.9

data_train = data[train_list].drop(["id"], axis=1)
data_val = data[val_list].drop(["id"], axis=1)
data_batch = data[batch_list].drop(["diagnosis"], axis=1)
data_batch_noID = data_batch.drop(["id"], axis=1)
```

In [ ]:
```
train_file = "train_data.csv"
data_train.to_csv(train_file, index=False, header=False)
sess.upload_data(train_file, key_prefix="{}/train".format(prefix))

validation_file = "validation_data.csv"
data_val.to_csv(validation_file, index=False, header=False)
sess.upload_data(validation_file, key_prefix="{}/validation".format(prefix))
```

```
batch_file = "batch_data.csv"
data_batch.to_csv(batch_file, index=False, header=False)
sess.upload_data(batch_file, key_prefix="{}/batch".format(prefix))

batch_file_noID = "batch_data_noID.csv"
data_batch_noID.to_csv(batch_file_noID, index=False, header=False)
sess.upload_data(batch_file_noID, key_prefix="{}/batch".format(prefix))
```

Out[ ]: 's3://sagemaker-us-east-1-004608622582/DEMO-breast-cancer-prediction-xgboost-highlevel/batch/batch_data_noID.csv'

Training and creating the model

In [ ]:
```
%%time
from time import gmtime, strftime

job_name = "xgb-" + strftime("%Y-%m-%d-%H-%M-%S", gmtime())
output_location = "s3://{}/{}/output/{}".format(bucket, prefix, job_name)
image = sagemaker.image_uris.retrieve(
    framework="xgboost", region=boto3.Session().region_name, version="1.7-1"
)

sm_estimator = sagemaker.estimator.Estimator(
    image,
    role,
    instance_count=1,
    instance_type="ml.m5.xlarge",
    volume_size=50,
    input_mode="File",
    output_path=output_location,
    sagemaker_session=sess,
)

sm_estimator.set_hyperparameters(
    objective="binary:logistic",
    max_depth=5,
    eta=0.2,
    gamma=4,
    min_child_weight=6,
    subsample=0.8,
    verbosity=0,
    num_round=100,
```

```
)

train_data = sagemaker.inputs.TrainingInput(
    "s3://{}/{}/train".format(bucket, prefix),
    distribution="FullyReplicated",
    content_type="text/csv",
    s3_data_type="S3Prefix",
)
validation_data = sagemaker.inputs.TrainingInput(
    "s3://{}/{}/validation".format(bucket, prefix),
    distribution="FullyReplicated",
    content_type="text/csv",
    s3_data_type="S3Prefix",
)
data_channels = {"train": train_data, "validation": validation_data}

# Start training by calling the fit method in the estimator
sm_estimator.fit(inputs=data_channels, job_name=job_name, logs=True)
```

```
INFO:sagemaker:Creating training-job with name: xgb-2024-05-31-23-53-31
```

```
2024-05-31 23:53:31 Starting - Starting the training job...
2024-05-31 23:53:47 Starting - Preparing the instances for training...
2024-05-31 23:54:11 Downloading - Downloading input data...
2024-05-31 23:54:36 Downloading - Downloading the training image...
2024-05-31 23:55:32 Training - Training image download completed. Training in progress.
2024-05-31 23:55:32 Uploading - Uploading generated training model.[2024-05-31 23:55:27.555 ip-10-0-202-27.ec2.intern
al:7 INFO utils.py:28] RULE_JOB_STOP_SIGNAL_FILENAME: None
[2024-05-31 23:55:27.576 ip-10-0-202-27.ec2.internal:7 INFO profiler_config_parser.py:111] User has disabled profile
r.
[2024-05-31:23:55:27:INFO] Imported framework sagemaker_xgboost_container.training
[2024-05-31:23:55:27:INFO] Failed to parse hyperparameter objective value binary:logistic to Json.
Returning the value itself
[2024-05-31:23:55:27:INFO] No GPUs detected (normal if no gpus installed)
[2024-05-31:23:55:27:INFO] Running XGBoost Sagemaker in algorithm mode
[2024-05-31:23:55:27:INFO] Determined 0 GPU(s) available on the instance.
[2024-05-31:23:55:27:INFO] Determined delimiter of CSV input is ','
[2024-05-31:23:55:27:INFO] Determined delimiter of CSV input is ','
[2024-05-31:23:55:27:INFO] File path /opt/ml/input/data/train of input files
[2024-05-31:23:55:27:INFO] Making smlinks from folder /opt/ml/input/data/train to folder /tmp/sagemaker_xgboost_input
_data
[2024-05-31:23:55:27:INFO] creating symlink between Path /opt/ml/input/data/train/train_data.csv and destination /tm
p/sagemaker_xgboost_input_data/train_data.csv3534245250621421320
[2024-05-31:23:55:27:INFO] files path: /tmp/sagemaker_xgboost_input_data
[2024-05-31:23:55:27:INFO] Determined delimiter of CSV input is ','
[2024-05-31:23:55:27:INFO] File path /opt/ml/input/data/validation of input files
[2024-05-31:23:55:27:INFO] Making smlinks from folder /opt/ml/input/data/validation to folder /tmp/sagemaker_xgboost_
input_data
[2024-05-31:23:55:27:INFO] creating symlink between Path /opt/ml/input/data/validation/validation_data.csv and destin
ation /tmp/sagemaker_xgboost_input_data/validation_data.csv8746957019974615854
[2024-05-31:23:55:27:INFO] files path: /tmp/sagemaker_xgboost_input_data
[2024-05-31:23:55:27:INFO] Determined delimiter of CSV input is ','
[2024-05-31:23:55:27:INFO] Single node training.
[2024-05-31:23:55:27:INFO] Train matrix has 439 rows and 30 columns
[2024-05-31:23:55:27:INFO] Validation matrix has 60 rows
[2024-05-31 23:55:27.994 ip-10-0-202-27.ec2.internal:7 INFO json_config.py:92] Creating hook from json_config at /op
t/ml/input/config/debughookconfig.json.
[2024-05-31 23:55:27.994 ip-10-0-202-27.ec2.internal:7 INFO hook.py:206] tensorboard_dir has not been set for the hoo
k. SMDebug will not be exporting tensorboard summaries.
[2024-05-31 23:55:27.995 ip-10-0-202-27.ec2.internal:7 INFO hook.py:259] Saving to /opt/ml/output/tensors
[2024-05-31 23:55:27.995 ip-10-0-202-27.ec2.internal:7 INFO state_store.py:77] The checkpoint config file /opt/ml/inp
ut/config/checkpointconfig.json does not exist.
[2024-05-31:23:55:27:INFO] Debug hook created from config
```

```
[0]#011train-logloss:0.54830#011validation-logloss:0.55022
[2024-05-31 23:55:27.999 ip-10-0-202-27.ec2.internal:7 INFO hook.py:427] Monitoring the collections: metrics
[2024-05-31 23:55:28.002 ip-10-0-202-27.ec2.internal:7 INFO hook.py:491] Hook is writing from the hook with pid: 7
[1]#011train-logloss:0.44957#011validation-logloss:0.45006
[2]#011train-logloss:0.37565#011validation-logloss:0.38500
[3]#011train-logloss:0.31937#011validation-logloss:0.32985
[4]#011train-logloss:0.27509#011validation-logloss:0.29116
[5]#011train-logloss:0.24204#011validation-logloss:0.26259
[6]#011train-logloss:0.21612#011validation-logloss:0.23594
[7]#011train-logloss:0.19311#011validation-logloss:0.21821
[8]#011train-logloss:0.17403#011validation-logloss:0.19818
[9]#011train-logloss:0.16206#011validation-logloss:0.19086
[10]#011train-logloss:0.14814#011validation-logloss:0.18048
[11]#011train-logloss:0.14006#011validation-logloss:0.17669
[12]#011train-logloss:0.13197#011validation-logloss:0.16541
[13]#011train-logloss:0.12379#011validation-logloss:0.15893
[14]#011train-logloss:0.11835#011validation-logloss:0.15331
[15]#011train-logloss:0.11381#011validation-logloss:0.14720
[16]#011train-logloss:0.10977#011validation-logloss:0.14062
[17]#011train-logloss:0.10671#011validation-logloss:0.13604
[18]#011train-logloss:0.10368#011validation-logloss:0.13235
[19]#011train-logloss:0.09846#011validation-logloss:0.12386
[20]#011train-logloss:0.09589#011validation-logloss:0.12082
[21]#011train-logloss:0.09370#011validation-logloss:0.11415
[22]#011train-logloss:0.09107#011validation-logloss:0.11156
[23]#011train-logloss:0.09108#011validation-logloss:0.11155
[24]#011train-logloss:0.09108#011validation-logloss:0.11155
[25]#011train-logloss:0.08927#011validation-logloss:0.10875
[26]#011train-logloss:0.08740#011validation-logloss:0.10302
[27]#011train-logloss:0.08741#011validation-logloss:0.10298
[28]#011train-logloss:0.08539#011validation-logloss:0.10070
[29]#011train-logloss:0.08342#011validation-logloss:0.09738
[30]#011train-logloss:0.08197#011validation-logloss:0.09972
[31]#011train-logloss:0.08195#011validation-logloss:0.09974
[32]#011train-logloss:0.08195#011validation-logloss:0.09975
[33]#011train-logloss:0.08195#011validation-logloss:0.09976
[34]#011train-logloss:0.08195#011validation-logloss:0.09972
[35]#011train-logloss:0.08195#011validation-logloss:0.09975
[36]#011train-logloss:0.08195#011validation-logloss:0.09973
[37]#011train-logloss:0.08196#011validation-logloss:0.09976
[38]#011train-logloss:0.08196#011validation-logloss:0.09977
[39]#011train-logloss:0.08196#011validation-logloss:0.09978
```

