Train an Un-Supervised Random cut forest model using the data stored in feature store

Setting up Sagemaker and feature store session

In [1]:
```python
# importing all the required libraries
import boto3
import sagemaker
from sagemaker.session import Session

#fetching the session region
region = boto3.Session().region_name

# creating a boto session
boto_session = boto3.Session(region_name=region)

# creating sagemaker and feature store sessions
sagemaker_client = boto_session.client(service_name='sagemaker', region_name=region)
featurestore_runtime = boto_session.client(service_name='sagemaker-featurestore-runtime', region_name=region)

feature_store_session = Session(
    boto_session=boto_session,
    sagemaker_client=sagemaker_client,
    sagemaker_featurestore_runtime_client=featurestore_runtime
)
```

getting the feature group

In [2]:
```python
%store -r
```

In [3]:
```python
from sagemaker.feature_store.feature_group import FeatureGroup

# Fetching data from feature group
fd_feature_group_name = 'transactionfeaturegroup'
fd_feature_group = FeatureGroup(name=fd_feature_group_name, sagemaker_session=feature_store_session)
```

setting up the Training Dataset

In [4]:
```python
# using the defualt bucket
default_s3_bucket_name = feature_store_session.default_bucket()
prefix = 'sagemaker-featurestore'

print(default_s3_bucket_name)
```

```
sagemaker-ap-south-1-080451317723
```

In [5]:
```python
# running athena query to get all the data from the feature group tables
transaction_query = fd_feature_group.athena_query()

transaction_table = transaction_query.table_name

print(transaction_table)

query_string = 'SELECT * FROM "'+transaction_table+'"'
print('Running ' + query_string)

# running the query and storing the results into data set variable as pandas dataframe
transaction_query.run(query_string=query_string, output_location='s3://'+default_s3_bucket_name+'/'+prefix+'/query_r
esults/')
transaction_query.wait()
dataset = transaction_query.as_dataframe()

dataset
```

```
transactionfeaturegroup-1639487824
Running SELECT * FROM "transactionfeaturegroup-1639487824"
```

Out[5]:

	time	v1	v2	v3	v4	v5	v6	v7	v8	v9	...	v26	v27	v28	amount
0	151287.0	0.088863	0.729612	-0.366984	-0.731235	1.604282	0.720851	0.896757	0.155347	0.183897	...	0.183513	0.105804	-0.150822	9.27
1	40667.0	1.355078	-0.724511	1.368546	-0.626317	-1.727245	-0.514134	-1.291331	0.010314	-0.250836	...	-0.044516	0.061437	0.034151	9.99
2	124341.0	0.084587	1.026924	-0.211915	-0.437726	0.767183	-1.021523	0.973648	-0.126873	-0.117500	...	0.088517	0.226348	0.089582	4.49
3	72605.0	1.106382	-0.182073	-1.692983	-1.465153	0.525804	-1.657673	1.336612	-0.626055	-0.168914	...	0.488529	-0.094244	-0.018595	128.81
4	72347.0	-1.177816	1.316669	1.057519	1.570301	0.717973	-1.039204	0.806413	0.251177	-1.823729	...	-0.009782	-0.206820	-0.162201	6.05
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
99995	125409.0	2.010149	0.093008	-1.612224	0.337610	0.405446	-0.576773	0.062787	-0.067284	0.241324	...	0.148692	-0.064047	-0.039637	1.98
99996	61555.0	-1.786858	2.220376	-0.354501	-0.114401	-0.535980	-0.685306	-0.326123	1.226695	-0.590223	...	0.137403	0.097970	0.008434	11.99
99997	74192.0	-0.565672	1.352774	0.937847	0.861238	0.063604	-0.343165	0.398966	0.365444	-0.916404	...	-0.304634	0.070989	0.099293	1.50
99998	21846.0	-1.396917	-0.649990	1.450089	-1.704585	-2.496589	1.599768	1.568052	-0.189034	0.630655	...	-0.054341	0.391464	-0.031456	544.01
99999	167032.0	-0.661520	-0.722317	1.439445	-2.110130	-0.289698	-0.951161	-0.305190	0.064561	-0.504968	...	-0.172078	0.098582	0.157863	50.00

