Create S3 bucket

```
import boto3
s3 = boto3.resource('s3')
```

Boto3 is a AWS SDK

```
region = 'eu-north-1'

# Create an S3 client with the specified region
s3 = boto3.client('s3', region_name=region)

bucket_name = 'yahoofinancestockprice-1'
try:
    # Create the S3 bucket with the specified region
    s3.create_bucket(Bucket=bucket_name,
CreateBucketConfiguration={'LocationConstraint': region})
    print('S3 bucket has been created')
except Exception as e:
    print('S3 error', e)


S3 bucket has been created
```

Create train and validate CSV

```
!pip install yfinance

Collecting yfinance
  Downloading yfinance-0.2.33-py2.py3-none-any.whl.metadata (11 kB)
Requirement already satisfied: pandas>=1.3.0 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from yfinance) (2.1.1)
Requirement already satisfied: numpy>=1.16.5 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from yfinance) (1.22.4)
Requirement already satisfied: requests>=2.31 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from yfinance) (2.31.0)
Collecting multitasking>=0.0.7 (from yfinance)
  Downloading multitasking-0.0.11-py3-none-any.whl (8.5 kB)
Collecting lxml>=4.9.1 (from yfinance)
  Downloading lxml-5.0.0-cp310-cp310-
manylinux_2_17_x86_64.manylinux2014_x86_64.manylinux_2_24_x86_64.whl.m
etadata (6.7 kB)
Collecting appdirs>=1.4.4 (from yfinance)
  Downloading appdirs-1.4.4-py2.py3-none-any.whl (9.6 kB)
Requirement already satisfied: pytz>=2022.5 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
```

```
(from yfinance) (2023.3.post1)
Collecting frozendict>=2.3.4 (from yfinance)
  Downloading frozendict-2.4.0-cp310-cp310-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (23 kB)
Collecting peewee>=3.16.2 (from yfinance)
  Downloading peewee-3.17.0.tar.gz (2.9 MB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.9/2.9 MB 80.8 MB/s eta
0:00:00:00:01
ents to build wheel ... etadata (pyproject.toml) ... ent already
satisfied: beautifulsoup4>=4.11.1 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from yfinance) (4.12.2)
Collecting html5lib>=1.1 (from yfinance)
  Downloading html5lib-1.1-py2.py3-none-any.whl (112 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 112.2/112.2 kB 1.4 MB/s eta
0:00:00a 0:00:01
ent already satisfied: soupsieve>1.2 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from beautifulsoup4>=4.11.1->yfinance) (2.5)
Requirement already satisfied: six>=1.9 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from html5lib>=1.1->yfinance) (1.16.0)
Requirement already satisfied: webencodings in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from html5lib>=1.1->yfinance) (0.5.1)
Requirement already satisfied: python-dateutil>=2.8.2 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from
pandas>=1.3.0->yfinance) (2.8.2)
Requirement already satisfied: tzdata>=2022.1 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from pandas>=1.3.0->yfinance) (2023.3)
Requirement already satisfied: charset-normalizer<4,>=2 in /home/ec2-
user/anaconda3/envs/python3/lib/python3.10/site-packages (from
requests>=2.31->yfinance) (3.3.1)
Requirement already satisfied: idna<4,>=2.5 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from requests>=2.31->yfinance) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from requests>=2.31->yfinance) (1.26.18)
Requirement already satisfied: certifi>=2017.4.17 in
/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages
(from requests>=2.31->yfinance) (2023.7.22)
Downloading yfinance-0.2.33-py2.py3-none-any.whl (69 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 69.0/69.0 kB 10.2 MB/s eta
0:00:00
anylinux_2_17_x86_64.manylinux2014_x86_64.whl (116 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 116.7/116.7 kB 15.0 MB/s eta
0:00:00
```

```
l-5.0.0-cp310-cp310-
manylinux_2_17_x86_64.manylinux2014_x86_64.manylinux_2_24_x86_64.whl
(7.1 MB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 7.1/7.1 MB 93.0 MB/s eta
0:00:00:00:0100:01
l) ... e=peewee-3.17.0-cp310-cp310-linux_x86_64.whl size=275946
sha256=cce7b0ea0b468fa697a3c976af561b4e6df77079d8094065d5bceaa4b5267fc
9
  Stored in directory:
/home/ec2-user/.cache/pip/wheels/c7/70/ad/212867e96e7004265a69c4aa5dcf
f00a95f547a67ba26e7e76
Successfully built peewee
Installing collected packages: peewee, multitasking, appdirs, lxml,
html5lib, frozendict, yfinance
Successfully installed appdirs-1.4.4 frozendict-2.4.0 html5lib-1.1
lxml-5.0.0 multitasking-0.0.11 peewee-3.17.0 yfinance-0.2.33
```

```python
import pandas as pd
from datetime import datetime
import yfinance as yf

#Initialize the start and end date
start_date = datetime(2019,1,1)
end_date = datetime(2024,1,1)

#get the data
df_data = yf.download('IIF', start = start_date, end= end_date)

df_data.reset_index(inplace=True)

df_data
```

```
[*********************100%%**********************]  1 of 1 completed
```

```
           Date       Open       High        Low      Close  Adj Close
Volume
0    2019-01-02  19.910000  19.969999  19.760000  19.910000  14.000065
59800
1    2019-01-03  19.780001  19.780001  19.520000  19.600000  13.782083
40800
2    2019-01-04  19.830000  20.580000  19.809999  20.500000  14.414933
19700
3    2019-01-07  20.190001  20.190001  19.959999  20.150000  14.168824
97500
4    2019-01-08  19.990000  20.320000  19.990000  20.309999  14.281331
25800
...         ...        ...        ...        ...        ...        ...
...
1253 2023-12-22  21.389999  21.520000  21.230000  21.360001  21.360001
61200
```

```
1254 2023-12-26  21.570000  21.570000  21.350000  21.379999  21.379999
37500
1255 2023-12-27  21.420000  21.520000  21.420000  21.469999  21.469999
32500
1256 2023-12-28  21.629999  21.780001  21.500000  21.549999  21.549999
47500
1257 2023-12-29  21.559999  21.580000  21.469999  21.469999  21.469999
26000

[1258 rows x 7 columns]
```