```
[40]#011train-logloss:0.08198#011validation-logloss:0.09981
[41]#011train-logloss:0.08200#011validation-logloss:0.09986
[42]#011train-logloss:0.08198#011validation-logloss:0.09982
[43]#011train-logloss:0.08196#011validation-logloss:0.09977
[44]#011train-logloss:0.08195#011validation-logloss:0.09975
[45]#011train-logloss:0.08196#011validation-logloss:0.09978
[46]#011train-logloss:0.08195#011validation-logloss:0.09975
[47]#011train-logloss:0.08195#011validation-logloss:0.09975
[48]#011train-logloss:0.08200#011validation-logloss:0.09973
[49]#011train-logloss:0.08073#011validation-logloss:0.09764
[50]#011train-logloss:0.08070#011validation-logloss:0.09766
[51]#011train-logloss:0.08070#011validation-logloss:0.09770
[52]#011train-logloss:0.08071#011validation-logloss:0.09774
[53]#011train-logloss:0.08070#011validation-logloss:0.09770
[54]#011train-logloss:0.08070#011validation-logloss:0.09768
[55]#011train-logloss:0.08071#011validation-logloss:0.09772
[56]#011train-logloss:0.08073#011validation-logloss:0.09779
[57]#011train-logloss:0.08070#011validation-logloss:0.09767
[58]#011train-logloss:0.08071#011validation-logloss:0.09765
[59]#011train-logloss:0.08073#011validation-logloss:0.09764
[60]#011train-logloss:0.08072#011validation-logloss:0.09764
[61]#011train-logloss:0.08073#011validation-logloss:0.09764
[62]#011train-logloss:0.08075#011validation-logloss:0.09764
[63]#011train-logloss:0.08072#011validation-logloss:0.09764
[64]#011train-logloss:0.08072#011validation-logloss:0.09764
[65]#011train-logloss:0.08072#011validation-logloss:0.09764
[66]#011train-logloss:0.08073#011validation-logloss:0.09764
[67]#011train-logloss:0.08072#011validation-logloss:0.09764
[68]#011train-logloss:0.08074#011validation-logloss:0.09764
[69]#011train-logloss:0.08070#011validation-logloss:0.09768
[70]#011train-logloss:0.08070#011validation-logloss:0.09768
[71]#011train-logloss:0.08070#011validation-logloss:0.09766
[72]#011train-logloss:0.08072#011validation-logloss:0.09764
[73]#011train-logloss:0.08072#011validation-logloss:0.09764
[74]#011train-logloss:0.08071#011validation-logloss:0.09765
[75]#011train-logloss:0.08071#011validation-logloss:0.09765
[76]#011train-logloss:0.08070#011validation-logloss:0.09767
[77]#011train-logloss:0.07900#011validation-logloss:0.09589
[78]#011train-logloss:0.07900#011validation-logloss:0.09588
[79]#011train-logloss:0.07900#011validation-logloss:0.09589
[80]#011train-logloss:0.07900#011validation-logloss:0.09589
[81]#011train-logloss:0.07900#011validation-logloss:0.09589
```

```
[82]#011train-logloss:0.07900#011validation-logloss:0.09590
[83]#011train-logloss:0.07900#011validation-logloss:0.09590
[84]#011train-logloss:0.07900#011validation-logloss:0.09590
[85]#011train-logloss:0.07900#011validation-logloss:0.09592
[86]#011train-logloss:0.07901#011validation-logloss:0.09594
[87]#011train-logloss:0.07904#011validation-logloss:0.09599
[88]#011train-logloss:0.07903#011validation-logloss:0.09597
[89]#011train-logloss:0.07901#011validation-logloss:0.09594
[90]#011train-logloss:0.07904#011validation-logloss:0.09599
[91]#011train-logloss:0.07900#011validation-logloss:0.09591
[92]#011train-logloss:0.07900#011validation-logloss:0.09591
[93]#011train-logloss:0.07900#011validation-logloss:0.09592
[94]#011train-logloss:0.07900#011validation-logloss:0.09591
[95]#011train-logloss:0.07903#011validation-logloss:0.09596
[96]#011train-logloss:0.07899#011validation-logloss:0.09589
[97]#011train-logloss:0.07900#011validation-logloss:0.09588
[98]#011train-logloss:0.07900#011validation-logloss:0.09588
[99]#011train-logloss:0.07900#011validation-logloss:0.09588

2024-05-31 23:55:45 Completed - Training job completed
Training seconds: 94
Billable seconds: 94
CPU times: user 417 ms, sys: 15.5 ms, total: 432 ms
Wall time: 2min 42s
```

Create Batch Transform

In [ ]:

```
%%time

# Create transformer
sm_transformer = sm_estimator.transformer(1, "ml.m4.xlarge")

# start a transform job
input_location = "s3://{}/{}/batch/{}".format(
    bucket, prefix, batch_file_noID
)  # Use input data without ID column

sm_transformer.transform(input_location, content_type="text/csv", split_type="Line")

sm_transformer.wait()
```

```
INFO:sagemaker:Creating model with name: sagemaker-xgboost-2024-05-31-23-56-13-241
INFO:sagemaker:Creating transform job with name: sagemaker-xgboost-2024-05-31-23-56-13-876
```

```
.........................................[2024-06-01:00:03:19:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:19:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:19:INFO] nginx config:
worker_processes auto;
daemon off;
pid /tmp/nginx.pid;
error_log  /dev/stderr;
worker_rlimit_nofile 4096;
events {
  worker_connections 2048;
}
http {
  include /etc/nginx/mime.types;
  default_type application/octet-stream;
  access_log /dev/stdout combined;
  upstream gunicorn {
    server unix:/tmp/gunicorn.sock;
  }
  server {
    listen 8080 deferred;
    client_max_body_size 0;
    keepalive_timeout 3;
    location ~ ^/(ping|invocations|execution-parameters) {
      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header Host $http_host;
      proxy_redirect off;
      proxy_read_timeout 60s;
      proxy_pass http://gunicorn;
    }
    location / {
      return 404 "{}";
    }
  }
}
[2024-06-01 00:03:19 +0000] [19] [INFO] Starting gunicorn 19.10.0
[2024-06-01 00:03:19 +0000] [19] [INFO] Listening at: unix:/tmp/gunicorn.sock (19)
[2024-06-01 00:03:19 +0000] [19] [INFO] Using worker: gevent
/miniconda3/lib/python3.8/os.py:1023: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, th
e default buffer size will be used
  return io.open(fd, *args, **kwargs)
[2024-06-01 00:03:19 +0000] [25] [INFO] Booting worker with pid: 25
[2024-06-01 00:03:19 +0000] [26] [INFO] Booting worker with pid: 26
```

```
[2024-06-01 00:03:19 +0000] [27] [INFO] Booting worker with pid: 27
[2024-06-01 00:03:20 +0000] [28] [INFO] Booting worker with pid: 28
[2024-06-01:00:03:21:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:21:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:03:21:INFO] Model objective : binary:logistic
[2024-06-01:00:03:21:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:21:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:03:21:INFO] Model objective : binary:logistic
[2024-06-01:00:03:22:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:22:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:03:22:INFO] Model objective : binary:logistic
[2024-06-01:00:03:22:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:22:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:03:22:INFO] Model objective : binary:logistic
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "GET /ping HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "GET /ping HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "GET /execution-parameters HTTP/1.1" 200 84 "-" "Go-http-client/1.1"
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:25:INFO] Determined delimiter of CSV input is ','
/miniconda3/lib/python3.8/site-packages/xgboost/core.py:122: UserWarning: ntree_limit is deprecated, use `iteration_r
ange` or model slicing instead.
  warnings.warn(
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "POST /invocations HTTP/1.1" 200 1397 "-" "Go-http-client/1.1"
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "GET /execution-parameters HTTP/1.1" 200 84 "-" "Go-http-client/1.1"
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:25:INFO] Determined delimiter of CSV input is ','
/miniconda3/lib/python3.8/site-packages/xgboost/core.py:122: UserWarning: ntree_limit is deprecated, use `iteration_r
ange` or model slicing instead.
  warnings.warn(
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "POST /invocations HTTP/1.1" 200 1397 "-" "Go-http-client/1.1"
2024-06-01T00:03:25.680:[sagemaker logs]: MaxConcurrentTransforms=4, MaxPayloadInMB=6, BatchStrategy=MULTI_RECORD