100000 rows × 36 columns

In [6]:
```python
# selecting the useful columns for our dataset
dataset = dataset[['time', 'v1', 'v2', 'v3', 'v4', 'v5', 'v6', 'v7', 'v8', 'v9', 'v10',
       'v11', 'v12', 'v13', 'v14', 'v15', 'v16', 'v17', 'v18', 'v19', 'v20',
       'v21', 'v22', 'v23', 'v24', 'v25', 'v26', 'v27', 'v28', 'amount',
       'class']]
```

In [7]:
```python
# dropping all the NANs
dataset = dataset.dropna()
```

In [8]:
```python
# finding out the total number of cases by classes
nonfrauds, frauds = dataset.groupby('class').size()
print('Number of frauds: ', frauds)
print('Number of non-frauds: ', nonfrauds)
print('Percentage of fradulent data:', 100.*frauds/(frauds + nonfrauds))
```

```
Number of frauds:  172
Number of non-frauds:  99828
Percentage of fradulent data: 0.172
```

In [9]:
```python
#segregating the feature and label columns
feature_columns = dataset.columns[:-1]
label_column = dataset.columns[-1]

# storing the columns separately
features = dataset[feature_columns].values.astype('float32')
labels = (dataset[label_column].values).astype('float32')
```

In [10]:
```python
from sklearn.model_selection import train_test_split
# dividing the data into test and train splits
X_train, X_test, y_train, y_test = train_test_split(
    features, labels, test_size=0.1, random_state=42)
```

Training the data

In [11]:
```python
import os
import sagemaker
# creating the sagemaker sessions
session = sagemaker.Session()
bucket = default_s3_bucket_name
prefix = 'rcf-fraud-classifier'
```

In [12]:
```python
# fetching the IAM role
sagemaker_iam_role = sagemaker.get_execution_role()
```

In [13]:
```python
from sagemaker import RandomCutForest

# initialising the parameters for the training job
rcf = RandomCutForest(role=sagemaker_iam_role,
                      instance_count=1,
                      instance_type="ml.m5.xlarge",
                      data_location='s3://{}/{}/'.format(bucket, prefix),
                      output_path='s3://{}/{}/output'.format(bucket, prefix),
                      base_job_name="{}-rcf".format("fraud-detection"),
                      num_samples_per_tree=512,
                      num_trees=50)
```