Etract , Load and Transform the data

```
#Dropping the columns
df_data.drop(axis=1, columns=['Adj Close'], inplace=True)
df_data.drop(axis=1, columns=['Date'], inplace = True)

#Extract the features
df_data_features = df_data.iloc[: -1 , :]
df_data_features
```

```
           Open       High        Low      Close  Volume
0     19.910000  19.969999  19.760000  19.910000   59800
1     19.780001  19.780001  19.520000  19.600000   40800
2     19.830000  20.580000  19.809999  20.500000   19700
3     20.190001  20.190001  19.959999  20.150000   97500
4     19.990000  20.320000  19.990000  20.309999   25800
...         ...        ...        ...        ...     ...
1252  21.080000  21.290001  21.080000  21.230000   47900
1253  21.389999  21.520000  21.230000  21.360001   61200
1254  21.570000  21.570000  21.350000  21.379999   37500
1255  21.420000  21.520000  21.420000  21.469999   32500
1256  21.629999  21.780001  21.500000  21.549999   47500

[1257 rows x 5 columns]
```

```
#Extract only the open price from the 2nd day(i.e the next day) for
the prediction
df_data_target = df_data.iloc[1:,0].rename('Target')
df_data_target
```

```
1       19.780001
2       19.830000
3       20.190001
4       19.990000
5       20.240000
          ...
1253    21.389999
1254    21.570000
1255    21.420000
```

```
1256    21.629999
1257    21.559999
Name: Target, Length: 1257, dtype: float64
```

```
#Final data
df_data_features['Target'] = list(df_data_target)

first_col = df_data_features.pop('Target')
df_data_features.insert(0,'Target', first_col)

df_data_final = df_data_features

df_data_final
```

```
/tmp/ipykernel_8574/1034691346.py:2: SettingWithCopyWarning: 
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: 
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  df_data_features['Target'] = list(df_data_target)
```

```
         Target       Open       High        Low      Close  Volume
0     19.780001  19.910000  19.969999  19.760000  19.910000   59800
1     19.830000  19.780001  19.780001  19.520000  19.600000   40800
2     20.190001  19.830000  20.580000  19.809999  20.500000   19700
3     19.990000  20.190001  20.190001  19.959999  20.150000   97500
4     20.240000  19.990000  20.320000  19.990000  20.309999   25800
...         ...        ...        ...        ...        ...     ...
1252  21.389999  21.080000  21.290001  21.080000  21.230000   47900
1253  21.570000  21.389999  21.520000  21.230000  21.360001   61200
1254  21.420000  21.570000  21.570000  21.350000  21.379999   37500
1255  21.629999  21.420000  21.520000  21.420000  21.469999   32500
1256  21.559999  21.629999  21.780001  21.500000  21.549999   47500

[1257 rows x 6 columns]
```