[2024-06-01:00:03:19:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:19:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:19:INFO] nginx config:
worker_processes auto;
daemon off;
pid /tmp/nginx.pid;
```

```
error_log  /dev/stderr;
worker_rlimit_nofile 4096;
events {
  worker_connections 2048;
}
[2024-06-01:00:03:19:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:19:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:19:INFO] nginx config:
worker_processes auto;
daemon off;
pid /tmp/nginx.pid;
error_log  /dev/stderr;
worker_rlimit_nofile 4096;
events {
  worker_connections 2048;
}
http {
  include /etc/nginx/mime.types;
  default_type application/octet-stream;
  access_log /dev/stdout combined;
  upstream gunicorn {
    server unix:/tmp/gunicorn.sock;
  }
  server {
    listen 8080 deferred;
    client_max_body_size 0;
    keepalive_timeout 3;
    location ~ ^/(ping|invocations|execution-parameters) {
      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header Host $http_host;
      proxy_redirect off;
      proxy_read_timeout 60s;
      proxy_pass http://gunicorn;
    }
    location / {
      return 404 "{}";
    }
  }
}
[2024-06-01 00:03:19 +0000] [19] [INFO] Starting gunicorn 19.10.0
[2024-06-01 00:03:19 +0000] [19] [INFO] Listening at: unix:/tmp/gunicorn.sock (19)
[2024-06-01 00:03:19 +0000] [19] [INFO] Using worker: gevent
```

```
/miniconda3/lib/python3.8/os.py:1023: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, th
e default buffer size will be used
  return io.open(fd, *args, **kwargs)
[2024-06-01 00:03:19 +0000] [25] [INFO] Booting worker with pid: 25
[2024-06-01 00:03:19 +0000] [26] [INFO] Booting worker with pid: 26
[2024-06-01 00:03:19 +0000] [27] [INFO] Booting worker with pid: 27
[2024-06-01 00:03:20 +0000] [28] [INFO] Booting worker with pid: 28
http {
  include /etc/nginx/mime.types;
  default_type application/octet-stream;
  access_log /dev/stdout combined;
  upstream gunicorn {
    server unix:/tmp/gunicorn.sock;
  }
  server {
    listen 8080 deferred;
    client_max_body_size 0;
    keepalive_timeout 3;
    location ~ ^/(ping|invocations|execution-parameters) {
      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header Host $http_host;
      proxy_redirect off;
      proxy_read_timeout 60s;
      proxy_pass http://gunicorn;
    }
    location / {
      return 404 "{}";
    }
  }
}
[2024-06-01 00:03:19 +0000] [19] [INFO] Starting gunicorn 19.10.0
[2024-06-01 00:03:19 +0000] [19] [INFO] Listening at: unix:/tmp/gunicorn.sock (19)
[2024-06-01 00:03:19 +0000] [19] [INFO] Using worker: gevent
/miniconda3/lib/python3.8/os.py:1023: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, th
e default buffer size will be used
  return io.open(fd, *args, **kwargs)
[2024-06-01 00:03:19 +0000] [25] [INFO] Booting worker with pid: 25
[2024-06-01 00:03:19 +0000] [26] [INFO] Booting worker with pid: 26
[2024-06-01 00:03:19 +0000] [27] [INFO] Booting worker with pid: 27
[2024-06-01 00:03:20 +0000] [28] [INFO] Booting worker with pid: 28
[2024-06-01:00:03:21:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:21:INFO] Loading the model from /opt/ml/model/xgboost-model
```

```
[2024-06-01:00:03:21:INFO] Model objective : binary:logistic
[2024-06-01:00:03:21:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:21:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:03:21:INFO] Model objective : binary:logistic
[2024-06-01:00:03:22:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:22:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:03:22:INFO] Model objective : binary:logistic
[2024-06-01:00:03:22:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:22:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:03:22:INFO] Model objective : binary:logistic
[2024-06-01:00:03:21:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:21:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:03:21:INFO] Model objective : binary:logistic
[2024-06-01:00:03:21:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:21:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:03:21:INFO] Model objective : binary:logistic
[2024-06-01:00:03:22:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:22:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:03:22:INFO] Model objective : binary:logistic
[2024-06-01:00:03:22:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:22:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:03:22:INFO] Model objective : binary:logistic
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "GET /ping HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "GET /ping HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "GET /execution-parameters HTTP/1.1" 200 84 "-" "Go-http-client/1.1"
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:25:INFO] Determined delimiter of CSV input is ','
/miniconda3/lib/python3.8/site-packages/xgboost/core.py:122: UserWarning: ntree_limit is deprecated, use `iteration_r
ange` or model slicing instead.
  warnings.warn(
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "POST /invocations HTTP/1.1" 200 1397 "-" "Go-http-client/1.1"
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "GET /execution-parameters HTTP/1.1" 200 84 "-" "Go-http-client/1.1"
[2024-06-01:00:03:25:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:03:25:INFO] Determined delimiter of CSV input is ','
/miniconda3/lib/python3.8/site-packages/xgboost/core.py:122: UserWarning: ntree_limit is deprecated, use `iteration_r
ange` or model slicing instead.
  warnings.warn(
169.254.255.130 - - [01/Jun/2024:00:03:25 +0000] "POST /invocations HTTP/1.1" 200 1397 "-" "Go-http-client/1.1"
```

```
2024-06-01T00:03:25.680:[sagemaker logs]: MaxConcurrentTransforms=4, MaxPayloadInMB=6, BatchStrategy=MULTI_RECORD
CPU times: user 945 ms, sys: 46.6 ms, total: 991 ms
Wall time: 8min 5s
```

In [ ]:
```
# Grabbing output - display first 8 values
output_df = get_csv_output_from_s3(sm_transformer.output_path, batch_file_noID)
output_df.head(8)
```

Out[ ]:

	0
0	0.976714
1	0.707197
2	0.993038
3	0.060286
4	0.841149
5	0.968204
6	0.031473
7	0.009060

In [ ]:
```
# Join input with prediction results

# Content_type / accept and split_type / assemble_with are required to use IO joining feature
sm_transformer.assemble_with = "Line"
sm_transformer.accept = "text/csv"

# Start a transform job
input_location = "s3://{}/{}/batch/{}".format(
    bucket, prefix, batch_file
)

# Use input data with ID column cause InputFilter will filter it out
sm_transformer.transform(
    input_location,
    split_type="Line",
    content_type="text/csv",
    input_filter="$[1:]",
```

```
    join_source="Input",
)

sm_transformer.wait()
```

```
INFO:sagemaker:Creating transform job with name: sagemaker-xgboost-2024-06-01-00-04-18-898
```

```
.........................................[2024-06-01:00:11:37:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:37:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:37:INFO] nginx config:
worker_processes auto;
daemon off;
pid /tmp/nginx.pid;
error_log  /dev/stderr;
worker_rlimit_nofile 4096;
events {
  worker_connections 2048;
}
[2024-06-01:00:11:37:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:37:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:37:INFO] nginx config:
worker_processes auto;
daemon off;
pid /tmp/nginx.pid;
error_log  /dev/stderr;
worker_rlimit_nofile 4096;
events {
  worker_connections 2048;
}
http {
  include /etc/nginx/mime.types;
  default_type application/octet-stream;
  access_log /dev/stdout combined;
  upstream gunicorn {
    server unix:/tmp/gunicorn.sock;
  }
  server {
    listen 8080 deferred;
    client_max_body_size 0;
    keepalive_timeout 3;
    location ~ ^/(ping|invocations|execution-parameters) {
      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header Host $http_host;
      proxy_redirect off;
      proxy_read_timeout 60s;
      proxy_pass http://gunicorn;
    }
    location / {
      return 404 "{}";
```