In [14]:
```python
# fitting out data to the model
rcf.fit(rcf.record_set(X_train))
```

```
Defaulting to the only supported framework/algorithm version: 1. Ignoring framework/algorithm version: 1.
```

```
2021-12-14 17:01:52 Starting - Starting the training job...
2021-12-14 17:01:54 Starting - Launching requested ML instancesProfilerReport-1639501312: InProgress
...
2021-12-14 17:02:43 Starting - Preparing the instances for training......
2021-12-14 17:03:47 Downloading - Downloading input data
2021-12-14 17:03:47 Training - Downloading the training image...
2021-12-14 17:04:22 Training - Training image download completed. Training in progress..Docker entrypoint called with
argument(s): train
Running default environment configuration script
[12/14/2021 17:04:24 INFO 139672757847872] Reading default configuration from /opt/amazon/lib/python3.7/site-package
s/algorithm/resources/default-conf.json: {'num_samples_per_tree': 256, 'num_trees': 100, 'force_dense': 'true', 'eval
_metrics': ['accuracy', 'precision_recall_fscore'], 'epochs': 1, 'mini_batch_size': 1000, '_log_level': 'info', '_kvs
tore': 'dist_async', '_num_kv_servers': 'auto', '_num_gpus': 'auto', '_tuning_objective_metric': '', '_ftp_port': 899
9}
[12/14/2021 17:04:24 INFO 139672757847872] Merging with provided configuration from /opt/ml/input/config/hyperparamet
ers.json: {'num_trees': '50', 'num_samples_per_tree': '512', 'feature_dim': '30', 'mini_batch_size': '1000'}
[12/14/2021 17:04:24 INFO 139672757847872] Final configuration: {'num_samples_per_tree': '512', 'num_trees': '50', 'f
orce_dense': 'true', 'eval_metrics': ['accuracy', 'precision_recall_fscore'], 'epochs': 1, 'mini_batch_size': '1000',
'_log_level': 'info', '_kvstore': 'dist_async', '_num_kv_servers': 'auto', '_num_gpus': 'auto', '_tuning_objective_me
tric': '', '_ftp_port': 8999, 'feature_dim': '30'}
[12/14/2021 17:04:24 WARNING 139672757847872] Loggers have already been setup.
[12/14/2021 17:04:24 INFO 139672757847872] Launching parameter server for role scheduler
[12/14/2021 17:04:24 INFO 139672757847872] {'ENVROOT': '/opt/amazon', 'PROTOCOL_BUFFERS_PYTHON_IMPLEMENTATION': 'cp
p', 'HOSTNAME': 'ip-10-0-114-160.ap-south-1.compute.internal', 'TRAINING_JOB_NAME': 'fraud-detection-rcf-2021-12-14-1
7-01-52-095', 'NVIDIA_REQUIRE_CUDA': 'cuda>=9.0', 'TRAINING_JOB_ARN': 'arn:aws:sagemaker:ap-south-1:080451317723:trai
ning-job/fraud-detection-rcf-2021-12-14-17-01-52-095', 'AWS_CONTAINER_CREDENTIALS_RELATIVE_URI': '/v2/credentials/b48
9cf5c-c0a1-4a94-80d3-8a7c8ac2b1c1', 'CANONICAL_ENVROOT': '/opt/amazon', 'PYTHONUNBUFFERED': 'TRUE', 'NVIDIA_VISIBLE_D
EVICES': 'void', 'LD_LIBRARY_PATH': '/opt/amazon/lib/python3.7/site-packages/cv2/../../../../lib:/usr/local/nvidia/li
b64:/opt/amazon/lib', 'MXNET_KVSTORE_BIGARRAY_BOUND': '400000000', 'NVIDIA_DRIVER_CAPABILITIES': 'compute,utility',
 'AWS_EXECUTION_ENV': 'AWS_ECS_EC2', 'PATH': '/opt/amazon/bin:/usr/local/nvidia/bin:/usr/local/sbin:/usr/local/bin:/u
sr/sbin:/usr/bin:/sbin:/bin:/opt/amazon/bin:/opt/amazon/bin', 'PWD': '/', 'LANG': 'en_US.utf8', 'SAGEMAKER_METRICS_DI
RECTORY': '/opt/ml/output/metrics/sagemaker', 'AWS_REGION': 'ap-south-1', 'HOME': '/root', 'SHLVL': '1', 'PROTOCOL_BU
FFERS_PYTHON_IMPLEMENTATION_VERSION': '2', 'OMP_NUM_THREADS': '2', 'ECS_CONTAINER_METADATA_URI': 'http://169.254.170.
2/v3/61159bdc-a8cf-4a21-96c9-efca6066a6bf', 'DMLC_INTERFACE': 'eth0', 'ECS_CONTAINER_METADATA_URI_V4': 'http://169.25
4.170.2/v4/61159bdc-a8cf-4a21-96c9-efca6066a6bf', 'SAGEMAKER_HTTP_PORT': '8080', 'SAGEMAKER_DATA_PATH': '/opt/ml', 'K
MP_DUPLICATE_LIB_OK': 'True', 'KMP_INIT_AT_FORK': 'FALSE'}