Train Test SPlit

```
import numpy as np

df_randomized = df_data_final.sample(frac=1, random_state=123)

df_randomized
```

```
         Target       Open       High        Low      Close  Volume
870   21.920000  22.320000  22.510000  22.200001  22.379999   18400
367   15.480000  15.120000  15.450000  15.110000  15.410000   90500
875   22.010000  21.770000  21.990000  21.670000  21.900000   13200
```

```
525   22.850000  22.570000  22.760000  22.510000  22.750000   14900
1256  21.559999  21.629999  21.780001  21.500000  21.549999   47500
...         ...        ...        ...        ...        ...     ...
1238  23.120001  22.660000  22.969999  22.660000  22.950001   43100
1147  22.450001  22.360001  22.450001  22.350000  22.400000   39500
106   21.230000  21.570000  21.570000  20.980000  21.290001   48900
1041  20.040001  20.190001  20.190001  20.090000  20.090000   18800
1122  21.170000  21.219999  21.420000  21.219999  21.240000   41500

[1257 rows x 6 columns]

train_data, test_data = np.split(df_randomized,
[int(0.8*len(df_randomized))])
print(train_data.shape, test_data.shape)

(1005, 6) (252, 6)

/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages/
numpy/core/fromnumeric.py:57: FutureWarning: 'DataFrame.swapaxes' is
deprecated and will be removed in a future version. Please use
'DataFrame.transpose' instead.
  return bound(*args, **kwds)
```

Set a path upload data to S3 bucket

```
import os

prefix = 'xgboost-as-a-built-in-algo'

train_csv_path = 's3://{}/{}/{}/{}'.format(bucket_name,
prefix,'train','train.csv')
test_csv_path = 's3://{}/{}/{}/{}'.format(bucket_name,
prefix,'test','test.csv')

print(train_csv_path)
print(test_csv_path)

s3://yahoofinancestockprice-1/xgboost-as-a-built-in-algo/train/
train.csv
s3://yahoofinancestockprice-1/xgboost-as-a-built-in-algo/test/test.csv

train_data.to_csv(train_csv_path, index= False, header= False)
test_data.to_csv(test_csv_path, index= False, header= False)

/home/ec2-user/anaconda3/envs/python3/lib/python3.10/site-packages/
fsspec/registry.py:272: UserWarning: Your installed version of s3fs is
very old and known to cause
severe performance issues, see also
https://github.com/dask/dask/issues/10276

To fix, you should specify a lower version bound on s3fs, or
```

```
update the current installation.

  warnings.warn(s3_msg)
```