```
    }
  }
}
[2024-06-01 00:11:37 +0000] [19] [INFO] Starting gunicorn 19.10.0
[2024-06-01 00:11:37 +0000] [19] [INFO] Listening at: unix:/tmp/gunicorn.sock (19)
[2024-06-01 00:11:37 +0000] [19] [INFO] Using worker: gevent
/miniconda3/lib/python3.8/os.py:1023: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, th
e default buffer size will be used
  return io.open(fd, *args, **kwargs)
[2024-06-01 00:11:37 +0000] [25] [INFO] Booting worker with pid: 25
[2024-06-01 00:11:37 +0000] [26] [INFO] Booting worker with pid: 26
[2024-06-01 00:11:37 +0000] [27] [INFO] Booting worker with pid: 27
[2024-06-01 00:11:37 +0000] [28] [INFO] Booting worker with pid: 28
http {
  include /etc/nginx/mime.types;
  default_type application/octet-stream;
  access_log /dev/stdout combined;
  upstream gunicorn {
    server unix:/tmp/gunicorn.sock;
  }
  server {
    listen 8080 deferred;
    client_max_body_size 0;
    keepalive_timeout 3;
    location ~ ^/(ping|invocations|execution-parameters) {
      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header Host $http_host;
      proxy_redirect off;
      proxy_read_timeout 60s;
      proxy_pass http://gunicorn;
    }
    location / {
      return 404 "{}";
    }
  }
}
[2024-06-01 00:11:37 +0000] [19] [INFO] Starting gunicorn 19.10.0
[2024-06-01 00:11:37 +0000] [19] [INFO] Listening at: unix:/tmp/gunicorn.sock (19)
[2024-06-01 00:11:37 +0000] [19] [INFO] Using worker: gevent
/miniconda3/lib/python3.8/os.py:1023: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, th
e default buffer size will be used
  return io.open(fd, *args, **kwargs)
```

```
[2024-06-01 00:11:37 +0000] [25] [INFO] Booting worker with pid: 25
[2024-06-01 00:11:37 +0000] [26] [INFO] Booting worker with pid: 26
[2024-06-01 00:11:37 +0000] [27] [INFO] Booting worker with pid: 27
[2024-06-01 00:11:37 +0000] [28] [INFO] Booting worker with pid: 28
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:42:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:11:42 +0000] "GET /ping HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
[2024-06-01:00:11:42:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:11:42 +0000] "GET /ping HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
169.254.255.130 - - [01/Jun/2024:00:11:42 +0000] "GET /execution-parameters HTTP/1.1" 200 84 "-" "Go-http-client/1.1"
169.254.255.130 - - [01/Jun/2024:00:11:42 +0000] "GET /execution-parameters HTTP/1.1" 200 84 "-" "Go-http-client/1.1"
[2024-06-01:00:11:43:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:43:INFO] Determined delimiter of CSV input is ','
/miniconda3/lib/python3.8/site-packages/xgboost/core.py:122: UserWarning: ntree_limit is deprecated, use `iteration_r
ange` or model slicing instead.
  warnings.warn(
169.254.255.130 - - [01/Jun/2024:00:11:43 +0000] "POST /invocations HTTP/1.1" 200 1397 "-" "Go-http-client/1.1"
[2024-06-01:00:11:43:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:43:INFO] Determined delimiter of CSV input is ','
```

```
/miniconda3/lib/python3.8/site-packages/xgboost/core.py:122: UserWarning: ntree_limit is deprecated, use `iteration_r
ange` or model slicing instead.
  warnings.warn(
169.254.255.130 - - [01/Jun/2024:00:11:43 +0000] "POST /invocations HTTP/1.1" 200 1397 "-" "Go-http-client/1.1"
2024-06-01T00:11:42.976:[sagemaker logs]: MaxConcurrentTransforms=4, MaxPayloadInMB=6, BatchStrategy=MULTI_RECORD

[2024-06-01:00:11:37:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:37:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:37:INFO] nginx config:
worker_processes auto;
daemon off;
pid /tmp/nginx.pid;
error_log  /dev/stderr;
worker_rlimit_nofile 4096;
events {
  worker_connections 2048;
}
[2024-06-01:00:11:37:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:37:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:37:INFO] nginx config:
worker_processes auto;
daemon off;
pid /tmp/nginx.pid;
error_log  /dev/stderr;
worker_rlimit_nofile 4096;
events {
  worker_connections 2048;
}
http {
  include /etc/nginx/mime.types;
  default_type application/octet-stream;
  access_log /dev/stdout combined;
  upstream gunicorn {
    server unix:/tmp/gunicorn.sock;
  }
  server {
    listen 8080 deferred;
    client_max_body_size 0;
    keepalive_timeout 3;
    location ~ ^/(ping|invocations|execution-parameters) {
      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header Host $http_host;
```

```
      proxy_redirect off;
      proxy_read_timeout 60s;
      proxy_pass http://gunicorn;
    }
    location / {
      return 404 "{}";
    }
  }
}
[2024-06-01 00:11:37 +0000] [19] [INFO] Starting gunicorn 19.10.0
[2024-06-01 00:11:37 +0000] [19] [INFO] Listening at: unix:/tmp/gunicorn.sock (19)
[2024-06-01 00:11:37 +0000] [19] [INFO] Using worker: gevent
/miniconda3/lib/python3.8/os.py:1023: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, th
e default buffer size will be used
  return io.open(fd, *args, **kwargs)
[2024-06-01 00:11:37 +0000] [25] [INFO] Booting worker with pid: 25
[2024-06-01 00:11:37 +0000] [26] [INFO] Booting worker with pid: 26
[2024-06-01 00:11:37 +0000] [27] [INFO] Booting worker with pid: 27
[2024-06-01 00:11:37 +0000] [28] [INFO] Booting worker with pid: 28
http {
  include /etc/nginx/mime.types;
  default_type application/octet-stream;
  access_log /dev/stdout combined;
  upstream gunicorn {
    server unix:/tmp/gunicorn.sock;
  }
  server {
    listen 8080 deferred;
    client_max_body_size 0;
    keepalive_timeout 3;
    location ~ ^/(ping|invocations|execution-parameters) {
      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header Host $http_host;
      proxy_redirect off;
      proxy_read_timeout 60s;
      proxy_pass http://gunicorn;
    }
    location / {
      return 404 "{}";
    }
  }
}
```

```
[2024-06-01 00:11:37 +0000] [19] [INFO] Starting gunicorn 19.10.0
[2024-06-01 00:11:37 +0000] [19] [INFO] Listening at: unix:/tmp/gunicorn.sock (19)
[2024-06-01 00:11:37 +0000] [19] [INFO] Using worker: gevent
/miniconda3/lib/python3.8/os.py:1023: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, th
e default buffer size will be used
  return io.open(fd, *args, **kwargs)
[2024-06-01 00:11:37 +0000] [25] [INFO] Booting worker with pid: 25
[2024-06-01 00:11:37 +0000] [26] [INFO] Booting worker with pid: 26
[2024-06-01 00:11:37 +0000] [27] [INFO] Booting worker with pid: 27
[2024-06-01 00:11:37 +0000] [28] [INFO] Booting worker with pid: 28
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:39:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:39:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:11:39:INFO] Model objective : binary:logistic
[2024-06-01:00:11:42:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:11:42 +0000] "GET /ping HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
[2024-06-01:00:11:42:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:11:42 +0000] "GET /ping HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
169.254.255.130 - - [01/Jun/2024:00:11:42 +0000] "GET /execution-parameters HTTP/1.1" 200 84 "-" "Go-http-client/1.1"
169.254.255.130 - - [01/Jun/2024:00:11:42 +0000] "GET /execution-parameters HTTP/1.1" 200 84 "-" "Go-http-client/1.1"
[2024-06-01:00:11:43:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:43:INFO] Determined delimiter of CSV input is ','
```