[12/14/2021 17:04:24 INFO 139672757847872] envs={'ENVROOT': '/opt/amazon', 'PROTOCOL_BUFFERS_PYTHON_IMPLEMENTATION':
 'cpp', 'HOSTNAME': 'ip-10-0-114-160.ap-south-1.compute.internal', 'TRAINING_JOB_NAME': 'fraud-detection-rcf-2021-12-
14-17-01-52-095', 'NVIDIA_REQUIRE_CUDA': 'cuda>=9.0', 'TRAINING_JOB_ARN': 'arn:aws:sagemaker:ap-south-1:080451317723:
training-job/fraud-detection-rcf-2021-12-14-17-01-52-095', 'AWS_CONTAINER_CREDENTIALS_RELATIVE_URI': '/v2/credential
s/b489cf5c-c0a1-4a94-80d3-8a7c8ac2b1c1', 'CANONICAL_ENVROOT': '/opt/amazon', 'PYTHONUNBUFFERED': 'TRUE', 'NVIDIA_VISI
BLE_DEVICES': 'void', 'LD_LIBRARY_PATH': '/opt/amazon/lib/python3.7/site-packages/cv2/../../../../lib:/usr/local/nvid
ia/lib64:/opt/amazon/lib', 'MXNET_KVSTORE_BIGARRAY_BOUND': '400000000', 'NVIDIA_DRIVER_CAPABILITIES': 'compute,utilit
y', 'AWS_EXECUTION_ENV': 'AWS_ECS_EC2', 'PATH': '/opt/amazon/bin:/usr/local/nvidia/bin:/usr/local/sbin:/usr/local/bi
n:/usr/sbin:/usr/bin:/sbin:/bin:/opt/amazon/bin:/opt/amazon/bin', 'PWD': '/', 'LANG': 'en_US.utf8', 'SAGEMAKER_METRIC
S_DIRECTORY': '/opt/ml/output/metrics/sagemaker', 'AWS_REGION': 'ap-south-1', 'HOME': '/root', 'SHLVL': '1', 'PROTOCO
L_BUFFERS_PYTHON_IMPLEMENTATION_VERSION': '2', 'OMP_NUM_THREADS': '2', 'ECS_CONTAINER_METADATA_URI': 'http://169.254.
170.2/v3/61159bdc-a8cf-4a21-96c9-efca6066a6bf', 'DMLC_INTERFACE': 'eth0', 'ECS_CONTAINER_METADATA_URI_V4': 'http://16
9.254.170.2/v4/61159bdc-a8cf-4a21-96c9-efca6066a6bf', 'SAGEMAKER_HTTP_PORT': '8080', 'SAGEMAKER_DATA_PATH': '/opt/m
l', 'KMP_DUPLICATE_LIB_OK': 'True', 'KMP_INIT_AT_FORK': 'FALSE', 'DMLC_ROLE': 'scheduler', 'DMLC_PS_ROOT_URI': '10.0.
114.160', 'DMLC_PS_ROOT_PORT': '9000', 'DMLC_NUM_SERVER': '1', 'DMLC_NUM_WORKER': '1'}
[12/14/2021 17:04:24 INFO 139672757847872] Launching parameter server for role server
[12/14/2021 17:04:24 INFO 139672757847872] {'ENVROOT': '/opt/amazon', 'PROTOCOL_BUFFERS_PYTHON_IMPLEMENTATION': 'cp
p', 'HOSTNAME': 'ip-10-0-114-160.ap-south-1.compute.internal', 'TRAINING_JOB_NAME': 'fraud-detection-rcf-2021-12-14-1
7-01-52-095', 'NVIDIA_REQUIRE_CUDA': 'cuda>=9.0', 'TRAINING_JOB_ARN': 'arn:aws:sagemaker:ap-south-1:080451317723:trai
ning-job/fraud-detection-rcf-2021-12-14-17-01-52-095', 'AWS_CONTAINER_CREDENTIALS_RELATIVE_URI': '/v2/credentials/b48
9cf5c-c0a1-4a94-80d3-8a7c8ac2b1c1', 'CANONICAL_ENVROOT': '/opt/amazon', 'PYTHONUNBUFFERED': 'TRUE', 'NVIDIA_VISIBLE_D
EVICES': 'void', 'LD_LIBRARY_PATH': '/opt/amazon/lib/python3.7/site-packages/cv2/../../../../lib:/usr/local/nvidia/li
b64:/opt/amazon/lib', 'MXNET_KVSTORE_BIGARRAY_BOUND': '400000000', 'NVIDIA_DRIVER_CAPABILITIES': 'compute,utility',
 'AWS_EXECUTION_ENV': 'AWS_ECS_EC2', 'PATH': '/opt/amazon/bin:/usr/local/nvidia/bin:/usr/local/sbin:/usr/local/bin:/u
sr/sbin:/usr/bin:/sbin:/bin:/opt/amazon/bin:/opt/amazon/bin', 'PWD': '/', 'LANG': 'en_US.utf8', 'SAGEMAKER_METRICS_DI
RECTORY': '/opt/ml/output/metrics/sagemaker', 'AWS_REGION': 'ap-south-1', 'HOME': '/root', 'SHLVL': '1', 'PROTOCOL_BU
FFERS_PYTHON_IMPLEMENTATION_VERSION': '2', 'OMP_NUM_THREADS': '2', 'ECS_CONTAINER_METADATA_URI': 'http://169.254.170.
2/v3/61159bdc-a8cf-4a21-96c9-efca6066a6bf', 'DMLC_INTERFACE': 'eth0', 'ECS_CONTAINER_METADATA_URI_V4': 'http://169.25
4.170.2/v4/61159bdc-a8cf-4a21-96c9-efca6066a6bf', 'SAGEMAKER_HTTP_PORT': '8080', 'SAGEMAKER_DATA_PATH': '/opt/ml', 'K
MP_DUPLICATE_LIB_OK': 'True', 'KMP_INIT_AT_FORK': 'FALSE'}
[12/14/2021 17:04:24 INFO 139672757847872] envs={'ENVROOT': '/opt/amazon', 'PROTOCOL_BUFFERS_PYTHON_IMPLEMENTATION':
 'cpp', 'HOSTNAME': 'ip-10-0-114-160.ap-south-1.compute.internal', 'TRAINING_JOB_NAME': 'fraud-detection-rcf-2021-12-