BUILD XGBOOST MODEL

```
#XGBOOST as a built in algo
import sagemaker
from sagemaker import image_uris
from sagemaker.session import Session
from sagemaker.inputs import TrainingInput


sagemaker.config INFO - Not applying SDK defaults from location:
/etc/xdg/sagemaker/config.yaml
sagemaker.config INFO - Not applying SDK defaults from location:
/home/ec2-user/.config/sagemaker/config.yaml
```

Find Xgboost image URI adn build an Xgboost cotainer

```
xgboost_container = image_uris.retrieve("xgboost", region,"1.5-1")

display(xgboost_container)

'662702820516.dkr.ecr.eu-north-1.amazonaws.com/sagemaker-xgboost:1.5-
1'
```

INitialize Hyperparameters

```
hyperparameters = {
        "max_depth":"5",
        "eta":"0.2",
        "gamma":"4",
        "min_child_weight":"6",
        "subsample":"0.7",
        "verbosity":"1",
        "objective":"reg:squarederror",
        "early_stopping_rounds" : 10,
        "num_round":50}
```

Output path to save the trained model

```
output_path = 's3://{}/{}/{}/'.format(bucket_name, prefix,'output')

print(output_path)

s3://yahoofinancestockprice-1/xgboost-as-a-built-in-algo/output/
```

Build a sagemaker estimator that calls Xgboost container

```
estimator = sagemaker.estimator.Estimator(image_uri=xgboost_container,


hyperparameters=hyperparameters,

role=sagemaker.get_execution_role(),
                                          instance_count=1,

instance_type='ml.m5.2xlarge',
                                          volume_size=5, # 5 GB
                                          output_path=output_path,
                                          use_spot_instances = True,
#wait until the next instance to some
                                          max_run = 300,
                                          max_wait = 600
                                         )

sagemaker.config INFO - Not applying SDK defaults from location:
/etc/xdg/sagemaker/config.yaml
sagemaker.config INFO - Not applying SDK defaults from location:
/home/ec2-user/.config/sagemaker/config.yaml
sagemaker.config INFO - Not applying SDK defaults from location:
/etc/xdg/sagemaker/config.yaml
sagemaker.config INFO - Not applying SDK defaults from location:
/home/ec2-user/.config/sagemaker/config.yaml
```

Defining data typea and paths for the training and validation dataset which was saved earlier

```
content_type ='csv'

train_input = TrainingInput("s3://{}/{}/{}/".format(bucket_name,
prefix, 'train'), content_type=content_type)
test_input = TrainingInput("s3://{}/{}/{}/".format(bucket_name,
prefix, 'test'), content_type=content_type)
```

Execution of Xgboost algo

```
# execute the XGBoost training job
estimator.fit({'train': train_input, 'validation': test_input})

INFO:sagemaker:Creating training-job with name: sagemaker-xgboost-
2024-01-04-19-24-47-568

2024-01-04 19:24:47 Starting - Starting the training job...
2024-01-04 19:25:02 Starting - Preparing the instances for
training.........
2024-01-04 19:26:21 Downloading - Downloading input data...
2024-01-04 19:27:06 Training - Training image download completed.
Training in
progress.../miniconda3/lib/python3.8/site-packages/xgboost/compat.py:3
```

```
6: FutureWarning: pandas.Int64Index is deprecated and will be removed
from pandas in a future version. Use pandas.Index with the appropriate
dtype instead.
  from pandas import MultiIndex, Int64Index
[2024-01-04 19:27:28.159 ip-10-0-245-0.eu-north-1.compute.internal:7
INFO utils.py:28] RULE_JOB_STOP_SIGNAL_FILENAME: None
[2024-01-04 19:27:28.180 ip-10-0-245-0.eu-north-1.compute.internal:7
INFO profiler_config_parser.py:111] User has disabled profiler.
[2024-01-04:19:27:28:INFO] Imported framework
sagemaker_xgboost_container.training
[2024-01-04:19:27:28:INFO] Failed to parse hyperparameter objective
value reg:squarederror to Json.
Returning the value itself
[2024-01-04:19:27:28:INFO] No GPUs detected (normal if no gpus
installed)
[2024-01-04:19:27:28:INFO] Running XGBoost Sagemaker in algorithm mode
[2024-01-04:19:27:28:INFO] Determined 0 GPU(s) available on the
instance.
[2024-01-04:19:27:28:INFO] Determined delimiter of CSV input is ','
[2024-01-04:19:27:28:INFO] Determined delimiter of CSV input is ','
[2024-01-04:19:27:28:INFO] files path: /opt/ml/input/data/train
[2024-01-04:19:27:28:INFO] Determined delimiter of CSV input is ','
[2024-01-04:19:27:28:INFO] files path: /opt/ml/input/data/validation
[2024-01-04:19:27:28:INFO] Determined delimiter of CSV input is ','
[2024-01-04:19:27:28:INFO] Single node training.
[2024-01-04:19:27:28:INFO] Train matrix has 1005 rows and 5 columns
[2024-01-04:19:27:28:INFO] Validation matrix has 252 rows
[2024-01-04 19:27:28.562 ip-10-0-245-0.eu-north-1.compute.internal:7
INFO json_config.py:92] Creating hook from json_config at
/opt/ml/input/config/debughookconfig.json.
[2024-01-04 19:27:28.563 ip-10-0-245-0.eu-north-1.compute.internal:7
INFO hook.py:206] tensorboard_dir has not been set for the hook.
SMDebug will not be exporting tensorboard summaries.
[2024-01-04 19:27:28.564 ip-10-0-245-0.eu-north-1.compute.internal:7
INFO hook.py:259] Saving to /opt/ml/output/tensors
[2024-01-04 19:27:28.564 ip-10-0-245-0.eu-north-1.compute.internal:7
INFO state_store.py:77] The checkpoint config file
/opt/ml/input/config/checkpointconfig.json does not exist.
[2024-01-04:19:27:28:INFO] Debug hook created from config
[2024-01-04 19:27:28.579 ip-10-0-245-0.eu-north-1.compute.internal:7
INFO hook.py:427] Monitoring the collections: metrics
[2024-01-04 19:27:28.582 ip-10-0-245-0.eu-north-1.compute.internal:7
INFO hook.py:491] Hook is writing from the hook with pid: 7
[0]#011train-rmse:17.07535#011validation-rmse:16.91446
[1]#011train-rmse:13.69300#011validation-rmse:13.56620
[2]#011train-rmse:10.97522#011validation-rmse:10.85885
[3]#011train-rmse:8.79980#011validation-rmse:8.69970
[4]#011train-rmse:7.05672#011validation-rmse:6.97936
[5]#011train-rmse:5.66078#011validation-rmse:5.59705
```

```
[6]#011train-rmse:4.54020#011validation-rmse:4.48828
[7]#011train-rmse:3.64397#011validation-rmse:3.60182
[8]#011train-rmse:2.92590#011validation-rmse:2.89110
[9]#011train-rmse:2.35254#011validation-rmse:2.32456
[10]#011train-rmse:1.89105#011validation-rmse:1.86864
[11]#011train-rmse:1.52754#011validation-rmse:1.51276
[12]#011train-rmse:1.23202#011validation-rmse:1.22250
[13]#011train-rmse:1.00156#011validation-rmse:0.99385
[14]#011train-rmse:0.81727#011validation-rmse:0.81215
[15]#011train-rmse:0.67565#011validation-rmse:0.67045
[16]#011train-rmse:0.56389#011validation-rmse:0.55758
[17]#011train-rmse:0.48040#011validation-rmse:0.47434
[18]#011train-rmse:0.41642#011validation-rmse:0.40788
[19]#011train-rmse:0.36872#011validation-rmse:0.36064
[20]#011train-rmse:0.33086#011validation-rmse:0.32451
[21]#011train-rmse:0.30761#011validation-rmse:0.30041
[22]#011train-rmse:0.29173#011validation-rmse:0.28382
[23]#011train-rmse:0.28143#011validation-rmse:0.27296
[24]#011train-rmse:0.27490#011validation-rmse:0.26603
[25]#011train-rmse:0.27065#011validation-rmse:0.26147
[26]#011train-rmse:0.26759#011validation-rmse:0.25816
[27]#011train-rmse:0.26602#011validation-rmse:0.25644
[28]#011train-rmse:0.26490#011validation-rmse:0.25520
[29]#011train-rmse:0.26400#011validation-rmse:0.25419
[30]#011train-rmse:0.26343#011validation-rmse:0.25353
[31]#011train-rmse:0.26312#011validation-rmse:0.25317
[32]#011train-rmse:0.26294#011validation-rmse:0.25295
[33]#011train-rmse:0.26280#011validation-rmse:0.25279
[34]#011train-rmse:0.26259#011validation-rmse:0.25251
[35]#011train-rmse:0.26255#011validation-rmse:0.25246
[36]#011train-rmse:0.26255#011validation-rmse:0.25246
[37]#011train-rmse:0.26253#011validation-rmse:0.25244
[38]#011train-rmse:0.26252#011validation-rmse:0.25242
[39]#011train-rmse:0.26250#011validation-rmse:0.25238
[40]#011train-rmse:0.25964#011validation-rmse:0.25252
[41]#011train-rmse:0.25962#011validation-rmse:0.25250
[42]#011train-rmse:0.25964#011validation-rmse:0.25252
[43]#011train-rmse:0.25620#011validation-rmse:0.24706
[44]#011train-rmse:0.25618#011validation-rmse:0.24702
[45]#011train-rmse:0.25618#011validation-rmse:0.24703
[46]#011train-rmse:0.25618#011validation-rmse:0.24704
[47]#011train-rmse:0.25618#011validation-rmse:0.24701
[48]#011train-rmse:0.25618#011validation-rmse:0.24701
[49]#011train-rmse:0.25618#011validation-rmse:0.24702

2024-01-04 19:27:43 Uploading - Uploading generated training model
2024-01-04 19:27:43 Completed - Training job completed
Training seconds: 82
```

```
Billable seconds: 30
Managed Spot Training savings: 63.4%
```