```
/miniconda3/lib/python3.8/site-packages/xgboost/core.py:122: UserWarning: ntree_limit is deprecated, use `iteration_r
ange` or model slicing instead.
  warnings.warn(
169.254.255.130 - - [01/Jun/2024:00:11:43 +0000] "POST /invocations HTTP/1.1" 200 1397 "-" "Go-http-client/1.1"
[2024-06-01:00:11:43:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:11:43:INFO] Determined delimiter of CSV input is ','
/miniconda3/lib/python3.8/site-packages/xgboost/core.py:122: UserWarning: ntree_limit is deprecated, use `iteration_r
ange` or model slicing instead.
  warnings.warn(
169.254.255.130 - - [01/Jun/2024:00:11:43 +0000] "POST /invocations HTTP/1.1" 200 1397 "-" "Go-http-client/1.1"
2024-06-01T00:11:42.976:[sagemaker logs]: MaxConcurrentTransforms=4, MaxPayloadInMB=6, BatchStrategy=MULTI_RECORD
```

In [ ]:
```
# Showcase outputs from the batch
output_df = get_csv_output_from_s3(sm_transformer.output_path, batch_file)
output_df.head(8)
```

Out[ ]:

	0	1	2	3	4	5	6	7	8	9	...	22	23	24	25	2
0	84458202	13.71	20.83	90.20	577.9	0.11890	0.16450	0.093660	0.059850	0.2196	...	28.14	110.60	897.0	0.1654	0.3682
1	845636	16.02	23.24	102.70	797.8	0.08206	0.06669	0.032990	0.033230	0.1528	...	33.88	123.80	1150.0	0.1181	0.1551
2	854002	19.27	26.47	127.90	1162.0	0.09401	0.17190	0.165700	0.075930	0.1853	...	30.90	161.40	1813.0	0.1509	0.6590
3	855167	13.44	21.58	86.18	563.0	0.08162	0.06031	0.031100	0.020310	0.1784	...	30.25	102.50	787.9	0.1094	0.2043
4	85638502	13.17	21.81	85.42	531.5	0.09714	0.10470	0.082590	0.052520	0.1746	...	29.89	105.50	740.7	0.1503	0.3904
5	857010	18.65	17.60	123.70	1076.0	0.10990	0.16860	0.197400	0.100900	0.1907	...	21.32	150.60	1567.0	0.1679	0.5090
6	857343	11.76	21.60	74.72	427.9	0.08637	0.04966	0.016570	0.011150	0.1495	...	25.72	82.98	516.5	0.1085	0.0861
7	857810	13.05	19.31	82.61	527.2	0.08060	0.03789	0.000692	0.004167	0.1819	...	22.25	90.24	624.1	0.1021	0.0619

8 rows × 32 columns

In [ ]:
```
# Update output filter to showcase only the ID and prediction results
sm_transformer.transform(
    input_location,
    split_type="Line",
    content_type="text/csv",
```

```
    input_filter="$[1:]",
    join_source="Input",
    output_filter="$[0,-1]",
)
sm_transformer.wait()
```

```
INFO:sagemaker:Creating transform job with name: sagemaker-xgboost-2024-06-01-00-12-23-876
```

```
.........................................
[2024-06-01:00:19:55:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:55:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:55:INFO] nginx config:
worker_processes auto;
daemon off;
pid /tmp/nginx.pid;
error_log  /dev/stderr;
worker_rlimit_nofile 4096;
events {
  worker_connections 2048;
}
http {
  include /etc/nginx/mime.types;
  default_type application/octet-stream;
  access_log /dev/stdout combined;
  upstream gunicorn {
    server unix:/tmp/gunicorn.sock;
  }
  server {
    listen 8080 deferred;
    client_max_body_size 0;
    keepalive_timeout 3;
    location ~ ^/(ping|invocations|execution-parameters) {
      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header Host $http_host;
      proxy_redirect off;
      proxy_read_timeout 60s;
      proxy_pass http://gunicorn;
    }
    location / {
      return 404 "{}";
    }
  }
}
[2024-06-01 00:19:55 +0000] [19] [INFO] Starting gunicorn 19.10.0
[2024-06-01 00:19:55 +0000] [19] [INFO] Listening at: unix:/tmp/gunicorn.sock (19)
[2024-06-01 00:19:55 +0000] [19] [INFO] Using worker: gevent
/miniconda3/lib/python3.8/os.py:1023: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, th
e default buffer size will be used
  return io.open(fd, *args, **kwargs)
[2024-06-01 00:19:55 +0000] [25] [INFO] Booting worker with pid: 25
```

```
[2024-06-01 00:19:55 +0000] [26] [INFO] Booting worker with pid: 26
[2024-06-01 00:19:55 +0000] [27] [INFO] Booting worker with pid: 27
[2024-06-01 00:19:55 +0000] [28] [INFO] Booting worker with pid: 28
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
[2024-06-01:00:20:01:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:20:01 +0000] "GET /ping HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
[2024-06-01:00:20:01:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:20:01 +0000] "GET /execution-parameters HTTP/1.1" 200 84 "-" "Go-http-client/1.1"
[2024-06-01:00:20:01:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:20:01:INFO] Determined delimiter of CSV input is ','
/miniconda3/lib/python3.8/site-packages/xgboost/core.py:122: UserWarning: ntree_limit is deprecated, use `iteration_r
ange` or model slicing instead.
  warnings.warn(
169.254.255.130 - - [01/Jun/2024:00:20:01 +0000] "POST /invocations HTTP/1.1" 200 1397 "-" "Go-http-client/1.1"
2024-06-01T00:20:01.533:[sagemaker logs]: MaxConcurrentTransforms=4, MaxPayloadInMB=6, BatchStrategy=MULTI_RECORD
[2024-06-01:00:19:55:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:55:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:55:INFO] nginx config:
worker_processes auto;
daemon off;
pid /tmp/nginx.pid;
error_log  /dev/stderr;
worker_rlimit_nofile 4096;
events {
  worker_connections 2048;
}
[2024-06-01:00:19:55:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:55:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:55:INFO] nginx config:
worker_processes auto;
daemon off;
```

```
pid /tmp/nginx.pid;
error_log  /dev/stderr;
worker_rlimit_nofile 4096;
events {
  worker_connections 2048;
}
http {
  include /etc/nginx/mime.types;
  default_type application/octet-stream;
  access_log /dev/stdout combined;
  upstream gunicorn {
    server unix:/tmp/gunicorn.sock;
  }
  server {
    listen 8080 deferred;
    client_max_body_size 0;
    keepalive_timeout 3;
    location ~ ^/(ping|invocations|execution-parameters) {
      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header Host $http_host;
      proxy_redirect off;
      proxy_read_timeout 60s;
      proxy_pass http://gunicorn;
    }
    location / {
      return 404 "{}";
    }
  }
}
[2024-06-01 00:19:55 +0000] [19] [INFO] Starting gunicorn 19.10.0
[2024-06-01 00:19:55 +0000] [19] [INFO] Listening at: unix:/tmp/gunicorn.sock (19)
[2024-06-01 00:19:55 +0000] [19] [INFO] Using worker: gevent
/miniconda3/lib/python3.8/os.py:1023: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, th
e default buffer size will be used
  return io.open(fd, *args, **kwargs)
[2024-06-01 00:19:55 +0000] [25] [INFO] Booting worker with pid: 25
[2024-06-01 00:19:55 +0000] [26] [INFO] Booting worker with pid: 26
[2024-06-01 00:19:55 +0000] [27] [INFO] Booting worker with pid: 27
[2024-06-01 00:19:55 +0000] [28] [INFO] Booting worker with pid: 28
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
```

```
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
http {
  include /etc/nginx/mime.types;
  default_type application/octet-stream;
  access_log /dev/stdout combined;
  upstream gunicorn {
    server unix:/tmp/gunicorn.sock;
  }
  server {
    listen 8080 deferred;
    client_max_body_size 0;
    keepalive_timeout 3;
    location ~ ^/(ping|invocations|execution-parameters) {
      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header Host $http_host;
      proxy_redirect off;
      proxy_read_timeout 60s;
      proxy_pass http://gunicorn;
    }
    location / {
      return 404 "{}";
    }
  }
}
[2024-06-01 00:19:55 +0000] [19] [INFO] Starting gunicorn 19.10.0
[2024-06-01 00:19:55 +0000] [19] [INFO] Listening at: unix:/tmp/gunicorn.sock (19)
[2024-06-01 00:19:55 +0000] [19] [INFO] Using worker: gevent
/miniconda3/lib/python3.8/os.py:1023: RuntimeWarning: line buffering (buffering=1) isn't supported in binary mode, th
e default buffer size will be used
  return io.open(fd, *args, **kwargs)
[2024-06-01 00:19:55 +0000] [25] [INFO] Booting worker with pid: 25
[2024-06-01 00:19:55 +0000] [26] [INFO] Booting worker with pid: 26
[2024-06-01 00:19:55 +0000] [27] [INFO] Booting worker with pid: 27
[2024-06-01 00:19:55 +0000] [28] [INFO] Booting worker with pid: 28
```