14-17-01-52-095', 'NVIDIA_REQUIRE_CUDA': 'cuda>=9.0', 'TRAINING_JOB_ARN': 'arn:aws:sagemaker:ap-south-1:080451317723:
training-job/fraud-detection-rcf-2021-12-14-17-01-52-095', 'AWS_CONTAINER_CREDENTIALS_RELATIVE_URI': '/v2/credential
s/b489cf5c-c0a1-4a94-80d3-8a7c8ac2b1c1', 'CANONICAL_ENVROOT': '/opt/amazon', 'PYTHONUNBUFFERED': 'TRUE', 'NVIDIA_VISI
BLE_DEVICES': 'void', 'LD_LIBRARY_PATH': '/opt/amazon/lib/python3.7/site-packages/cv2/../../../../lib:/usr/local/nvid
ia/lib64:/opt/amazon/lib', 'MXNET_KVSTORE_BIGARRAY_BOUND': '400000000', 'NVIDIA_DRIVER_CAPABILITIES': 'compute,utilit
y', 'AWS_EXECUTION_ENV': 'AWS_ECS_EC2', 'PATH': '/opt/amazon/bin:/usr/local/nvidia/bin:/usr/local/sbin:/usr/local/bi
n:/usr/sbin:/usr/bin:/sbin:/bin:/opt/amazon/bin:/opt/amazon/bin', 'PWD': '/', 'LANG': 'en_US.utf8', 'SAGEMAKER_METRIC
S_DIRECTORY': '/opt/ml/output/metrics/sagemaker', 'AWS_REGION': 'ap-south-1', 'HOME': '/root', 'SHLVL': '1', 'PROTOCO
L_BUFFERS_PYTHON_IMPLEMENTATION_VERSION': '2', 'OMP_NUM_THREADS': '2', 'ECS_CONTAINER_METADATA_URI': 'http://169.254.
170.2/v3/61159bdc-a8cf-4a21-96c9-efca6066a6bf', 'DMLC_INTERFACE': 'eth0', 'ECS_CONTAINER_METADATA_URI_V4': 'http://16
9.254.170.2/v4/61159bdc-a8cf-4a21-96c9-efca6066a6bf', 'SAGEMAKER_HTTP_PORT': '8080', 'SAGEMAKER_DATA_PATH': '/opt/m
l', 'KMP_DUPLICATE_LIB_OK': 'True', 'KMP_INIT_AT_FORK': 'FALSE', 'DMLC_ROLE': 'server', 'DMLC_PS_ROOT_URI': '10.0.11
4.160', 'DMLC_PS_ROOT_PORT': '9000', 'DMLC_NUM_SERVER': '1', 'DMLC_NUM_WORKER': '1'}
[12/14/2021 17:04:24 INFO 139672757847872] Environment: {'ENVROOT': '/opt/amazon', 'PROTOCOL_BUFFERS_PYTHON_IMPLEMENT
ATION': 'cpp', 'HOSTNAME': 'ip-10-0-114-160.ap-south-1.compute.internal', 'TRAINING_JOB_NAME': 'fraud-detection-rcf-2
021-12-14-17-01-52-095', 'NVIDIA_REQUIRE_CUDA': 'cuda>=9.0', 'TRAINING_JOB_ARN': 'arn:aws:sagemaker:ap-south-1:080451
317723:training-job/fraud-detection-rcf-2021-12-14-17-01-52-095', 'AWS_CONTAINER_CREDENTIALS_RELATIVE_URI': '/v2/cred
entials/b489cf5c-c0a1-4a94-80d3-8a7c8ac2b1c1', 'CANONICAL_ENVROOT': '/opt/amazon', 'PYTHONUNBUFFERED': 'TRUE', 'NVIDI
A_VISIBLE_DEVICES': 'void', 'LD_LIBRARY_PATH': '/opt/amazon/lib/python3.7/site-packages/cv2/../../../../lib:/usr/loca
l/nvidia/lib64:/opt/amazon/lib', 'MXNET_KVSTORE_BIGARRAY_BOUND': '400000000', 'NVIDIA_DRIVER_CAPABILITIES': 'compute,
utility', 'AWS_EXECUTION_ENV': 'AWS_ECS_EC2', 'PATH': '/opt/amazon/bin:/usr/local/nvidia/bin:/usr/local/sbin:/usr/loc
al/bin:/usr/sbin:/usr/bin:/sbin:/bin:/opt/amazon/bin:/opt/amazon/bin', 'PWD': '/', 'LANG': 'en_US.utf8', 'SAGEMAKER_M
ETRICS_DIRECTORY': '/opt/ml/output/metrics/sagemaker', 'AWS_REGION': 'ap-south-1', 'HOME': '/root', 'SHLVL': '1', 'PR
OTOCOL_BUFFERS_PYTHON_IMPLEMENTATION_VERSION': '2', 'OMP_NUM_THREADS': '2', 'ECS_CONTAINER_METADATA_URI': 'http://16
9.254.170.2/v3/61159bdc-a8cf-4a21-96c9-efca6066a6bf', 'DMLC_INTERFACE': 'eth0', 'ECS_CONTAINER_METADATA_URI_V4': 'htt
p://169.254.170.2/v4/61159bdc-a8cf-4a21-96c9-efca6066a6bf', 'SAGEMAKER_HTTP_PORT': '8080', 'SAGEMAKER_DATA_PATH': '/o
pt/ml', 'KMP_DUPLICATE_LIB_OK': 'True', 'KMP_INIT_AT_FORK': 'FALSE', 'DMLC_ROLE': 'worker', 'DMLC_PS_ROOT_URI': '10.
0.114.160', 'DMLC_PS_ROOT_PORT': '9000', 'DMLC_NUM_SERVER': '1', 'DMLC_NUM_WORKER': '1'}
Process 35 is a shell:scheduler.
Process 44 is a shell:server.
Process 1 is a worker.
[12/14/2021 17:04:24 INFO 139672757847872] Using default worker.
[12/14/2021 17:04:24 INFO 139672757847872] Loaded iterator creator application/x-recordio-protobuf for content type
 ('application/x-recordio-protobuf', '1.0')