**Deploy the trained xgb model as Endpoint

```
from sagemaker.serializers import CSVSerializer

xgb_predictor = estimator.deploy(initial_instance_count=1,
instance_type = 'ml.m5.2xlarge', serializer = CSVSerializer()) #iic=1,
the number of instance deployed ,,, serializer - serialize input data
of various formats to a cSV format coz xgboots accepts only tat

INFO:sagemaker:Creating model with name: sagemaker-xgboost-2024-01-05-
06-40-57-245
INFO:sagemaker:Creating endpoint-config with name sagemaker-xgboost-
2024-01-05-06-40-57-245
INFO:sagemaker:Creating endpoint with name sagemaker-xgboost-2024-01-
05-06-40-57-245

----!

xgb_predictor.endpoint_name

'sagemaker-xgboost-2024-01-05-06-40-57-245'
```

**Make predictions with the use of Endpoints

```
start_date = datetime(2024,1,2)
end_date = datetime(2024,1,3)

#get the data
df_data = yf.download('IIF', start = start_date, end= end_date)

df_data.reset_index(inplace=True)

df_data

[*********************100%%**********************]  1 of 1 completed

        Date       Open   High        Low  Close  Adj Close  Volume
0 2024-01-02  21.459999  21.59  21.379999   21.4       21.4   68200

df_data.drop(axis=1, columns=['Adj Close'], inplace=True)
df_data.drop(axis=1, columns=['Date'], inplace = True)

data_features_array = df_data.values
data_features_array

array([[2.14599991e+01, 2.15900002e+01, 2.13799992e+01,
2.13999996e+01,
        6.82000000e+04]])
```

Serialize Data

```
#serialize input by sagemaker function
y_pred_fn = xgb_predictor.predict(data_features_array).decode('utf-8')
#utf-8 to convert bytes to string
y_pred_fn

'21.51511573791504\n'

#OR the same above code can also be written as
# from sagemaker.serialzer import CSVSErializer

###Serialized_ip_fn = CSVSerializer().serailize([[2.14599991e+01,
2.15900002e+01, 2.13799992e+01, 2.13999996e+01,
#         6.82000000e+04]])
# print(Serialized_ip_fn)

#y_pred_fn = xgb_predictor.predict(Serialized_ip_fn).decode('utf-8')
#utf-8 to convert bytes to string
#y_pred_fn
#it would give the same output of 21.5...

#serialize input by built in fn(Lambda fn)

Input = [[2.14599991e+01, 2.15900002e+01, 2.13799992e+01,
2.13999996e+01,
        6.82000000e+04],
        [2.14599991e+01, 2.15900002e+01, 2.13799992e+01,
2.13999996e+01,
        6.82000000e+04],
        [2.14599991e+01, 2.15900002e+01, 2.13799992e+01,
2.13999996e+01,
        6.82000000e+04]]

serialized_input = ','.join(map(str, Input[0])) #take the first
element from the list as str and join them with the ,

print(serialized_input, type(serialized_input))

y_pred_fn = xgb_predictor.predict(serialized_input).decode('utf-8')
#utf-8 to convert bytes to string
y_pred_fn

21.4599991,21.5900002,21.3799992,21.3999996,68200.0 <class 'str'>

'21.51511573791504\n'
```

**Lambda function to invoke the endpoint

1. trigger Endpoint

2. Trigger SNS With lambda you can run code virtually any type of application or backend service

**Lambda function handler

```
import boto3

ENDPOINT_NAME = 'sagemaker-xgboost-2024-01-05-06-40-57-245' #We got
this in the previous cells
runtime = boto3.client('runtime.sagemaker')

def lambda_handler(event, context):

    inputs = event['data']
    result = []

    for input in inputs:
        serialized_input = ','.join(map(str, input)) #take the first
element from the list as str and join them with the ,

        response = runtime.invoke_endpoint(EndpointName=
ENDPOINT_NAME,
                                      ContentType = 'text/csv',
                                      Body= serialized_input)

        result.append(response['Body'].read().decode())

    return result

Input_json = {'data':
             [[2.14599991e+01, 2.15900002e+01, 2.13799992e+01,
2.13999996e+01,
        6.82000000e+04],
        [2.14599991e+01, 2.15900002e+01, 2.13799992e+01,
2.13999996e+01,
        6.82000000e+04],
        [2.14599991e+01, 2.15900002e+01, 2.13799992e+01,
2.13999996e+01,
        6.82000000e+04]]
             }

result = lambda_handler(Input_json, _)
result

['21.51511573791504\n', '21.51511573791504\n', '21.51511573791504\n']
```

**Post Request

```
import requests
```

```
API_ENDPOINT = "https://qr94phmnc3.execute-api.us-west-2.amazonaws.com/Xgboostmodel"

json = {"data":[[2.14599991e+01, 2.15900002e+01, 2.13799992e+01, 2.13999996e+01,
        6.82000000e+04],
        [2.14599991e+01, 2.15900002e+01, 2.13799992e+01, 2.13999996e+01,
        6.82000000e+04],
        [2.14599991e+01, 2.15900002e+01, 2.13799992e+01, 2.13999996e+01,
        6.82000000e+04]]
       }

r = requests.post(url = API_ENDPOINT,  json =json)

print(f"Status Code :{r.status_code}, Response : {r.json()} " )
```

```
Status Code :200, Response : ['21.51511573791504\n', '21.51511573791504\n', '21.51511573791504\n']
```

**Close and Terminate

```
#sagemaker.Session().delete_endpoint(xgb_predictor.endpoint)

#bucket_to_delete = boto3.resource('s3').Bucket(bucket_name)
#bucket_to_delete.objects.all().delete()
```