```
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
[2024-06-01:00:19:57:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:19:57:INFO] Loading the model from /opt/ml/model/xgboost-model
[2024-06-01:00:19:57:INFO] Model objective : binary:logistic
[2024-06-01:00:20:01:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:20:01 +0000] "GET /ping HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
[2024-06-01:00:20:01:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:20:01:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:20:01 +0000] "GET /ping HTTP/1.1" 200 0 "-" "Go-http-client/1.1"
[2024-06-01:00:20:01:INFO] No GPUs detected (normal if no gpus installed)
169.254.255.130 - - [01/Jun/2024:00:20:01 +0000] "GET /execution-parameters HTTP/1.1" 200 84 "-" "Go-http-client/1.1"
[2024-06-01:00:20:01:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:20:01:INFO] Determined delimiter of CSV input is ','
/miniconda3/lib/python3.8/site-packages/xgboost/core.py:122: UserWarning: ntree_limit is deprecated, use `iteration_r
ange` or model slicing instead.
  warnings.warn(
169.254.255.130 - - [01/Jun/2024:00:20:01 +0000] "POST /invocations HTTP/1.1" 200 1397 "-" "Go-http-client/1.1"
169.254.255.130 - - [01/Jun/2024:00:20:01 +0000] "GET /execution-parameters HTTP/1.1" 200 84 "-" "Go-http-client/1.1"
[2024-06-01:00:20:01:INFO] No GPUs detected (normal if no gpus installed)
[2024-06-01:00:20:01:INFO] Determined delimiter of CSV input is ','
/miniconda3/lib/python3.8/site-packages/xgboost/core.py:122: UserWarning: ntree_limit is deprecated, use `iteration_r
ange` or model slicing instead.
  warnings.warn(
169.254.255.130 - - [01/Jun/2024:00:20:01 +0000] "POST /invocations HTTP/1.1" 200 1397 "-" "Go-http-client/1.1"
2024-06-01T00:20:01.533:[sagemaker logs]: MaxConcurrentTransforms=4, MaxPayloadInMB=6, BatchStrategy=MULTI_RECORD
```

In [ ]:

```python
# Inspect new ouptut for the transformer
output_df = get_csv_output_from_s3(sm_transformer.output_path, batch_file)
output_df.head(8)
```

Out[ ]:

	0	1
0	84458202	0.976714
1	845636	0.707197
2	854002	0.993038
3	855167	0.060286
4	85638502	0.841149
5	857010	0.968204
6	857343	0.031473
7	857810	0.009060

Upload the Sagemaker Model created during our training job to the Sagemaker Model Registry

In [ ]:
```
sagemaker = boto3.client("sagemaker")

model_name = job_name
print(model_name)


info = sagemaker.describe_training_job(TrainingJobName=model_name)
model_data = info["ModelArtifacts"]["S3ModelArtifacts"]

primary_container = {"Image": image, "ModelDataUrl": model_data}

# Save our model to the Sagemaker Model Registry
create_model_response = sagemaker.create_model(
    ModelName=model_name, ExecutionRoleArn=role, PrimaryContainer=primary_container
)

print(create_model_response["ModelArn"])
```

```
xgb-2024-05-31-23-53-31
arn:aws:sagemaker:us-east-1:004608622582:model/xgb-2024-05-31-23-53-31
```

Create Endpont Configuration

In [ ]:
```
# Create Endpoint Configuration


# Create an endpoint config name. Here we create one based on the date
# so it we can search endpoints based on creation time.
endpoint_config_name = 'lab4-1-endpoint-config' + strftime("%Y-%m-%d-%H-%M-%S", gmtime())

instance_type = 'ml.m5.xlarge'

endpoint_config_response = sagemaker.create_endpoint_config(
    EndpointConfigName=endpoint_config_name, # You will specify this name in a CreateEndpoint request.
    # List of ProductionVariant objects, one for each model that you want to host at this endpoint.
    ProductionVariants=[
        {
            "VariantName": "variant1", # The name of the production variant.
            "ModelName": model_name,
            "InstanceType": instance_type, # Specify the compute instance type.
            "InitialInstanceCount": 1 # Number of instances to launch initially.
        }
    ]
)

print(f"Created EndpointConfig: {endpoint_config_response['EndpointConfigArn']}")
```

```
Created EndpointConfig: arn:aws:sagemaker:us-east-1:004608622582:endpoint-config/lab4-1-endpoint-config2024-06-01-00-20-29
```

In [ ]:
```
# Deploy our model to real-time endpoint

endpoint_name = 'lab4-1-endpoint' + strftime("%Y-%m-%d-%H-%M-%S", gmtime())


create_endpoint_response = sagemaker.create_endpoint(
                                            EndpointName=endpoint_name,
                                            EndpointConfigName=endpoint_config_name)
```

In [ ]:
```
# Wait for endpoint to spin up
```

```
sagemaker.describe_endpoint(EndpointName=endpoint_name)

while True:
    print("Getting Job Status")
    res = sagemaker.describe_endpoint(EndpointName=endpoint_name)
    state = res["EndpointStatus"]

    if state == "InService":
        print("Endpoint in Service")
        break
    elif state == "Creating":
        print("Endpoint still creating...")
        sleep(60)
    else:
        print("Endpoint Creation Error - Check Sagemaker Console")
        break
```

```
Getting Job Status
Endpoint still creating...
Getting Job Status
Endpoint still creating...
Getting Job Status
Endpoint still creating...
Getting Job Status
Endpoint still creating...
Getting Job Status
Endpoint in Service
```

In [ ]:
```
# Invoke Endpoint

sagemaker_runtime = boto3.client("sagemaker-runtime", region_name=region)

response = sagemaker_runtime.invoke_endpoint(
                            EndpointName=endpoint_name,
                            ContentType='text/csv',
                            Body=data_batch_noID.to_csv(header=None, index=False).strip('\n').split('\n')[0]
                            )
print(response['Body'].read().decode('utf-8'))
```

```
0.9767142534255981
```

In [ ]:
```
# Checkout out the response
response
```

Out[ ]:
```
{'ResponseMetadata': {'RequestId': '73129524-76b5-43d7-b2f1-757d074f3bd7',
  'HTTPStatusCode': 200,
  'HTTPHeaders': {'x-amzn-requestid': '73129524-76b5-43d7-b2f1-757d074f3bd7',
   'x-amzn-invoked-production-variant': 'variant1',
   'date': 'Sat, 01 Jun 2024 00:24:30 GMT',
   'content-type': 'text/csv; charset=utf-8',
   'content-length': '19',
   'connection': 'keep-alive'},
  'RetryAttempts': 0},
 'ContentType': 'text/csv; charset=utf-8',
 'InvokedProductionVariant': 'variant1',
 'Body': <botocore.response.StreamingBody at 0x7f371978d6f0>}
```

In [ ]:
```
# Delete Endpoint
#sagemaker.delete_endpoint(EndpointName=endpoint_name)
```

Part 1

Setup Group Model

In [ ]:
```
# Notes
# region = region
# role = role

# Create sm client
sm_client = boto3.client('sagemaker', region_name=region)
```

In [ ]:
```
# Create Model Group
model_package_group_name = "breast-cancer-group-name-detector-" + str(round(time.time()))
model_package_group_input_dict = {
 "ModelPackageGroupName" : model_package_group_name,
 "ModelPackageGroupDescription" : "Sample model package group"
}

create_model_package_group_response = sm_client.create_model_package_group(**model_package_group_input_dict)
print('ModelPackageGroup Arn : {}'.format(create_model_package_group_response['ModelPackageGroupArn']))
```

```
ModelPackageGroup Arn : arn:aws:sagemaker:us-east-1:004608622582:model-package-group/breast-cancer-group-name-detecto
r-1717207799
```

In [ ]:
```
# Running the describe_model_package_group
response = sm_client.describe_model_package_group(
    ModelPackageGroupName=model_package_group_name
)

print(response)
```

```
{'ModelPackageGroupName': 'breast-cancer-group-name-detector-1717207799', 'ModelPackageGroupArn': 'arn:aws:sagemaker:
us-east-1:004608622582:model-package-group/breast-cancer-group-name-detector-1717207799', 'ModelPackageGroupDescripti
on': 'Sample model package group', 'CreationTime': datetime.datetime(2024, 6, 1, 2, 9, 59, 60000, tzinfo=tzlocal()),
'CreatedBy': {'UserProfileArn': 'arn:aws:sagemaker:us-east-1:004608622582:user-profile/d-ot3x26nvt9y2/pthai', 'UserPr
ofileName': 'pthai', 'DomainId': 'd-ot3x26nvt9y2', 'IamIdentity': {'Arn': 'arn:aws:sts::004608622582:assumed-role/Lab
Role/SageMaker', 'PrincipalId': 'AROAQCEVR773FGX7Y4SZW:SageMaker'}}, 'ModelPackageGroupStatus': 'Completed', 'Respons
eMetadata': {'RequestId': '67055d90-500b-4482-ae4d-5aa088fdeeb4', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requ
estid': '67055d90-500b-4482-ae4d-5aa088fdeeb4', 'content-type': 'application/x-amz-json-1.1', 'content-length': '62
3', 'date': 'Sat, 01 Jun 2024 02:10:05 GMT'}, 'RetryAttempts': 0}}
```

In [ ]:
```
sm_client.list_model_packages(ModelPackageGroupName="breast-cancer-group-name-detector-1717207799")
```

Out[ ]:
```
{'ModelPackageSummaryList': [],
 'ResponseMetadata': {'RequestId': 'dbff337a-8236-4d0a-8171-14af14e9f3bb',
  'HTTPStatusCode': 200,
  'HTTPHeaders': {'x-amzn-requestid': 'dbff337a-8236-4d0a-8171-14af14e9f3bb',
   'content-type': 'application/x-amz-json-1.1',
   'content-length': '30',
   'date': 'Sat, 01 Jun 2024 02:15:24 GMT'},
  'RetryAttempts': 0}}
```

Part 2

Set Up Model Package

In [ ]:
```
# Specify model source
model_url = "s3://sagemaker-us-east-1-004608622582/DEMO-breast-cancer-prediction-xgboost-highlevel/output/xgb-2024-05
image_uri = "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.7-1"