[12/14/2021 17:04:24 INFO 139672757847872] Checkpoint loading and saving are disabled.
[12/14/2021 17:04:24 INFO 139672757847872] Verifying hyperparamemters...
[12/14/2021 17:04:24 INFO 139672757847872] Hyperparameters are correct.
[12/14/2021 17:04:24 INFO 139672757847872] Validating that feature_dim agrees with dimensions in training data...
[12/14/2021 17:04:24 INFO 139672757847872] feature_dim is correct.
[12/14/2021 17:04:24 INFO 139672757847872] Validating memory limits...
[12/14/2021 17:04:24 INFO 139672757847872] Available memory in bytes: 14793555968
[12/14/2021 17:04:24 INFO 139672757847872] Estimated sample size in bytes: 6144000
[12/14/2021 17:04:24 INFO 139672757847872] Estimated memory needed to build the forest in bytes: 30720000
[12/14/2021 17:04:24 INFO 139672757847872] Memory limits validated.
[12/14/2021 17:04:24 INFO 139672757847872] Starting cluster sharing facilities...
[12/14/2021 17:04:24 INFO 139671237543680] concurrency model: async
[12/14/2021 17:04:24 INFO 139671237543680] masquerade (NAT) address: None
[12/14/2021 17:04:24 INFO 139671237543680] passive ports: None
[12/14/2021 17:04:24 INFO 139671237543680] >>> starting FTP server on 0.0.0.0:8999, pid=1 <<<
[12/14/2021 17:04:24 INFO 139672757847872] Create Store: dist_async
[12/14/2021 17:04:25 INFO 139672757847872] Cluster sharing facilities started.
[12/14/2021 17:04:25 INFO 139672757847872] Verifying all workers are accessible...
[12/14/2021 17:04:25 INFO 139672757847872] All workers accessible.
[12/14/2021 17:04:25 INFO 139672757847872] Initializing Sampler...
[12/14/2021 17:04:25 INFO 139672757847872] Sampler correctly initialized.
#metrics {"StartTime": 1639501464.5404818, "EndTime": 1639501465.6693044, "Dimensions": {"Algorithm": "RandomCutFores
t", "Host": "algo-1", "Operation": "training"}, "Metrics": {"initialize.time": {"sum": 1123.1274604797363, "count":
 1, "min": 1123.1274604797363, "max": 1123.1274604797363}}}
#metrics {"StartTime": 1639501465.6694782, "EndTime": 1639501465.669517, "Dimensions": {"Algorithm": "RandomCutFores
t", "Host": "algo-1", "Operation": "training", "Meta": "init_train_data_iter"}, "Metrics": {"Total Records Seen": {"s
um": 0.0, "count": 1, "min": 0, "max": 0}, "Total Batches Seen": {"sum": 0.0, "count": 1, "min": 0, "max": 0}, "Max R
ecords Seen Between Resets": {"sum": 0.0, "count": 1, "min": 0, "max": 0}, "Max Batches Seen Between Resets": {"sum":
0.0, "count": 1, "min": 0, "max": 0}, "Reset Count": {"sum": 0.0, "count": 1, "min": 0, "max": 0}, "Number of Records
Since Last Reset": {"sum": 0.0, "count": 1, "min": 0, "max": 0}, "Number of Batches Since Last Reset": {"sum": 0.0,
 "count": 1, "min": 0, "max": 0}}}
[2021-12-14 17:04:25.670] [tensorio] [info] epoch_stats={"data_pipeline": "/opt/ml/input/data/train", "epoch": 0, "du
ration": 1129, "num_examples": 1, "num_bytes": 148000}
[12/14/2021 17:04:25 INFO 139672757847872] Sampling training data...
[2021-12-14 17:04:25.765] [tensorio] [info] epoch_stats={"data_pipeline": "/opt/ml/input/data/train", "epoch": 1, "du
ration": 94, "num_examples": 90, "num_bytes": 13320000}
[12/14/2021 17:04:25 INFO 139672757847872] Sampling training data completed.
#metrics {"StartTime": 1639501465.669432, "EndTime": 1639501465.7765176, "Dimensions": {"Algorithm": "RandomCutFores
t", "Host": "algo-1", "Operation": "training"}, "Metrics": {"epochs": {"sum": 1.0, "count": 1, "min": 1, "max": 1},
 "update.time": {"sum": 106.25576972961426, "count": 1, "min": 106.25576972961426, "max": 106.25576972961426}}}
[12/14/2021 17:04:25 INFO 139672757847872] Early stop condition met. Stopping training.