modelpackage_inference_specification =  {
    "InferenceSpecification": {
```

```
        "Containers": [
            {
                "Image": image_uri,
                "ModelDataUrl": model_url
            }
        ],
        "SupportedTransformInstanceTypes": [
            'ml.m4.xlarge'
        ],
        "SupportedRealtimeInferenceInstanceTypes": [
            'ml.m4.xlarge'
        ],
        "SupportedContentTypes": ["text/csv"],
        "SupportedResponseMIMETypes": ["text/csv"],
    }
}
modelpackage_inference_specification
```

Out[ ]:
```
{'InferenceSpecification': {'Containers': [{'Image': '683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.7-1',
    'ModelDataUrl': 's3://sagemaker-us-east-1-004608622582/DEMO-breast-cancer-prediction-xgboost-highlevel/output/xgb-2024-05-31-19-31-38/xgb-2024-05-31-19-31-38/output/model.tar.gz'}],
  'SupportedTransformInstanceTypes': ['ml.m4.xlarge'],
  'SupportedRealtimeInferenceInstanceTypes': ['ml.m4.xlarge'],
  'SupportedContentTypes': ['text/csv'],
  'SupportedResponseMIMETypes': ['text/csv']}}
```

In [ ]:
```
# Create model package input dictionary
model_package_name = "breast-cancer-model-name-detector-" + str(round(time.time()))
create_model_package_input_dict = {
    "ModelPackageGroupName" : model_package_group_name,
    "ModelPackageDescription" : "Model to detect breast cancer",
    "ModelApprovalStatus" : "PendingManualApproval"
}
create_model_package_input_dict.update(modelpackage_inference_specification)
```

In [ ]:
```
create_model_package_input_dict
```

Out[ ]:
```
{'ModelPackageGroupName': 'breast-cancer-group-name-detector-1717207799',
 'ModelPackageDescription': 'Model to detect breast cancer',
 'ModelApprovalStatus': 'PendingManualApproval',
 'InferenceSpecification': {'Containers': [{'Image': '683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.7-1',
    'ModelDataUrl': 's3://sagemaker-us-east-1-004608622582/DEMO-breast-cancer-prediction-xgboost-highlevel/output/xgb-2024-05-31-19-31-38/xgb-2024-05-31-19-31-38/output/model.tar.gz'}],
  'SupportedTransformInstanceTypes': ['ml.m4.xlarge'],
  'SupportedRealtimeInferenceInstanceTypes': ['ml.m4.xlarge'],
  'SupportedContentTypes': ['text/csv'],
  'SupportedResponseMIMETypes': ['text/csv']}}
```

In [ ]:
```
# Create model package
create_model_package_response = sm_client.create_model_package(**create_model_package_input_dict)
model_package_arn = create_model_package_response["ModelPackageArn"]
print('ModelPackage Version ARN : {}'.format(model_package_arn))
```

```
ModelPackage Version ARN : arn:aws:sagemaker:us-east-1:004608622582:model-package/breast-cancer-group-name-detector-1717207799/1
```

In [ ]:
```
# Running the describe_model_package_group
response = sm_client.describe_model_package(
    ModelPackageName="arn:aws:sagemaker:us-east-1:004608622582:model-package/breast-cancer-group-name-detector-171720
)

print(response)
```

{'ModelPackageGroupName': 'breast-cancer-group-name-detector-1717207799', 'ModelPackageVersion': 1, 'ModelPackageAr
n': 'arn:aws:sagemaker:us-east-1:004608622582:model-package/breast-cancer-group-name-detector-1717207799/1', 'ModelPa
ckageDescription': 'Model to detect breast cancer', 'CreationTime': datetime.datetime(2024, 6, 1, 2, 20, 40, 742000,
tzinfo=tzlocal()), 'InferenceSpecification': {'Containers': [{'Image': '683313688378.dkr.ecr.us-east-1.amazonaws.com/
sagemaker-xgboost:1.7-1', 'ImageDigest': 'sha256:cf81520a3b695293022793e292cf8bc3732b79231a6ebe1fb308086f6163a875',
'ModelDataUrl': 's3://sagemaker-us-east-1-004608622582/DEMO-breast-cancer-prediction-xgboost-highlevel/output/xgb-202
4-05-31-19-31-38/xgb-2024-05-31-19-31-38/output/model.tar.gz'}], 'SupportedTransformInstanceTypes': ['ml.m4.xlarge'],
'SupportedRealtimeInferenceInstanceTypes': ['ml.m4.xlarge'], 'SupportedContentTypes': ['text/csv'], 'SupportedRespons
eMIMETypes': ['text/csv']}, 'ModelPackageStatus': 'Completed', 'ModelPackageStatusDetails': {'ValidationStatuses':
[], 'ImageScanStatuses': []}, 'CertifyForMarketplace': False, 'ModelApprovalStatus': 'PendingManualApproval', 'Create
dBy': {'UserProfileArn': 'arn:aws:sagemaker:us-east-1:004608622582:user-profile/d-ot3x26nvt9y2/pthai', 'UserProfileNa
me': 'pthai', 'DomainId': 'd-ot3x26nvt9y2', 'IamIdentity': {'Arn': 'arn:aws:sts::004608622582:assumed-role/LabRole/Sa
geMaker', 'PrincipalId': 'AROAQCEVR773FGX7Y4SZW:SageMaker'}}, 'ResponseMetadata': {'RequestId': 'e38bdd72-ceb6-4345-a
2e6-9644df4f5c7a', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': 'e38bdd72-ceb6-4345-a2e6-9644df4f5c7a',
'content-type': 'application/x-amz-json-1.1', 'content-length': '1363', 'date': 'Sat, 01 Jun 2024 02:21:59 GMT'}, 'Re
tryAttempts': 0}}

Part 3

Creating the Model Card

In [ ]:
```
# Create the model card based on our ARN
mp_details = ModelPackage.from_model_package_arn(
    model_package_arn="arn:aws:sagemaker:us-east-1:004608622582:model-package/breast-cancer-group-name-detector-17172
    sagemaker_session=sess,
)
```

In [ ]:
```
print('model pacakge', mp_details.__dict__)
```

model pacakge {'model_package_arn': 'arn:aws:sagemaker:us-east-1:004608622582:model-package/breast-cancer-group-name-
detector-1717207799/1', 'model_package_description': 'Model to detect breast cancer', '_model_package_status': <Model
PackageStatusEnum.COMPLETED: 'Completed'>, '_model_approval_status': <ModelApprovalStatusEnum.PENDING_MANUAL_APPROVA
L: 'PendingManualApproval'>, 'approval_description': None, 'model_package_group_name': 'breast-cancer-group-name-dete
ctor-1717207799', 'model_package_name': None, 'model_package_version': 1, 'domain': None, 'task': None, '_created_b
y': <sagemaker.model_card.model_card.ModelPackageCreator object at 0x7f3719a2b790>, '_source_algorithms': [], '_infer
ence_specification': <sagemaker.model_card.model_card.InferenceSpecification object at 0x7f3718811900>, '_model_metri
cs': None}

In [ ]:
```
# Create the model card
model_card_name = model_package_group_name
my_card = ModelCard(
```

```
    name=model_card_name,
    sagemaker_session = sess,
    model_package_details = mp_details
)
```

```
INFO:sagemaker.model_card.model_card:Evaluation details auto-discovery was unsuccessful. ModelMetrics was not found i
n the given model package. Please create one from scratch with EvaluationJob.
```

In [ ]:
```
# Other Option
```

In [ ]:
```
model_name="xgb-2024-05-31-23-53-31"
```

In [ ]:
```
# Create the model
model_overview = ModelOverview.from_model_name(
    model_name=model_name,
    sagemaker_session=sess,
    model_description="Breast Cancer Identification",
    problem_type="Classification",
    algorithm_type="CNNs",
    model_creator="Assignment4",
    model_owner="Assignment4",
)
print(f"Model id: {model_overview.model_id}")
print(f"Model training images: {model_overview.inference_environment.container_image}")
print(f"Model: {model_overview.model_artifact}")
```

```
Model id: arn:aws:sagemaker:us-east-1:004608622582:model/xgb-2024-05-31-23-53-31
Model training images: ['683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.7-1']
Model: ['s3://sagemaker-us-east-1-004608622582/DEMO-breast-cancer-prediction-xgboost-highlevel/output/xgb-2024-05-31-
23-53-31/xgb-2024-05-31-23-53-31/output/model.tar.gz']
```

In [ ]:
```
# Auto collect training details
objective_function = ObjectiveFunction(
    function=Function(
        function=ObjectiveFunctionEnum.MINIMIZE,
        facet=FacetEnum.LOSS,
    ),
    notes="This is an example objective function.",
)
training_details = TrainingDetails.from_model_overview(
    model_overview=model_overview,
    sagemaker_session=sess,
```

```
    objective_function=objective_function,
    training_observations="Add model training observations here.",
)
print(f"Training job id: {training_details.training_job_details.training_arn}")
print(
    f"Training image: {training_details.training_job_details.training_environment.container_image}"
)
print("Training Metrics: ")
print(
    [
        {"name": i.name, "value": i.value}
        for i in training_details.training_job_details.training_metrics
    ]
)
```

```
Training job id: arn:aws:sagemaker:us-east-1:004608622582:training-job/xgb-2024-05-31-23-53-31
Training image: ['683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.7-1']
Training Metrics:
[{'name': 'validation:logloss', 'value': 0.09588000178337097}, {'name': 'train:logloss', 'value': 0.0790000036358833
3}]
```

In [ ]:
```
# Collect Evaluation Data
manual_metric_group = MetricGroup(
    name="binary classification metrics",
    metric_data=[Metric(name="accuracy", type=MetricTypeEnum.NUMBER, value=0.5)],
)
example_evaluation_job = EvaluationJob(
    name="Example evaluation job",
    evaluation_observation="Evaluation observations.",
    datasets=["s3://path/to/evaluation/data"],
    metric_groups=[manual_metric_group],
)
evaluation_details = [example_evaluation_job]
```

In [ ]:
```
# More Details
intended_uses = IntendedUses(
    purpose_of_model="Predict Breaset Cancer",
    intended_uses="Not used except this test.",
    factors_affecting_model_efficiency="No.",
    risk_rating=RiskRatingEnum.LOW,
    explanations_for_risk_rating="Just an example.",
)
```

```
business_details = BusinessDetails(
    business_problem="Fighting Cancer",
    business_stakeholders="Patients, Doctors, Hosptital, Family, Friends",
    line_of_business="Health Care",
)
additional_information = AdditionalInformation(
    ethical_considerations="TBD",
    caveats_and_recommendations="Needs some EDA",
    custom_details={"custom details1": "details value"},
)
```

In [ ]:
```
# Model Card Parameters
model_card_name = "breast-cancer-model-card"
my_card = ModelCard(
    name=model_card_name,
    status=ModelCardStatusEnum.DRAFT,
    model_overview=model_overview,
    training_details=training_details,
    intended_uses=intended_uses,
    business_details=business_details,
    evaluation_details=evaluation_details,
    additional_information=additional_information,
    sagemaker_session=sess,
)
```

In [ ]:
```
# Create Model Card
my_card.create()
print(f"Model card {my_card.name} is successfully created with id {my_card.arn}")
```

```
INFO:sagemaker.model_card.model_card:Creating model card with name: breast-cancer-model-card
Model card breast-cancer-model-card is successfully created with id arn:aws:sagemaker:us-east-1:004608622582:model-ca
rd/breast-cancer-model-card
```

In [ ]:
```
# Descrive Model Card
response = sm_client.describe_model_card(
    ModelCardName=model_card_name
)
print(response)
```

{'ModelCardArn': 'arn:aws:sagemaker:us-east-1:004608622582:model-card/breast-cancer-model-card', 'ModelCardName': 'br
east-cancer-model-card', 'ModelCardVersion': 1, 'Content': '{"model_overview": {"model_id": "arn:aws:sagemaker:us-eas
t-1:004608622582:model/xgb-2024-05-31-23-53-31", "model_name": "xgb-2024-05-31-23-53-31", "model_description": "Breas
t Cancer Identification", "problem_type": "Classification", "algorithm_type": "CNNs", "model_creator": "Assignment4",
"model_owner": "Assignment4", "model_artifact": ["s3://sagemaker-us-east-1-004608622582/DEMO-breast-cancer-prediction
-xgboost-highlevel/output/xgb-2024-05-31-23-53-31/xgb-2024-05-31-23-53-31/output/model.tar.gz"], "inference_environme
nt": {"container_image": ["683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.7-1"]}}, "intended_uses":
{"purpose_of_model": "Predict Breaset Cancer", "intended_uses": "Not used except this test.", "factors_affecting_mode
l_efficiency": "No.", "risk_rating": "Low", "explanations_for_risk_rating": "Just an example."}, "business_details":
{"business_problem": "Fighting Cancer", "business_stakeholders": "Patients, Doctors, Hosptital, Family, Friends", "li
ne_of_business": "Health Care"}, "training_details": {"objective_function": {"function": {"function": "Minimize", "fa
cet": "Loss"}, "notes": "This is an example objective function."}, "training_observations": "Add model training obser
vations here.", "training_job_details": {"training_arn": "arn:aws:sagemaker:us-east-1:004608622582:training-job/xgb-2
024-05-31-23-53-31", "training_datasets": [], "training_environment": {"container_image": ["683313688378.dkr.ecr.us-e
ast-1.amazonaws.com/sagemaker-xgboost:1.7-1"]}, "training_metrics": [{"name": "validation:logloss", "value": 0.095880
00178337097}, {"name": "train:logloss", "value": 0.07900000363588333}], "user_provided_training_metrics": [], "hyper_
parameters": [{"name": "eta", "value": "0.2"}, {"name": "gamma", "value": "4"}, {"name": "max_depth", "value": "5"},
{"name": "min_child_weight", "value": "6"}, {"name": "num_round", "value": "100"}, {"name": "objective", "value": "bi
nary:logistic"}, {"name": "subsample", "value": "0.8"}, {"name": "verbosity", "value": "0"}], "user_provided_hyper_pa
rameters": []}}, "evaluation_details": [{"name": "Example evaluation job", "evaluation_observation": "Evaluation obse
rvations.", "datasets": ["s3://path/to/evaluation/data"], "metric_groups": [{"name": "binary classification metrics",
"metric_data": [{"name": "accuracy", "type": "number", "value": 0.5}]}]}], "additional_information": {"ethical_consid
erations": "TBD", "caveats_and_recommendations": "Needs some EDA", "custom_details": {"custom details1": "details val
ue"}}}', 'ModelCardStatus': 'Draft', 'CreationTime': datetime.datetime(2024, 6, 1, 2, 55, 20, 895000, tzinfo=tzlocal
()), 'CreatedBy': {'UserProfileArn': 'arn:aws:sagemaker:us-east-1:004608622582:user-profile/d-ot3x26nvt9y2/pthai', 'U
serProfileName': 'pthai', 'DomainId': 'd-ot3x26nvt9y2'}, 'LastModifiedTime': datetime.datetime(2024, 6, 1, 2, 55, 20,
895000, tzinfo=tzlocal()), 'LastModifiedBy': {'UserProfileArn': 'arn:aws:sagemaker:us-east-1:004608622582:user-profil
e/d-ot3x26nvt9y2/pthai', 'UserProfileName': 'pthai', 'DomainId': 'd-ot3x26nvt9y2'}, 'ResponseMetadata': {'RequestId':
'42f6963b-40de-4dc9-bf23-3fdada8a4ea0', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': '42f6963b-40de-4dc
9-bf23-3fdada8a4ea0', 'content-type': 'application/x-amz-json-1.1', 'content-length': '3362', 'date': 'Sat, 01 Jun 20
24 02:57:04 GMT'}, 'RetryAttempts': 0}}

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:
```
a
```

In [ ]:
```
%%html

<p><b>Shutting down your kernel for this notebook to release resources.</b></p>
<button class="sm-command-button" data-commandlinker-command="kernelmenu:shutdown" style="display:none;">Shutdown Ker

<script>
try {
    els = document.getElementsByClassName("sm-command-button");
    els[0].click();
}
catch(err) {
    // NoOp
}
</script>
```

Shutting down your kernel for this notebook to release resources.