[12/14/2021 17:04:25 INFO 139672757847872] #progress_metric: host=algo-1, completed 100 % epochs
#metrics {"StartTime": 1639501465.6702247, "EndTime": 1639501465.7768385, "Dimensions": {"Algorithm": "RandomCutFores
t", "Host": "algo-1", "Operation": "training", "epoch": 0, "Meta": "training_data_iter"}, "Metrics": {"Total Records
 Seen": {"sum": 90000.0, "count": 1, "min": 90000, "max": 90000}, "Total Batches Seen": {"sum": 90.0, "count": 1, "mi
n": 90, "max": 90}, "Max Records Seen Between Resets": {"sum": 90000.0, "count": 1, "min": 90000, "max": 90000}, "Max
Batches Seen Between Resets": {"sum": 90.0, "count": 1, "min": 90, "max": 90}, "Reset Count": {"sum": 1.0, "count":
 1, "min": 1, "max": 1}, "Number of Records Since Last Reset": {"sum": 90000.0, "count": 1, "min": 90000, "max": 9000
0}, "Number of Batches Since Last Reset": {"sum": 90.0, "count": 1, "min": 90, "max": 90}}}
[12/14/2021 17:04:25 INFO 139672757847872] #throughput_metric: host=algo-1, train throughput=843249.444888732 record
s/second
[12/14/2021 17:04:25 INFO 139672757847872] Master node: building Random Cut Forest...
[12/14/2021 17:04:25 INFO 139672757847872] Gathering samples...
[12/14/2021 17:04:25 INFO 139672757847872] 25600 samples gathered
[12/14/2021 17:04:25 INFO 139672757847872] Building Random Cut Forest...
[12/14/2021 17:04:25 INFO 139672757847872] Random Cut Forest built:
ForestInfo{num_trees: 50, num_samples_in_forest: 25600, num_samples_per_tree: 512, sample_dim: 30, shingle_size: 1, t
rees_num_nodes: [879, 913, 897, 887, 903, 893, 917, 885, 895, 909, 893, 907, 903, 881, 897, 903, 895, 911, 925, 905,
 911, 897, 905, 929, 911, 897, 921, 897, 901, 917, 895, 907, 903, 893, 919, 895, 903, 915, 921, 905, 919, 907, 905, 9
11, 901, 901, 883, 907, 901, 879, ], trees_depth: [20, 20, 23, 23, 17, 21, 23, 21, 25, 21, 19, 20, 20, 17, 18, 21, 2
3, 22, 20, 22, 19, 19, 21, 24, 23, 20, 21, 18, 26, 18, 20, 20, 18, 21, 23, 21, 20, 26, 26, 21, 21, 19, 21, 21, 22, 1
9, 20, 20, 20, 24, ], max_num_nodes: 929, min_num_nodes: 879, avg_num_nodes: 903, max_tree_depth: 26, min_tree_depth:
17, avg_tree_depth: 20, mem_size: 25647920}
#metrics {"StartTime": 1639501465.7765963, "EndTime": 1639501465.85027, "Dimensions": {"Algorithm": "RandomCutFores
t", "Host": "algo-1", "Operation": "training"}, "Metrics": {"fit_model.time": {"sum": 53.44343185424805, "count": 1,
 "min": 53.44343185424805, "max": 53.44343185424805}, "model.bytes": {"sum": 25647920.0, "count": 1, "min": 25647920,
"max": 25647920}, "finalize.time": {"sum": 73.14705848693848, "count": 1, "min": 73.14705848693848, "max": 73.1470584
8693848}}}
[12/14/2021 17:04:26 INFO 139672757847872] Master node: Serializing the RandomCutForest model
#metrics {"StartTime": 1639501465.8504066, "EndTime": 1639501466.5402968, "Dimensions": {"Algorithm": "RandomCutFores
t", "Host": "algo-1", "Operation": "training"}, "Metrics": {"serialize_model.time": {"sum": 689.8391246795654, "coun
t": 1, "min": 689.8391246795654, "max": 689.8391246795654}}}
[12/14/2021 17:04:26 INFO 139672757847872] Test data is not provided.
#metrics {"StartTime": 1639501466.5403695, "EndTime": 1639501466.5404856, "Dimensions": {"Algorithm": "RandomCutFores
t", "Host": "algo-1", "Operation": "training"}, "Metrics": {"setuptime": {"sum": 22.64261245727539, "count": 1, "mi
n": 22.64261245727539, "max": 22.64261245727539}, "totaltime": {"sum": 2029.7093391418457, "count": 1, "min": 2029.70
93391418457, "max": 2029.7093391418457}}}

2021-12-14 17:04:42 Uploading - Uploading generated training model
2021-12-14 17:04:42 Completed - Training job completed
Training seconds: 59
Billable seconds: 59
```

Deploying the model

In [15]:
```python
# deploying the model to be used for inferences
rcf_predictor = rcf.deploy(
    model_name="{}-rcf".format("fraud-detection"),
    endpoint_name="{}-rcf".format("fraud-detection"),
    initial_instance_count=1,
    instance_type="ml.c5.xlarge")
```

```
Defaulting to the only supported framework/algorithm version: 1. Ignoring framework/algorithm version: 1.
```

```
-----!
```

In [16]:
```python
from sagemaker.predictor import CSVSerializer,JSONDeserializer

rcf_predictor.serializer = CSVSerializer()
rcf_predictor.deserializer = JSONDeserializer()
```

Testing the model

In [17]:
```python
# predicting the model , data splitted into small batches
def predict_rcf(current_predictor, data, rows=500):
    split_array = np.array_split(data, int(data.shape[0] / float(rows) + 1))
    predictions = []
    for array in split_array:
        array_preds = [s['score'] for s in current_predictor.predict(array)['scores']]
        predictions.append(array_preds)

    return np.concatenate([np.array(batch) for batch in predictions])
```

In [18]:
```python
import numpy as np

positives = X_test[y_test == 1]
positives_scores = predict_rcf(rcf_predictor, positives)

negatives = X_test[y_test == 0]
negatives_scores = predict_rcf(rcf_predictor, negatives)
```

In [19]:
```python
import seaborn as sns
import matplotlib.pyplot as plt
sns.set(color_codes=True)
```

In [20]:
```python
# plotting the positives and negatives as predicted by the model
sns.distplot(positives_scores, label='fraud', bins=20)
sns.distplot(negatives_scores, label='not-fraud', bins=20)
plt.legend()
```

Out[20]:
```
<matplotlib.legend.Legend at 0x7f223c0642d0>
```


In [ ]: