

Train a model with Amazon SageMaker Autopilot

Introduction

In this lab, you will use Amazon Sagemaker Autopilot to train a BERT-based natural language processing (NLP) model. The model will analyze customer feedback and classify the messages into positive (1), neutral (0) and negative (-1) sentiment.

Table of Contents

- [1. Review transformed dataset](#)
- [2. Configure the Autopilot job](#)
 - [2.1. Upload data to S3 bucket](#)
 - [2.2. S3 output for generated assets](#)
 - [2.3. Configure the Autopilot job](#)
 - [Exercise 1](#)
- [3. Launch the Autopilot job](#)
 - [Exercise 2](#)
- [4. Track Autopilot job progress](#)
 - [4.1. Autopilot job description](#)
 - [4.2. Autopilot job status](#)
 - [4.3. Review the SageMaker processing jobs](#)
 - [4.4. Wait for the data analysis step to finish](#)
 - [4.5. View generated notebooks](#)
 - [Exercise 3](#)
 - [Exercise 4](#)
- [5. Feature engineering](#)
 - [Exercise 5](#)
- [6. Model training and tuning](#)
 - [6.1. Wait for training and tuning](#)
 - [Exercise 6](#)
 - [6.2. Compare model candidates](#)
 - [Exercise 7](#)
 - [6.3. Review best candidate](#)
 - [Exercise 8](#)
- [7. Review all output in S3 bucket](#)
- [8. Deploy and test best candidate model](#)
 - [8.1. Deploy best candidate model](#)
 - [8.2. Test the model](#)

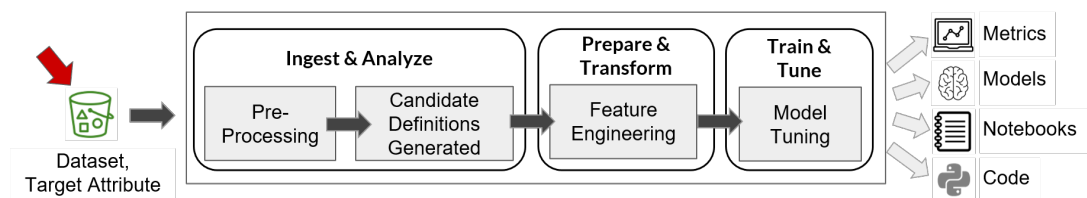
Amazon SageMaker Autopilot automatically trains and tunes the best machine

learning models for classification or regression, based on your data while allowing to maintain full control and visibility.

SageMaker Autopilot will inspect the raw dataset, apply feature processors, pick the best set of algorithms, train and tune multiple models, and then rank the models based on performance - all with just a few clicks. Autopilot transparently generates a set of Python scripts and notebooks for a complete end-to-end pipeline including data analysis, candidate generation, feature engineering, and model training/tuning.

SageMaker Autopilot job consists of the following high-level steps:

- *Data analysis* where the data is summarized and analyzed to determine which feature engineering techniques, hyper-parameters, and models to explore.
- *Feature engineering* where the data is scrubbed, balanced, combined, and split into train and validation.
- *Model training and tuning* where the top performing features, hyper-parameters, and models are selected and trained.



These re-usable scripts and notebooks give us full visibility into how the model candidates were created. Since Autopilot integrates natively with SageMaker Studio, we can visually explore the different models generated by SageMaker Autopilot.

SageMaker Autopilot can be used by people without machine learning experience to automatically train a model from a dataset. Additionally, experienced developers can use Autopilot to train a baseline model from which they can iterate and manually improve.

Autopilot is available through the SageMaker Studio UI and AWS Python SDK. In this notebook, you will use the AWS Python SDK to train a series of text-classification models and deploy the model with the highest accuracy.

For more details on Autopilot, have a look at this [Amazon Science Publication](#).

Use case: analyze customer sentiment

Customer feedback appears across many channels including social media and partner websites. As a company, you want to capture this valuable product feedback to spot negative trends and improve the situation, if needed. Here you will train a model to classify the feedback messages into positive (1), neutral (0) and negative (-1) sentiment.

First, let's install and import required modules.

```
In [2]: # please ignore warning messages during the installation
!pip install --disable-pip-version-check -q sagemaker==2.35.0
```

WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: <https://pip.pypa.io/warnings/venv>

```
In [3]: import boto3
import sagemaker
import pandas as pd
import numpy as np
import botocore
import time
import json

config = botocore.config.Config(user_agent_extra='dlai-pds/c1/w3')

# low-level service client of the boto3 session
sm = boto3.client(service_name='sagemaker',
                  config=config)

sm_runtime = boto3.client('sagemaker-runtime',
                          config=config)

sess = sagemaker.Session(sagemaker_client=sm,
                        sagemaker_runtime_client=sm_runtime)

bucket = sess.default_bucket()
role = sagemaker.get_execution_role()
region = sess.boto_region_name
```

```
In [4]: import matplotlib.pyplot as plt
%matplotlib inline
%config InlineBackend.figure_format='retina'
```

1. Review transformed dataset

Let's transform the dataset into a format that Autopilot recognizes. Specifically, a comma-separated file of `label, features` as shown here:

```
sentiment,review_body
-1,"this is bad"
0,"this is ok"
1,"this is great"
...
```

Sentiment is one of three classes: negative (-1), neutral (0), or positive (1). Autopilot requires that the target variable, `sentiment` is first and the set of features, just `review_body` in this case, come next.

```
In [5]: !aws s3 cp 's3://dlai-practical-data-science/data/balanced/womens_clothin
```

```
download: s3://dlai-practical-data-science/data/balanced/womens_clothing_
ecommerce_reviews_balanced.csv to ./womens_clothing_ecommerce_reviews_bal
anced.csv
```

```
In [6]: path = './womens_clothing_ecommerce_reviews_balanced.csv'

df = pd.read_csv(path, delimiter=',')
df.head()
```

```
Out[6]:
```

	sentiment	review_body	product_category
0	-1	This suit did nothing for me. the top has zero...	Swim
1	-1	Like other reviewers i saw this dress on the ...	Dresses
2	-1	I wish i had read the reviews before purchasin...	Knits
3	-1	I ordered these pants in my usual size (xl) an...	Legwear
4	-1	I noticed this top on one of the sales associa...	Knits

```
In [7]: path_autopilot = './womens_clothing_ecommerce_reviews_balanced_for_autopi

df[['sentiment', 'review_body']].to_csv(path_autopilot,
                                         sep=',',
                                         index=False)
```

2. Configure the Autopilot job

2.1. Upload data to S3 bucket

```
In [8]: autopilot_train_s3_uri = sess.upload_data(bucket=bucket, key_prefix='auto
autopilot_train_s3_uri
```

```
Out[8]: 's3://sagemaker-us-east-1-917875584351/autopilot/data/womens_clothing_eco
mmerce_reviews_balanced_for_autopilot.csv'
```

Check the existence of the dataset in this S3 bucket folder:

```
In [9]: !aws s3 ls $autopilot_train_s3_uri

2022-08-23 13:49:40      2253749 womens_clothing_ecommerce_reviews_balanced
_for_autopilot.csv
```

2.2. S3 output for generated assets

Set the S3 output path for the Autopilot outputs. This includes Jupyter notebooks (analysis), Python scripts (feature engineering), and trained models.

```
In [10]: model_output_s3_uri = 's3://{}/autopilot'.format(bucket)

print(model_output_s3_uri)
```

2.3. Configure the Autopilot job

Create the Autopilot job name.

```
In [11]: import time

timestamp = int(time.time())

auto_ml_job_name = 'automl-dm-{}'.format(timestamp)
```

When configuring our Autopilot job, you need to specify the maximum number of candidates, `max_candidates`, to explore as well as the input/output S3 locations and target column to predict. In this case, you want to predict `sentiment` from the review text.

Exercise 1

Configure the Autopilot job.

Instructions: Create an instance of the `sagemaker.automl.automl.AutoML` estimator class passing the required configuration parameters. Target attribute for predictions here is `sentiment`.

```
automl = sagemaker.automl.automl.AutoML(
    target_attribute_name='...', # the name of the target
    attribute for predictions
    base_job_name=..., # Autopilot job name
    output_path=..., # output data path
    max_candidates=..., # maximum number of candidates
    sagemaker_session=sess,
    role=role,
    max_runtime_per_training_job_in_seconds=1200,
    total_job_runtime_in_seconds=7200
)
```

```
In [14]: max_candidates = 3

automl = sagemaker.automl.automl.AutoML(
    ### BEGIN SOLUTION - DO NOT delete this comment for grading purposes
    target_attribute_name='sentiment', # Replace None
    base_job_name=auto_ml_job_name, # Replace None
    output_path=model_output_s3_uri, # Replace None
    ### END SOLUTION - DO NOT delete this comment for grading purposes
    max_candidates=max_candidates,
    sagemaker_session=sess,
    role=role,
    max_runtime_per_training_job_in_seconds=1200,
    total_job_runtime_in_seconds=7200
)
```

3. Launch the Autopilot job

Exercise 2

Launch the Autopilot job.

Instructions: Call `fit` function of the configured estimator passing the S3 bucket input data path and the Autopilot job name.

```
automl.fit(  
    ..., # input data path  
    job_name=auto_ml_job_name, # Autopilot job name  
    wait=False,  
    logs=False  
)
```

```
In [15]: automl.fit(  
    ### BEGIN SOLUTION - DO NOT delete this comment for grading purposes  
    autopilot_train_s3_uri, # Replace None  
    ### END SOLUTION - DO NOT delete this comment for grading purposes  
    job_name=auto_ml_job_name,  
    wait=False,  
    logs=False  
)
```

4. Track Autopilot job progress

Once the Autopilot job has been launched, you can track the job progress directly from the notebook using the SDK capabilities.

4.1. Autopilot job description

Function `describe_auto_ml_job` of the Amazon SageMaker service returns the information about the AutoML job in dictionary format. You can review the response syntax and response elements in the [documentation](#).

```
In [16]: job_description_response = automl.describe_auto_ml_job(job_name=auto_ml_j
```

4.2. Autopilot job status

To track the job progress you can use two response elements: `AutoMLJobStatus` and `AutoMLJobSecondaryStatus`, which correspond to the primary (Completed | InProgress | Failed | Stopped | Stopping) and secondary (AnalyzingData | FeatureEngineering | ModelTuning etc.) job states respectively. To see if the AutoML job has started, you can check the existence of the `AutoMLJobStatus` and `AutoMLJobSecondaryStatus` elements in the job description response.

In this notebook, you will use the following scheme to track the job progress:

```
# check if the job is still at certain stage
while [check 'AutoMLJobStatus' and 'AutoMLJobSecondaryStatus']
in job_description_response:
    # update the job description response
    job_description_response =
automl.describe_auto_ml_job(AutoMLJobName=auto_ml_job_name)
    # print the message the Autopilot job is in the stage ...
    print([message])
    # git a time step to check the status again
    sleep(15)
print("Autopilot job complete...")
```

```
In [17]: while 'AutoMLJobStatus' not in job_description_response.keys() and 'AutoM
job_description_response = automl.describe_auto_ml_job(job_name=auto_
print('[INFO] Autopilot job has not yet started. Please wait. ')
# function `json.dumps` encodes JSON string for printing.
print(json.dumps(job_description_response, indent=4, sort_keys=True,
print('[INFO] Waiting for Autopilot job to start...')
sleep(15)

print('[OK] AutoML job started.')
```

[OK] AutoML job started.

4.3. Review the SageMaker processing jobs

The Autopilot creates required SageMaker processing jobs during the run:

- First processing job (data splitter) checks the data sanity, performs stratified shuffling and splits the data into training and validation.
- Second processing job (candidate generator) first streams through the data to compute statistics for the dataset. Then, uses these statistics to identify the problem type, and possible types of every column-predictor: numeric, categorical, natural language, etc.

```
In [18]: from IPython.core.display import display, HTML

display(HTML('<b>Review <a target="blank" href="https://console.aws.amazo
```

Review processing jobs

You can review the updates on that page during the run of the Autopilot job.

4.4. Wait for the data analysis step to finish

Here you will use the same scheme as above to check the completion of the data analysis step. This step can be identified with the (primary) job status value

InProgress and secondary job status values Starting and then AnalyzingData .

This cell will take approximately 10 minutes to run.

[illegible]

InProgress AnalyzingData
InProgress FeatureEngineering
[OK] Data analysis phase completed.

```
{
  "AutoMLJobArn": "arn:aws:sagemaker:us-east-1:917875584351:automl-job/
automl-dm-1661262662",
  "AutoMLJobArtifacts": {
    "CandidateDefinitionNotebookLocation": "s3://sagemaker-us-east-1-
917875584351/autopilot/automl-dm-1661262662/sagemaker-automl-candidates/a
utoml-dm-1661262662-pr-1-61719c565070432bb34d947dd84accb25aafb/notebooks/
SageMakerAutopilotCandidateDefinitionNotebook.ipynb",
    "DataExplorationNotebookLocation": "s3://sagemaker-us-east-1-9178
75584351/autopilot/automl-dm-1661262662/sagemaker-automl-candidates/autom
l-dm-1661262662-pr-1-61719c565070432bb34d947dd84accb25aafb/notebooks/Sage
MakerAutopilotDataExplorationNotebook.ipynb"
  },
  "AutoMLJobConfig": {
    "CompletionCriteria": {
      "MaxAutoMLJobRuntimeInSeconds": 7200,
      "MaxCandidates": 3,
      "MaxRuntimePerTrainingJobInSeconds": 1200
    },
    "SecurityConfig": {
      "EnableInterContainerTrafficEncryption": false
    }
  },
  "AutoMLJobName": "automl-dm-1661262662",
  "AutoMLJobSecondaryStatus": "FeatureEngineering",
  "AutoMLJobStatus": "InProgress",
  "CreationTime": "2022-08-23 13:58:13.486000+00:00",
  "GenerateCandidateDefinitionsOnly": false,
  "InputDataConfig": [
    {
      "ChannelType": "training",
      "ContentType": "text/csv;header=present",
      "DataSource": {
        "S3DataSource": {
          "S3DataType": "S3Prefix",
          "S3Uri": "s3://sagemaker-us-east-1-917875584351/autop
ilot/data/womens_clothing_ecommerce_reviews_balanced_for_autopilot.csv"
        }
      },
      "TargetAttributeName": "sentiment"
    }
  ],
  "LastModifiedTime": "2022-08-23 14:07:30.547000+00:00",
  "OutputDataConfig": {
    "S3OutputPath": "s3://sagemaker-us-east-1-917875584351/autopilot"
  },
  "ResolvedAttributes": {
    "AutoMLJobObjective": {
      "MetricName": "Accuracy"
    },
    "CompletionCriteria": {
      "MaxAutoMLJobRuntimeInSeconds": 7200,
      "MaxCandidates": 3,
      "MaxRuntimePerTrainingJobInSeconds": 1200
    },
    "ProblemType": "MulticlassClassification"
  },
}
```

```

"ResponseMetadata": {
  "HTTPHeaders": {
    "content-length": "1811",
    "content-type": "application/x-amz-json-1.1",
    "date": "Tue, 23 Aug 2022 14:07:35 GMT",
    "x-amzn-requestid": "9c31db68-c5dc-4fcf-84e1-3e59debbdd9da"
  },
  "HTTPStatusCode": 200,
  "RequestId": "9c31db68-c5dc-4fcf-84e1-3e59debbdd9da",
  "RetryAttempts": 0
},
"RoleArn": "arn:aws:iam::917875584351:role/sagemaker-studio-vpc-firew
all-us-east-1-sagemaker-execution-role"
}
CPU times: user 357 ms, sys: 44.9 ms, total: 402 ms
Wall time: 7min 20s

```

Wait for Autopilot to finish generating the notebooks.

4.5. View generated notebooks

Once data analysis is complete, SageMaker AutoPilot generates two notebooks:

- Data exploration
- Candidate definition

Notebooks are included in the AutoML job artifacts generated during the run. Before checking the existence of the notebooks, you can check if the artifacts have been generated.

Exercise 3

Check if the Autopilot job artifacts have been generated.

Instructions: Use status check scheme described above. The generation of artifacts can be identified by existence of `AutoMLJobArtifacts` element in the keys of the job description response.

```

In [21]: ### BEGIN SOLUTION - DO NOT delete this comment for grading purposes
# get the information about the running Autopilot job
job_description_response = automl.describe_auto_ml_job(job_name=auto_ml_j

# keep in the while loop until the Autopilot job artifacts will be genera
while 'AutoMLJobArtifacts' not in job_description_response.keys(): # Repl
    # update the information about the running Autopilot job
    job_description_response = automl.describe_auto_ml_job(job_name=auto_
    ### END SOLUTION - DO NOT delete this comment for grading purposes
    print('[INFO] Autopilot job has not yet generated the artifacts. Plea
    print(json.dumps(job_description_response, indent=4, sort_keys=True,
    print('[INFO] Waiting for AutoMLJobArtifacts...')
    time.sleep(15)

print('[OK] AutoMLJobArtifacts generated.')

```

[OK] AutoMLJobArtifacts generated.

Wait for Autopilot to make the notebooks available.

Exercise 4

Check if the notebooks have been created.

Instructions: Use status check scheme described above. Notebooks creation can be identified by existence of `DataExplorationNotebookLocation` element in the keys of the `job_description_response['AutoMLJobArtifacts']` dictionary.

```
In [22]: ### BEGIN SOLUTION - DO NOT delete this comment for grading purposes
# get the information about the running Autopilot job
job_description_response = automl.describe_auto_ml_job(job_name=auto_ml_j

# keep in the while loop until the notebooks will be created
while 'DataExplorationNotebookLocation' not in job_description_response['
    # update the information about the running Autopilot job
    job_description_response = automl.describe_auto_ml_job(job_name=auto_
    ### END SOLUTION - DO NOT delete this comment for grading purposes
    print(['INFO] Autopilot job has not yet generated the notebooks. Plea
    print(json.dumps(job_description_response, indent=4, sort_keys=True,
    print(['INFO] Waiting for DataExplorationNotebookLocation...')
    time.sleep(15)

print(['OK] DataExplorationNotebookLocation found.')
```

[OK] DataExplorationNotebookLocation found.

Review the generated resources in S3 directly. Following the link, you can find the notebooks in the folder `notebooks` and download them by clicking on object `Actions / Object actions` -> `Download as / Download`.

```
In [23]: from IPython.core.display import display, HTML

generated_resources = job_description_response['AutoMLJobArtifacts']['Dat
download_path = generated_resources.rsplitt('/notebooks/SageMakerAutopilot
job_id = download_path.rsplitt('/', 1)[-1]

if not job_id:
    print('No AutoMLJobArtifacts found.')
else:
    display(HTML('<b>Review <a target="blank" href="https://s3.console.aw
```

Review [generated notebooks](#) in S3 bucket

5. Feature engineering

Exercise 5

Check the completion of the feature engineering step.

Instructions: Use status check scheme described above. Feature engineering step can be identified with the (primary) job status value `InProgress` and secondary job status value `FeatureEngineering`.

This cell will take approximately 10 minutes to run.

```
In [24]: %%time

job_description_response = automl.describe_auto_ml_job(job_name=auto_ml_j
job_status = job_description_response['AutoMLJobStatus']
job_sec_status = job_description_response['AutoMLJobSecondaryStatus']
print(job_status)
print(job_sec_status)
if job_status not in ('Stopped', 'Failed'):
    ### BEGIN SOLUTION - DO NOT delete this comment for grading purposes
    while job_status == None and job_sec_status == None: # Replace all No
    ### END SOLUTION - DO NOT delete this comment for grading purposes
        job_description_response = automl.describe_auto_ml_job(job_name=a
        job_status = job_description_response['AutoMLJobStatus']
        job_sec_status = job_description_response['AutoMLJobSecondaryStat
        print(job_status, job_sec_status)
        time.sleep(5)
    print('[OK] Feature engineering phase completed.\n')

print(json.dumps(job_description_response, indent=4, sort_keys=True, defa

InProgress
GeneratingModelInsightsReport
[OK] Feature engineering phase completed.

{
  "AutoMLJobArn": "arn:aws:sagemaker:us-east-1:917875584351:automl-job/
automl-dm-1661262662",
  "AutoMLJobArtifacts": {
    "CandidateDefinitionNotebookLocation": "s3://sagemaker-us-east-1-
917875584351/autopilot/automl-dm-1661262662/sagemaker-automl-candidates/a
utoml-dm-1661262662-pr-1-61719c565070432bb34d947dd84accb25aafb/notebooks/
SageMakerAutopilotCandidateDefinitionNotebook.ipynb",
    "DataExplorationNotebookLocation": "s3://sagemaker-us-east-1-9178
75584351/autopilot/automl-dm-1661262662/sagemaker-automl-candidates/autom
l-dm-1661262662-pr-1-61719c565070432bb34d947dd84accb25aafb/notebooks/Sage
MakerAutopilotDataExplorationNotebook.ipynb"
  },
  "AutoMLJobConfig": {
    "CompletionCriteria": {
      "MaxAutoMLJobRuntimeInSeconds": 7200,
      "MaxCandidates": 3,
      "MaxRuntimePerTrainingJobInSeconds": 1200
    },
    "SecurityConfig": {
      "EnableInterContainerTrafficEncryption": false
    }
  }
}
```

```

},
"AutoMLJobName": "automl-dm-1661262662",
"AutoMLJobSecondaryStatus": "GeneratingModelInsightsReport",
"AutoMLJobStatus": "InProgress",
"BestCandidate": {
  "CandidateName": "automl-dm-1661262662dOwiVeHvxc2g-002-105c69d9",
  "CandidateProperties": {
    "CandidateArtifactLocations": {
      "Explainability": "s3://sagemaker-us-east-1-917875584351/
autopilot/automl-dm-1661262662/documentation/explainability/output"
    },
    "CandidateMetrics": [
      {
        "MetricName": "Accuracy",
        "Set": "Validation",
        "StandardMetricName": "Accuracy",
        "Value": 0.6198300123214722
      },
      {
        "MetricName": "PrecisionMacro",
        "Set": "Validation",
        "StandardMetricName": "PrecisionMacro",
        "Value": 0.6221399903297424
      },
      {
        "MetricName": "BalancedAccuracy",
        "Set": "Validation",
        "StandardMetricName": "BalancedAccuracy",
        "Value": 0.6198300123214722
      },
      {
        "MetricName": "Flmacro",
        "Set": "Validation",
        "StandardMetricName": "Flmacro",
        "Value": 0.6208199858665466
      },
      {
        "MetricName": "RecallMacro",
        "Set": "Validation",
        "StandardMetricName": "RecallMacro",
        "Value": 0.6198300123214722
      }
    ]
  },
  "CandidateStatus": "Completed",
  "CandidateSteps": [
    {
      "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:processing-job/automl-dm-1661262662-db-1-53de988d6c0b4adb93322efa4a1
6ef99db41a",
      "CandidateStepName": "automl-dm-1661262662-db-1-53de988d6
c0b4adb93322efa4a16ef99db41a",
      "CandidateStepType": "AWS::SageMaker::ProcessingJob"
    },
    {
      "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:training-job/automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7
be09e58d",
      "CandidateStepName": "automl-dm-1661262662-dpp2-1-1080620
5885144c29a1074a63f7be09e58d",
      "CandidateStepType": "AWS::SageMaker::TrainingJob"
    }
  ]
}

```

```

    },
    {
        "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:transform-job/automl-dm-1661262662-dpp2-rpb-1-0efdb52b7b104271a6cdc7
231535111",
        "CandidateStepName": "automl-dm-1661262662-dpp2-rpb-1-0ef
db52b7b104271a6cdc7231535111",
        "CandidateStepType": "AWS::SageMaker::TransformJob"
    },
    {
        "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:training-job/automl-dm-1661262662dowivehvxc2g-002-105c69d9",
        "CandidateStepName": "automl-dm-1661262662dOwiVeHvxc2g-00
2-105c69d9",
        "CandidateStepType": "AWS::SageMaker::TrainingJob"
    }
],
"CreationTime": "2022-08-23 14:18:46+00:00",
"EndTime": "2022-08-23 14:20:54+00:00",
"FinalAutoMLJobObjectiveMetric": {
    "MetricName": "validation:accuracy",
    "Value": 0.6198300123214722
},
"InferenceContainers": [
    {
        "Environment": {
            "AUTOML_SPARSE_ENCODE_RECORDIO_PROTOBUF": "1",
            "AUTOML_TRANSFORM_MODE": "feature-transform",
            "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/
x-recordio-protobuf",
            "SAGEMAKER_PROGRAM": "sagemaker_serve",
            "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
        },
        "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sa
gemaker-sklearn-automl:2.5-1-cpu-py3",
        "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/au
topilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-d
pp2-1-10806205885144c29a1074a63f7be09e58d/output/model.tar.gz"
    },
    {
        "Environment": {
            "MAX_CONTENT_LENGTH": "20971520",
            "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
            "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
            "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,pro
bability,probabilities"
        },
        "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sa
gemaker-xgboost:1.3-1-cpu-py3",
        "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/au
topilot/automl-dm-1661262662/tuning/automl-dm--dpp2-xgb/automl-dm-1661262
662dOwiVeHvxc2g-002-105c69d9/output/model.tar.gz"
    },
    {
        "Environment": {
            "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
            "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
            "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
            "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
            "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,pro
bability,labels,probabilities",

```

```

        "SAGEMAKER_PROGRAM": "sagemaker_serve",
        "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7be09e58d/output/model.tar.gz"
}

],
"LastModifiedTime": "2022-08-23 14:31:44.722000+00:00",
"ObjectiveStatus": "Succeeded"
},
"CreationTime": "2022-08-23 13:58:13.486000+00:00",
"GenerateCandidateDefinitionsOnly": false,
"InputDataConfig": [
    {
        "ChannelType": "training",
        "ContentType": "text/csv;header=present",
        "DataSource": {
            "S3DataSource": {
                "S3DataType": "S3Prefix",
                "S3Uri": "s3://sagemaker-us-east-1-917875584351/autopilot/data/womens_clothing_ecommerce_reviews_balanced_for_autopilot.csv"
            }
        },
        "TargetAttributeName": "sentiment"
    }
],
"LastModifiedTime": "2022-08-23 14:43:56.320000+00:00",
"OutputDataConfig": {
    "S3OutputPath": "s3://sagemaker-us-east-1-917875584351/autopilot"
},
"ResolvedAttributes": {
    "AutoMLJobObjective": {
        "MetricName": "Accuracy"
    },
    "CompletionCriteria": {
        "MaxAutoMLJobRuntimeInSeconds": 7200,
        "MaxCandidates": 3,
        "MaxRuntimePerTrainingJobInSeconds": 1200
    },
    "ProblemType": "MulticlassClassification"
},
"ResponseMetadata": {
    "HTTPHeaders": {
        "content-length": "5789",
        "content-type": "application/x-amz-json-1.1",
        "date": "Tue, 23 Aug 2022 14:44:03 GMT",
        "x-amzn-requestid": "0f09f5d0-fd5e-498d-8ff2-95d4f954044e"
    },
    "HTTPStatusCode": 200,
    "RequestId": "0f09f5d0-fd5e-498d-8ff2-95d4f954044e",
    "RetryAttempts": 0
},
"RoleArn": "arn:aws:iam::917875584351:role/sagemaker-studio-vpc-firewall-us-east-1-sagemaker-execution-role"
}
CPU times: user 14.5 ms, sys: 6.52 ms, total: 21 ms
Wall time: 230 ms

```

6. Model training and tuning

When you launched the Autopilot job, you requested that 3 model candidates are generated and compared. Therefore, you should see three (3) SageMaker training jobs below.

```
In [25]: from IPython.core.display import display, HTML

display(HTML('<b>Review <a target="blank" href="https://console.aws.amazo
```

Review [hyper-parameter tuning jobs](https://console.aws.amazo)

6.1. Wait for training and tuning

Exercise 6

Check the completion of the model tuning step.

Instructions: Use status check scheme described above. Model tuning step can be identified with the (primary) job status value `InProgress` and secondary job status value `ModelTuning`.

This cell will take approximately 5-10 minutes to run.

```
In [26]: %%time

job_description_response = automl.describe_auto_ml_job(job_name=auto_ml_j
job_status = job_description_response['AutoMLJobStatus']
job_sec_status = job_description_response['AutoMLJobSecondaryStatus']
print(job_status)
print(job_sec_status)
if job_status not in ('Stopped', 'Failed'):
    ### BEGIN SOLUTION - DO NOT delete this comment for grading purposes
    while job_status == None and job_sec_status == None: # Replace all No
    ### END SOLUTION - DO NOT delete this comment for grading purposes
        job_description_response = automl.describe_auto_ml_job(job_name=a
        job_status = job_description_response['AutoMLJobStatus']
        job_sec_status = job_description_response['AutoMLJobSecondaryStat
        print(job_status, job_sec_status)
        time.sleep(5)
    print('[OK] Model tuning phase completed.\n')

print(json.dumps(job_description_response, indent=4, sort_keys=True, defa

InProgress
GeneratingModelInsightsReport
[OK] Model tuning phase completed.

{
    "AutoMLJobArn": "arn:aws:sagemaker:us-east-1:917875584351:auto_ml-job/
```



```

automl-dm-1661262662",
  "AutoMLJobArtifacts": {
    "CandidateDefinitionNotebookLocation": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/sagemaker-automl-candidates/automl-dm-1661262662-pr-1-61719c565070432bb34d947dd84accb25aafb/notebooks/SageMakerAutopilotCandidateDefinitionNotebook.ipynb",
    "DataExplorationNotebookLocation": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/sagemaker-automl-candidates/automl-dm-1661262662-pr-1-61719c565070432bb34d947dd84accb25aafb/notebooks/SageMakerAutopilotDataExplorationNotebook.ipynb"
  },
  "AutoMLJobConfig": {
    "CompletionCriteria": {
      "MaxAutoMLJobRuntimeInSeconds": 7200,
      "MaxCandidates": 3,
      "MaxRuntimePerTrainingJobInSeconds": 1200
    },
    "SecurityConfig": {
      "EnableInterContainerTrafficEncryption": false
    }
  },
  "AutoMLJobName": "automl-dm-1661262662",
  "AutoMLJobSecondaryStatus": "GeneratingModelInsightsReport",
  "AutoMLJobStatus": "InProgress",
  "BestCandidate": {
    "CandidateName": "automl-dm-1661262662dOwiVeHvxc2g-002-105c69d9",
    "CandidateProperties": {
      "CandidateArtifactLocations": {
        "Explainability": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/documentation/explainability/output"
      }
    },
    "CandidateMetrics": [
      {
        "MetricName": "Accuracy",
        "Set": "Validation",
        "StandardMetricName": "Accuracy",
        "Value": 0.6198300123214722
      },
      {
        "MetricName": "PrecisionMacro",
        "Set": "Validation",
        "StandardMetricName": "PrecisionMacro",
        "Value": 0.6221399903297424
      },
      {
        "MetricName": "BalancedAccuracy",
        "Set": "Validation",
        "StandardMetricName": "BalancedAccuracy",
        "Value": 0.6198300123214722
      },
      {
        "MetricName": "F1macro",
        "Set": "Validation",
        "StandardMetricName": "F1macro",
        "Value": 0.6208199858665466
      },
      {
        "MetricName": "RecallMacro",
        "Set": "Validation",
        "StandardMetricName": "RecallMacro",
        "Value": 0.6198300123214722
      }
    ]
  }
}

```

```

    }
  ],
  "CandidateStatus": "Completed",
  "CandidateSteps": [
    {
      "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:processing-job/automl-dm-1661262662-db-1-53de988d6c0b4adb93322efa4a1
6ef99db41a",
      "CandidateStepName": "automl-dm-1661262662-db-1-53de988d6
c0b4adb93322efa4a16ef99db41a",
      "CandidateStepType": "AWS::SageMaker::ProcessingJob"
    },
    {
      "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:training-job/automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7
be09e58d",
      "CandidateStepName": "automl-dm-1661262662-dpp2-1-1080620
5885144c29a1074a63f7be09e58d",
      "CandidateStepType": "AWS::SageMaker::TrainingJob"
    },
    {
      "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:transform-job/automl-dm-1661262662-dpp2-rpb-1-0efdb52b7b104271a6cdc7
231535111",
      "CandidateStepName": "automl-dm-1661262662-dpp2-rpb-1-0ef
db52b7b104271a6cdc7231535111",
      "CandidateStepType": "AWS::SageMaker::TransformJob"
    },
    {
      "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:training-job/automl-dm-1661262662dowivehvx2g-002-105c69d9",
      "CandidateStepName": "automl-dm-1661262662dOwiVeHvxc2g-00
2-105c69d9",
      "CandidateStepType": "AWS::SageMaker::TrainingJob"
    }
  ],
  "CreationTime": "2022-08-23 14:18:46+00:00",
  "EndTime": "2022-08-23 14:20:54+00:00",
  "FinalAutoMLJobObjectiveMetric": {
    "MetricName": "validation:accuracy",
    "Value": 0.6198300123214722
  },
  "InferenceContainers": [
    {
      "Environment": {
        "AUTOML_SPARSE_ENCODE_RECORDIO_PROTOBUF": "1",
        "AUTOML_TRANSFORM_MODE": "feature-transform",
        "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/
x-recordio-protobuf",
        "SAGEMAKER_PROGRAM": "sagemaker_serve",
        "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
      },
      "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sa
gemaker-sklearn-automl:2.5-1-cpu-py3",
      "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/au
topilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-d
pp2-1-10806205885144c29a1074a63f7be09e58d/output/model.tar.gz"
    },
    {
      "Environment": {

```

```

        "MAX_CONTENT_LENGTH": "20971520",
        "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
        "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,probabilities"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/tuning/automl-dm--dpp2-xgb/automl-dm-1661262662dOwiVeHvxc2g-002-105c69d9/output/model.tar.gz"
},
{
    "Environment": {
        "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
        "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
        "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,labels,probabilities",
        "SAGEMAKER_PROGRAM": "sagemaker_serve",
        "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7be09e58d/output/model.tar.gz"
}
],
"LastModifiedTime": "2022-08-23 14:31:44.722000+00:00",
"ObjectiveStatus": "Succeeded"
},
"CreationTime": "2022-08-23 13:58:13.486000+00:00",
"GenerateCandidateDefinitionsOnly": false,
"InputDataConfig": [
    {
        "ChannelType": "training",
        "ContentType": "text/csv;header=present",
        "DataSource": {
            "S3DataSource": {
                "S3DataType": "S3Prefix",
                "S3Uri": "s3://sagemaker-us-east-1-917875584351/autopilot/data/womens_clothing_ecommerce_reviews_balanced_for_autopilot.csv"
            }
        },
        "TargetAttributeName": "sentiment"
    }
],
"LastModifiedTime": "2022-08-23 14:44:13.394000+00:00",
"OutputDataConfig": {
    "S3OutputPath": "s3://sagemaker-us-east-1-917875584351/autopilot"
},
"ResolvedAttributes": {
    "AutoMLJobObjective": {
        "MetricName": "Accuracy"
    }
},
"CompletionCriteria": {
    "MaxAutoMLJobRuntimeInSeconds": 7200,
    "MaxCandidates": 3,
    "MaxRuntimePerTrainingJobInSeconds": 1200
}

```

```

    },
    "ProblemType": "MulticlassClassification"
  },
  "ResponseMetadata": {
    "HTTPHeaders": {
      "content-length": "5790",
      "content-type": "application/x-amz-json-1.1",
      "date": "Tue, 23 Aug 2022 14:44:14 GMT",
      "x-amzn-requestid": "90953695-43a4-4a73-a07c-6f250039c0ca"
    },
    "HTTPStatusCode": 200,
    "RequestId": "90953695-43a4-4a73-a07c-6f250039c0ca",
    "RetryAttempts": 0
  },
  "RoleArn": "arn:aws:iam::917875584351:role/sagemaker-studio-vpc-firew
all-us-east-1-sagemaker-execution-role"
}
CPU times: user 18.6 ms, sys: 3.32 ms, total: 21.9 ms
Wall time: 224 ms

```

Please wait until ^^ Autopilot ^^ completes above

Finally, you can check the completion of the Autopilot job looking for the **Completed** job status.

```

In [27]: %%time

from pprint import pprint

job_description_response = automl.describe_auto_ml_job(job_name=auto_ml_j
pprint(job_description_response)
job_status = job_description_response['AutoMLJobStatus']
job_sec_status = job_description_response['AutoMLJobSecondaryStatus']
print('Job status: {}'.format(job_status))
print('Secondary job status: {}'.format(job_sec_status))
if job_status not in ('Stopped', 'Failed'):
    while job_status not in ('Completed'):
        job_description_response = automl.describe_auto_ml_job(job_name=a
        job_status = job_description_response['AutoMLJobStatus']
        job_sec_status = job_description_response['AutoMLJobSecondaryStat
        print('Job status: {}'.format(job_status))
        print('Secondary job status: {}'.format(job_sec_status))
        time.sleep(10)
    print('[OK] Autopilot job completed.\n')
else:
    print('Job status: {}'.format(job_status))
    print('Secondary job status: {}'.format(job_status))

{'AutoMLJobArn': 'arn:aws:sagemaker:us-east-1:917875584351:automl-job/aut
oml-dm-1661262662',
 'AutoMLJobArtifacts': {'CandidateDefinitionNotebookLocation': 's3://sage
maker-us-east-1-917875584351/autopilot/automl-dm-1661262662/sagemaker-aut
oml-candidates/automl-dm-1661262662-pr-1-61719c565070432bb34d947dd84accb2
5aafb/notebooks/SageMakerAutopilotCandidateDefinitionNotebook.ipynb',
                        'DataExplorationNotebookLocation': 's3://sagemake
r-us-east-1-917875584351/autopilot/automl-dm-1661262662/sagemaker-automl-
candidates/automl-dm-1661262662-pr-1-61719c565070432bb34d947dd84accb25aaf
b/notebooks/SageMakerAutopilotDataExplorationNotebook.ipynb'},
 'AutoMLJobConfig': {'CompletionCriteria': {'MaxAutoMLJobRuntimeInSeconds
': 7200,

```

```

        'MaxCandidates': 3,
        'MaxRuntimePerTrainingJobInSeconds': 1200},
        'SecurityConfig': {'EnableInterContainerTrafficEncryption': False}},
        'AutoMLJobName': 'automl-dm-1661262662',
        'AutoMLJobSecondaryStatus': 'GeneratingModelInsightsReport',
        'AutoMLJobStatus': 'InProgress',
        'BestCandidate': {'CandidateName': 'automl-dm-1661262662dOwiVeHvxc2g-002-105c69d9',
        'CandidateProperties': {'CandidateArtifactLocations':
{'Explainability': 's3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/documentation/explainability/output'},
        'CandidateMetrics': [{'MetricName': 'Accuracy',
                                'Set': 'Validation',
                                'StandardMetricName': 'Accuracy',
                                'Value': 0.6198300123214722},
                                {'MetricName': 'PrecisionMacro',
                                'Set': 'Validation',
                                'StandardMetricName': 'PrecisionMacro',
                                'Value': 0.6221399903297424},
                                {'MetricName': 'BalancedAccuracy',
                                'Set': 'Validation',
                                'StandardMetricName': 'BalancedAccuracy',
                                'Value': 0.6198300123214722},
                                {'MetricName': 'F1macro',
                                'Set': 'Validation',
                                'StandardMetricName': 'F1macro',
                                'Value': 0.6208199858665466},
                                {'MetricName': 'RecallMacro',
                                'Set': 'Validation',
                                'StandardMetricName': 'RecallMacro',
                                'Value': 0.6198300123214722}]}},
        'CandidateStatus': 'Completed',
        'CandidateSteps': [{'CandidateStepArn': 'arn:aws:sagemaker:us-east-1:917875584351:processing-job/automl-dm-1661262662-db-1-53de988d6c0b4adb93322efa4a16ef99db41a',
                                'CandidateStepName': 'automl-dm-1661262662-db-1-53de988d6c0b4adb93322efa4a16ef99db41a',
                                'CandidateStepType': 'AWS::SageMaker::ProcessingJob'}],

```

```

        {'CandidateStepArn': 'arn:aws:sagem
aker:us-east-1:917875584351:training-job/automl-dm-1661262662-dpp2-1-1080
6205885144c29a1074a63f7be09e58d',
        'CandidateStepName': 'automl-dm-16
61262662-dpp2-1-10806205885144c29a1074a63f7be09e58d',
        'CandidateStepType': 'AWS::SageMak
er::TrainingJob'},
        {'CandidateStepArn': 'arn:aws:sagem
aker:us-east-1:917875584351:transform-job/automl-dm-1661262662-dpp2-rpb-1
-0efdb52b7b104271a6cdc7231535111',
        'CandidateStepName': 'automl-dm-16
61262662-dpp2-rpb-1-0efdb52b7b104271a6cdc7231535111',
        'CandidateStepType': 'AWS::SageMak
er::TransformJob'},
        {'CandidateStepArn': 'arn:aws:sagem
aker:us-east-1:917875584351:training-job/automl-dm-1661262662dowivehvxc2g
-002-105c69d9',
        'CandidateStepName': 'automl-dm-16
61262662dOwiVeHvxc2g-002-105c69d9',
        'CandidateStepType': 'AWS::SageMak
er::TrainingJob'}]],
        'CreationTime': datetime.datetime(2022, 8, 23, 14, 18,
46, tzinfo=tzlocal()),
        'EndTime': datetime.datetime(2022, 8, 23, 14, 20, 54,
tzinfo=tzlocal()),
        'FinalAutoMLJobObjectiveMetric': {'MetricName': 'valid
ation:accuracy',
        'Value': 0.619830012
3214722},
        'InferenceContainers': [{'Environment': {'AUTOML_SPARS
E_ENCODE_RECORDIO_PROTOBUF': '1',
        'AUTOML_TRANS
FORM_MODE': 'feature-transform',
        'SAGEMAKER_DE
FAULT_INVOCATIONS_ACCEPT': 'application/x-recordio-protobuf',
        'SAGEMAKER_PR
OGRAM': 'sagemaker_serve',
        'SAGEMAKER_SU
BMIT_DIRECTORY': '/opt/ml/model/code'},
        'Image': '683313688378.dkr.ec
r.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3',
        'ModelDataUrl': 's3://sagemak
er-us-east-1-917875584351/autopilot/automl-dm-1661262662/data-processor-m
odels/automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7be09e58d/out
put/model.tar.gz'},
        {'Environment': {'MAX_CONTENT_
LENGTH': '20971520',
        'SAGEMAKER_DE
FAULT_INVOCATIONS_ACCEPT': 'text/csv',
        'SAGEMAKER_IN
FERENCE_OUTPUT': 'predicted_label',
        'SAGEMAKER_IN
FERENCE_SUPPORTED': 'predicted_label,probability,probabilities'},
        'Image': '683313688378.dkr.ec
r.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3',
        'ModelDataUrl': 's3://sagemak
er-us-east-1-917875584351/autopilot/automl-dm-1661262662/tuning/automl-dm
--dpp2-xgb/automl-dm-1661262662dOwiVeHvxc2g-002-105c69d9/output/model.tar
.gz'},
        {'Environment': {'AUTOML_TRANS
FORM_MODE': 'inverse-label-transform',

```

```

'Image': '683313688378.dkr.ec
r.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3',
'ModelDataUrl': 's3://sagemak
er-us-east-1-917875584351/autopilot/automl-dm-1661262662/data-processor-m
odels/automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7be09e58d/out
put/model.tar.gz']],
'LastModifiedTime': datetime.datetime(2022, 8, 23, 14,
31, 44, 722000, tzinfo=tzlocal()),
'ObjectiveStatus': 'Succeeded',
'CreationTime': datetime.datetime(2022, 8, 23, 13, 58, 13, 486000, tzinf
o=tzlocal()),
'GenerateCandidateDefinitionsOnly': False,
'InputDataConfig': [{'ChannelType': 'training',
'ContentType': 'text/csv;header=present',
'DataSource': {'S3DataSource': {'S3DataType': 'S3Pr
efix',
'S3Uri': 's3://sage
maker-us-east-1-917875584351/autopilot/data/womens_clothing_ecommerce_rev
iews_balanced_for_autopilot.csv'}}},
'TargetAttributeName': 'sentiment']],
'LastModifiedTime': datetime.datetime(2022, 8, 23, 14, 44, 13, 394000, t
zinfo=tzlocal()),
'OutputDataConfig': {'S3OutputPath': 's3://sagemaker-us-east-1-917875584
351/autopilot'},
'ResolvedAttributes': {'AutoMLJobObjective': {'MetricName': 'Accuracy'},
'CompletionCriteria': {'MaxAutoMLJobRuntimeInSeco
nds': 7200,
'MaxCandidates': 3,
'MaxRuntimePerTrainingJobI
nSeconds': 1200},
'ProblemType': 'MulticlassClassification'},
'ResponseMetadata': {'HTTPHeaders': {'content-length': '5790',
'content-type': 'application/x-amz-
json-1.1',
'date': 'Tue, 23 Aug 2022 14:44:17
GMT',
'x-amzn-requestid': '524ce9df-c673-
44ab-9c9c-869cb7aa8a51'}},
'HTTPStatusCode': 200,
'RequestId': '524ce9df-c673-44ab-9c9c-869cb7aa8a51'
,
'RetryAttempts': 0},
'RoleArn': 'arn:aws:iam::917875584351:role/sagemaker-studio-vpc-firewall
-us-east-1-sagemaker-execution-role'}
Job status: InProgress
Secondary job status: GeneratingModelInsightsReport
Job status: InProgress
Secondary job status: GeneratingModelInsightsReport
Job status: InProgress

```


6.2. Compare model candidates

Once model tuning is complete, you can view all the candidates (pipeline evaluations with different hyperparameter combinations) that were explored by AutoML and sort them by their final performance metric.

Exercise 7

List candidates generated by Autopilot sorted by accuracy from highest to lowest.

Instructions: Use `list_candidates` function passing the Autopilot job name `auto_ml_job_name` with the accuracy field `FinalObjectiveMetricValue`. It returns the list of candidates with the information about them.

```
candidates = automl.list_candidates(  
    job_name=..., # Autopilot job name  
    sort_by='...' # accuracy field name  
)
```

```
In [29]: candidates = automl.list_candidates(  
    ### BEGIN SOLUTION - DO NOT delete this comment for grading purposes  
    job_name=auto_ml_job_name, # Replace None  
    sort_by='FinalObjectiveMetricValue' # Replace None  
    ### END SOLUTION - DO NOT delete this comment for grading purposes  
)
```

You can review the response syntax and response elements of the function `list_candidates` in the [documentation](#). Now let's put the candidate existence check into the loop:

```
In [30]: while candidates == []:  
    candidates = automl.list_candidates(job_name=auto_ml_job_name)  
    print(['INFO] Autopilot job is generating the candidates. Please wait  
    time.sleep(10)  
  
    print(['OK] Candidates generated.')
```

[OK] Candidates generated.

The information about each of the candidates is in the dictionary with the following keys:

```
In [31]: print(candidates[0].keys())  
  
dict_keys(['CandidateName', 'FinalAutoMLJobObjectiveMetric', 'ObjectiveSt  
atus', 'CandidateSteps', 'CandidateStatus', 'InferenceContainers', 'Creat  
ionTime', 'EndTime', 'LastModifiedTime', 'CandidateProperties'])
```

`CandidateName` contains the candidate name and the `FinalAutoMLJobObjectiveMetric` element contains the metric information which can be used to identify the best candidate later. Let's check that they were generated.

```
In [32]: while 'CandidateName' not in candidates[0]:
          candidates = automl.list_candidates(job_name=auto_ml_job_name)
          print('[INFO] Autopilot job is generating CandidateName. Please wait.
                sleep(10)

          print('[OK] CandidateName generated.')
```

[OK] CandidateName generated.

```
In [33]: while 'FinalAutoMLJobObjectiveMetric' not in candidates[0]:
          candidates = automl.list_candidates(job_name=auto_ml_job_name)
          print('[INFO] Autopilot job is generating FinalAutoMLJobObjectiveMetric
                sleep(10)

          print('[OK] FinalAutoMLJobObjectiveMetric generated.')
```

[OK] FinalAutoMLJobObjectiveMetric generated.

```
In [34]: print(json.dumps(candidates, indent=4, sort_keys=True, default=str))
```

```
[
  {
    "CandidateName": "automl-dm-1661262662dOwiVeHvxc2g-002-105c69d9",
    "CandidateProperties": {
      "CandidateArtifactLocations": {
        "Explainability": "s3://sagemaker-us-east-1-917875584351/
autopilot/automl-dm-1661262662/documentation/explainability/output",
        "ModelInsights": "s3://sagemaker-us-east-1-917875584351/a
utopilot/automl-dm-1661262662/documentation/model_monitor/output"
      },
      "CandidateMetrics": [
        {
          "MetricName": "Accuracy",
          "Set": "Validation",
          "StandardMetricName": "Accuracy",
          "Value": 0.6198300123214722
        },
        {
          "MetricName": "PrecisionMacro",
          "Set": "Validation",
          "StandardMetricName": "PrecisionMacro",
          "Value": 0.6221399903297424
        },
        {
          "MetricName": "BalancedAccuracy",
          "Set": "Validation",
          "StandardMetricName": "BalancedAccuracy",
          "Value": 0.6198300123214722
        },
        {
          "MetricName": "F1macro",
          "Set": "Validation",
          "StandardMetricName": "F1macro",
          "Value": 0.6208199858665466
        }
      ]
    }
  ]
]
```

```

    },
    {
      "MetricName": "RecallMacro",
      "Set": "Validation",
      "StandardMetricName": "RecallMacro",
      "Value": 0.6198300123214722
    }
  ],
  "CandidateStatus": "Completed",
  "CandidateSteps": [
    {
      "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:processing-job/automl-dm-1661262662-db-1-53de988d6c0b4adb93322efa4a16ef99db41a",
      "CandidateStepName": "automl-dm-1661262662-db-1-53de988d6c0b4adb93322efa4a16ef99db41a",
      "CandidateStepType": "AWS::SageMaker::ProcessingJob"
    },
    {
      "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:training-job/automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7be09e58d",
      "CandidateStepName": "automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7be09e58d",
      "CandidateStepType": "AWS::SageMaker::TrainingJob"
    },
    {
      "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:transform-job/automl-dm-1661262662-dpp2-rpb-1-0efdb52b7b104271a6cdc7231535111",
      "CandidateStepName": "automl-dm-1661262662-dpp2-rpb-1-0efdb52b7b104271a6cdc7231535111",
      "CandidateStepType": "AWS::SageMaker::TransformJob"
    },
    {
      "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:training-job/automl-dm-1661262662dowivehvxc2g-002-105c69d9",
      "CandidateStepName": "automl-dm-1661262662dOwiVeHvxc2g-002-105c69d9",
      "CandidateStepType": "AWS::SageMaker::TrainingJob"
    }
  ],
  "CreationTime": "2022-08-23 14:18:46+00:00",
  "EndTime": "2022-08-23 14:20:54+00:00",
  "FinalAutoMLJobObjectiveMetric": {
    "MetricName": "validation:accuracy",
    "Value": 0.6198300123214722
  },
  "InferenceContainers": [
    {
      "Environment": {
        "AUTOML_SPARSE_ENCODE_RECORDIO_PROTOBUF": "1",
        "AUTOML_TRANSFORM_MODE": "feature-transform",
        "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/x-recordio-protobuf",
        "SAGEMAKER_PROGRAM": "sagemaker_serve",
        "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
      },
      "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
    }
  ]
}

```

```

        "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/au
topilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-d
pp2-1-10806205885144c29a1074a63f7be09e58d/output/model.tar.gz"
    },
    {
        "Environment": {
            "MAX_CONTENT_LENGTH": "20971520",
            "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
            "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
            "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,pro
bability,probabilities"
        },
        "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sa
gemaker-xgboost:1.3-1-cpu-py3",
        "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/au
topilot/automl-dm-1661262662/tuning/automl-dm--dpp2-xgb/automl-dm-1661262
662dOwiVeHvxc2g-002-105c69d9/output/model.tar.gz"
    },
    {
        "Environment": {
            "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
            "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
            "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
            "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
            "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,pro
bability,labels,probabilities",
            "SAGEMAKER_PROGRAM": "sagemaker_serve",
            "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
        },
        "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sa
gemaker-sklearn-automl:2.5-1-cpu-py3",
        "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/au
topilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-d
pp2-1-10806205885144c29a1074a63f7be09e58d/output/model.tar.gz"
    }
],
    "LastModifiedTime": "2022-08-23 14:31:44.722000+00:00",
    "ObjectiveStatus": "Succeeded"
},
{
    "CandidateName": "automl-dm-1661262662dOwiVeHvxc2g-001-b0a3f023",
    "CandidateProperties": {
        "CandidateMetrics": [
            {
                "MetricName": "Accuracy",
                "Set": "Validation",
                "StandardMetricName": "Accuracy",
                "Value": 0.4041700065135956
            },
            {
                "MetricName": "PrecisionMacro",
                "Set": "Validation",
                "StandardMetricName": "PrecisionMacro",
                "Value": 0.4039599895477295
            },
            {
                "MetricName": "BalancedAccuracy",
                "Set": "Validation",
                "StandardMetricName": "BalancedAccuracy",
                "Value": 0.4041700065135956
            }
        ]
    }
}

```

```

        {
            "MetricName": "Flmacro",
            "Set": "Validation",
            "StandardMetricName": "Flmacro",
            "Value": 0.4038900136947632
        },
        {
            "MetricName": "RecallMacro",
            "Set": "Validation",
            "StandardMetricName": "RecallMacro",
            "Value": 0.4041700065135956
        }
    ],
    "CandidateStatus": "Completed",
    "CandidateSteps": [
        {
            "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:processing-job/automl-dm-1661262662-db-1-53de988d6c0b4adb93322efa4a16ef99db41a",
            "CandidateStepName": "automl-dm-1661262662-db-1-53de988d6c0b4adb93322efa4a16ef99db41a",
            "CandidateStepType": "AWS::SageMaker::ProcessingJob"
        },
        {
            "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:training-job/automl-dm-1661262662-dpp1-1-6de12eef194d4ed69d1f49b982cf26ad8f7",
            "CandidateStepName": "automl-dm-1661262662-dpp1-1-6de12eef194d4ed69d1f49b982cf26ad8f7",
            "CandidateStepType": "AWS::SageMaker::TrainingJob"
        },
        {
            "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:transform-job/automl-dm-1661262662-dpp1-csv-1-309d5a78731a47688872a631d7a6174",
            "CandidateStepName": "automl-dm-1661262662-dpp1-csv-1-309d5a78731a47688872a631d7a6174",
            "CandidateStepType": "AWS::SageMaker::TransformJob"
        },
        {
            "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:training-job/automl-dm-1661262662dowivehvxc2g-001-b0a3f023",
            "CandidateStepName": "automl-dm-1661262662dowivehvxc2g-001-b0a3f023",
            "CandidateStepType": "AWS::SageMaker::TrainingJob"
        }
    ],
    "CreationTime": "2022-08-23 14:18:48+00:00",
    "EndTime": "2022-08-23 14:21:56+00:00",
    "FinalAutoMLJobObjectiveMetric": {
        "MetricName": "validation:accuracy",
        "Value": 0.4041700065135956
    },
    "InferenceContainers": [
        {
            "Environment": {
                "AUTOML_TRANSFORM_MODE": "feature-transform",
                "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/x-recordio-protobuf",
                "SAGEMAKER_PROGRAM": "sagemaker_serve",

```

```

        "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-dppl-1-6de12eef194d4ed69d1f49b982cf26ad8f7/output/model.tar.gz"
},
{
    "Environment": {
        "MAX_CONTENT_LENGTH": "20971520",
        "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
        "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,probabilities"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/tuning/automl-dm--dppl-xgb/automl-dm-1661262662dOwiVeHvxc2g-001-b0a3f023/output/model.tar.gz"
},
{
    "Environment": {
        "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
        "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
        "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,labels,probabilities",
        "SAGEMAKER_PROGRAM": "sagemaker_serve",
        "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-dppl-1-6de12eef194d4ed69d1f49b982cf26ad8f7/output/model.tar.gz"
}
],
"LastModifiedTime": "2022-08-23 14:31:44.650000+00:00",
"ObjectiveStatus": "Succeeded"
},
{
    "CandidateName": "automl-dm-1661262662dOwiVeHvxc2g-003-ca78fb67",
    "CandidateProperties": {
        "CandidateMetrics": [
            {
                "MetricName": "Accuracy",
                "Set": "Validation",
                "StandardMetricName": "Accuracy",
                "Value": 0.40327998995780945
            },
            {
                "MetricName": "PrecisionMacro",
                "Set": "Validation",
                "StandardMetricName": "PrecisionMacro",
                "Value": 0.40283000469207764
            },
            {
                "MetricName": "BalancedAccuracy",

```

```

        "Set": "Validation",
        "StandardMetricName": "BalancedAccuracy",
        "Value": 0.40327998995780945
    },
    {
        "MetricName": "Flmacro",
        "Set": "Validation",
        "StandardMetricName": "Flmacro",
        "Value": 0.40290001034736633
    },
    {
        "MetricName": "RecallMacro",
        "Set": "Validation",
        "StandardMetricName": "RecallMacro",
        "Value": 0.40327998995780945
    }
]
},
"CandidateStatus": "Completed",
"CandidateSteps": [
    {
        "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:processing-job/automl-dm-1661262662-db-1-53de988d6c0b4adb93322efa4a1
6ef99db41a",
        "CandidateStepName": "automl-dm-1661262662-db-1-53de988d6
c0b4adb93322efa4a16ef99db41a",
        "CandidateStepType": "AWS::SageMaker::ProcessingJob"
    },
    {
        "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:training-job/automl-dm-1661262662-dpp1-1-6de12eef194d4ed69d1f49b982c
f26ad8f7",
        "CandidateStepName": "automl-dm-1661262662-dpp1-1-6de12ee
f194d4ed69d1f49b982cf26ad8f7",
        "CandidateStepType": "AWS::SageMaker::TrainingJob"
    },
    {
        "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:transform-job/automl-dm-1661262662-dpp1-csv-1-309d5a78731a47688872a6
31d7a6174",
        "CandidateStepName": "automl-dm-1661262662-dpp1-csv-1-309
d5a78731a47688872a631d7a6174",
        "CandidateStepType": "AWS::SageMaker::TransformJob"
    },
    {
        "CandidateStepArn": "arn:aws:sagemaker:us-east-1:91787558
4351:training-job/automl-dm-1661262662dowivehvxc2g-003-ca78fb67",
        "CandidateStepName": "automl-dm-1661262662dOwiVeHvxc2g-00
3-ca78fb67",
        "CandidateStepType": "AWS::SageMaker::TrainingJob"
    }
],
"CreationTime": "2022-08-23 14:18:52+00:00",
"EndTime": "2022-08-23 14:30:47+00:00",
"FinalAutoMLJobObjectiveMetric": {
    "MetricName": "validation:accuracy",
    "Value": 0.40327998995780945
},
"InferenceContainers": [
    {
        "Environment": {

```

```

        "AUTOML_TRANSFORM_MODE": "feature-transform",
        "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/
x-recordio-protobuf",
        "SAGEMAKER_PROGRAM": "sagemaker_serve",
        "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sa
gemaker-sklearn-automl:2.5-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/au
topilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-d
ppl-1-6de12eef194d4ed69d1f49b982cf26ad8f7/output/model.tar.gz"
},
{
    "Environment": {
        "MAX_CONTENT_LENGTH": "20971520",
        "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
        "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,pro
bability,probabilities"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sa
gemaker-xgboost:1.3-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/au
topilot/automl-dm-1661262662/tuning/automl-dm--dppl-xgb/automl-dm-1661262
662d0wiVeHvxc2g-003-ca78fb67/output/model.tar.gz"
},
{
    "Environment": {
        "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
        "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
        "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,pro
bability,labels,probabilities",
        "SAGEMAKER_PROGRAM": "sagemaker_serve",
        "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sa
gemaker-sklearn-automl:2.5-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/au
topilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-d
ppl-1-6de12eef194d4ed69d1f49b982cf26ad8f7/output/model.tar.gz"
    }
},
    "LastModifiedTime": "2022-08-23 14:31:44.650000+00:00",
    "ObjectiveStatus": "Succeeded"
}
]

```

You can print the names of the candidates with their metric values:

```

In [35]: print("metric " + str(candidates[0]['FinalAutoMLJobObjectiveMetric']['Met

for index, candidate in enumerate(candidates):
    print(str(index) + " "
          + candidate['CandidateName'] + " "
          + str(candidate['FinalAutoMLJobObjectiveMetric']['Value']))

```



```
metric validation:accuracy
0  automl-dm-1661262662d0wiVeHvxc2g-002-105c69d9  0.6198300123214722
1  automl-dm-1661262662d0wiVeHvxc2g-001-b0a3f023  0.4041700065135956
2  automl-dm-1661262662d0wiVeHvxc2g-003-ca78fb67  0.40327998995780945
```

6.3. Review best candidate

Now that you have successfully completed the Autopilot job on the dataset and visualized the trials, you can get the information about the best candidate model and review it.

Exercise 8

Get the information about the generated best candidate job.

Instructions: Use `best_candidate` function passing the Autopilot job name. This function will give an error if candidates have not been generated.

```
In [36]: candidates = automl.list_candidates(job_name=auto_ml_job_name)

if candidates != []:
    best_candidate = automl.best_candidate(
        ### BEGIN SOLUTION - DO NOT delete this comment for grading purpose
        job_name=auto_ml_job_name # Replace None
        ### END SOLUTION - DO NOT delete this comment for grading purpose
    )
    print(json.dumps(best_candidate, indent=4, sort_keys=True, default=str))

{
  "CandidateName": "automl-dm-1661262662d0wiVeHvxc2g-002-105c69d9",
  "CandidateProperties": {
    "CandidateArtifactLocations": {
      "Explainability": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/documentation/explainability/output",
      "ModelInsights": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/documentation/model_monitor/output"
    },
    "CandidateMetrics": [
      {
        "MetricName": "Accuracy",
        "Set": "Validation",
        "StandardMetricName": "Accuracy",
        "Value": 0.6198300123214722
      },
      {
        "MetricName": "PrecisionMacro",
        "Set": "Validation",
        "StandardMetricName": "PrecisionMacro",
        "Value": 0.6221399903297424
      },
      {
        "MetricName": "BalancedAccuracy",
        "Set": "Validation",
        "StandardMetricName": "BalancedAccuracy",
        "Value": 0.6198300123214722
      }
    ]
  }
}
```

```

    },
    {
      "MetricName": "F1macro",
      "Set": "Validation",
      "StandardMetricName": "F1macro",
      "Value": 0.6208199858665466
    },
    {
      "MetricName": "RecallMacro",
      "Set": "Validation",
      "StandardMetricName": "RecallMacro",
      "Value": 0.6198300123214722
    }
  ]
},
"CandidateStatus": "Completed",
"CandidateSteps": [
  {
    "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:processing-job/automl-dm-1661262662-db-1-53de988d6c0b4adb93322efa4a16ef99db41a",
    "CandidateStepName": "automl-dm-1661262662-db-1-53de988d6c0b4adb93322efa4a16ef99db41a",
    "CandidateStepType": "AWS::SageMaker::ProcessingJob"
  },
  {
    "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:training-job/automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7be09e58d",
    "CandidateStepName": "automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7be09e58d",
    "CandidateStepType": "AWS::SageMaker::TrainingJob"
  },
  {
    "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:transform-job/automl-dm-1661262662-dpp2-rpb-1-0efdb52b7b104271a6cdc7231535111",
    "CandidateStepName": "automl-dm-1661262662-dpp2-rpb-1-0efdb52b7b104271a6cdc7231535111",
    "CandidateStepType": "AWS::SageMaker::TransformJob"
  },
  {
    "CandidateStepArn": "arn:aws:sagemaker:us-east-1:917875584351:training-job/automl-dm-1661262662dowivehvxc2g-002-105c69d9",
    "CandidateStepName": "automl-dm-1661262662dowiveHvxc2g-002-105c69d9",
    "CandidateStepType": "AWS::SageMaker::TrainingJob"
  }
],
"CreationTime": "2022-08-23 14:18:46+00:00",
"EndTime": "2022-08-23 14:20:54+00:00",
"FinalAutoMLJobObjectiveMetric": {
  "MetricName": "validation:accuracy",
  "Value": 0.6198300123214722
},
"InferenceContainers": [
  {
    "Environment": {
      "AUTOML_SPARSE_ENCODE_RECORDIO_PROTOBUF": "1",
      "AUTOML_TRANSFORM_MODE": "feature-transform",
      "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "application/x-re

```

```

cordio-protobuf",
        "SAGEMAKER_PROGRAM": "sagemaker_serve",
        "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7be09e58d/output/model.tar.gz"
},
{
    "Environment": {
        "MAX_CONTENT_LENGTH": "20971520",
        "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
        "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,probabilities"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-xgboost:1.3-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/tuning/automl-dm--dpp2-xgb/automl-dm-1661262662dOwiVeHvxc2g-002-105c69d9/output/model.tar.gz"
},
{
    "Environment": {
        "AUTOML_TRANSFORM_MODE": "inverse-label-transform",
        "SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT": "text/csv",
        "SAGEMAKER_INFERENCE_INPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_OUTPUT": "predicted_label",
        "SAGEMAKER_INFERENCE_SUPPORTED": "predicted_label,probability,labels,probabilities",
        "SAGEMAKER_PROGRAM": "sagemaker_serve",
        "SAGEMAKER_SUBMIT_DIRECTORY": "/opt/ml/model/code"
    },
    "Image": "683313688378.dkr.ecr.us-east-1.amazonaws.com/sagemaker-sklearn-automl:2.5-1-cpu-py3",
    "ModelDataUrl": "s3://sagemaker-us-east-1-917875584351/autopilot/automl-dm-1661262662/data-processor-models/automl-dm-1661262662-dpp2-1-10806205885144c29a1074a63f7be09e58d/output/model.tar.gz"
}
},
    "LastModifiedTime": "2022-08-23 14:31:44.722000+00:00",
    "ObjectiveStatus": "Succeeded"
}

```

Check the existence of the candidate name for the best candidate.

```

In [37]: while 'CandidateName' not in best_candidate:
    best_candidate = automl.best_candidate(job_name=auto_ml_job_name)
    print('[INFO] Autopilot Job is generating BestCandidate CandidateName')
    print(json.dumps(best_candidate, indent=4, sort_keys=True, default=str))
    sleep(10)

print('[OK] BestCandidate CandidateName generated.')

```

[OK] BestCandidate CandidateName generated.

Check the existence of the metric value for the best candidate.

```
In [38]: while 'FinalAutoMLJobObjectiveMetric' not in best_candidate:
    best_candidate = automl.best_candidate(job_name=auto_ml_job_name)
    print('[INFO] Autopilot Job is generating BestCandidate FinalAutoMLJobObjectiveMetric')
    print(json.dumps(best_candidate, indent=4, sort_keys=True, default=str))
    sleep(10)

print('[OK] BestCandidate FinalAutoMLJobObjectiveMetric generated.')
```

[OK] BestCandidate FinalAutoMLJobObjectiveMetric generated.

Print the information about the best candidate:

```
In [39]: best_candidate_identifier = best_candidate['CandidateName']
print("Candidate name: " + best_candidate_identifier)
print("Metric name: " + best_candidate['FinalAutoMLJobObjectiveMetric']['MetricName'])
print("Metric value: " + str(best_candidate['FinalAutoMLJobObjectiveMetric']['MetricValue']))
```

Candidate name: automl-dm-1661262662dOwiVeHvxc2g-002-105c69d9
Metric name: validation:accuracy
Metric value: 0.6198300123214722

7. Review all output in S3 bucket

You will see the artifacts generated by Autopilot including the following:

data-processor-models/	# "models" learned to transform raw data into features
documentation/	# explainability and other documentation about your model
preprocessed-data/	# data for train and validation
sagemaker-automl-candidates/	# candidate models which autopilot compares
transformed-data/	# candidate-specific data for train and validation
tuning/	# candidate-specific tuning results
validations/	# validation results

```
In [40]: from IPython.core.display import display, HTML

display(
    HTML(
        '<b>Review all <a target="blank" href="https://s3.console.aws.amazon.com/s3/buckets/?region={region}&prefix={prefix}&show-versions=true">output</a> in S3 bucket, region, auto_ml_job_name'
    )
)
```

Review all [output in S3](#)

8. Deploy and test best candidate model

8.1. Deploy best candidate model

While batch transformations are supported, you will deploy our model as a REST Endpoint in this example.

First, you need to customize the inference response. The inference containers generated by SageMaker Autopilot allow you to select the response content for predictions. By default the inference containers are configured to generate the `predicted_label`. But you can add `probability` into the list of inference response keys.

```
In [41]: inference_response_keys = ['predicted_label', 'probability']
```

Now you will create a SageMaker endpoint from the best candidate generated by Autopilot. Wait for SageMaker to deploy the endpoint.

This cell will take approximately 5-10 minutes to run.

```
In [43]: autopilot_model = automl.deploy(  
    initial_instance_count=1,  
    instance_type='ml.m5.large',  
    candidate=best_candidate,  
    inference_response_keys=inference_response_keys,  
    predictor_cls=sagemaker.predictor.Predictor,  
    serializer=sagemaker.serializers.JSONSerializer(),  
    deserializer=sagemaker.deserializers.JSONDeserializer()  
)  
  
print('\nEndpoint name: {}'.format(autopilot_model.endpoint_name))
```

```
-----!  
Endpoint name: sagemaker-sklearn-automl-2022-08-23-15-03-12-767
```

Please wait until the ^^ endpoint ^^ is deployed.

Review the SageMaker endpoint in the AWS console.

```
In [44]: from IPython.core.display import display, HTML  
  
display(HTML('<b>Review <a target="blank" href="https://console.aws.amazo
```

Review [SageMaker REST endpoint](https://console.aws.amazon.com/sagemaker/home?region=us-east-1#/endpoints/sagemaker-sklearn-automl-2022-08-23-15-03-12-767)

8.2. Test the model

Invoke a few predictions for the actual reviews using the deployed endpoint.

```
In [45]: #sm_runtime = boto3.client('sagemaker-runtime')

review_list = ['This product is great!',
               'OK, but not great.',
               'This is not the right product.']

for review in review_list:

    # remove commas from the review since we're passing the inputs as a CSV
    review = review.replace(",", "")

    response = sm_runtime.invoke_endpoint(
        EndpointName=autopilot_model.endpoint_name, # endpoint name
        ContentType='text/csv', # type of input data
        Accept='text/csv', # type of the inference in the response
        Body=review # review text
    )

    response_body=response['Body'].read().decode('utf-8').strip().split(',')

    print('Review: ', review, ' Predicated class: {}'.format(response_body[0]))

print("(-1 = Negative, 0=Neutral, 1=Positive)")

Review: This product is great! Predicated class: 1
Review: OK but not great. Predicated class: 0
Review: This is not the right product. Predicated class: -1
(-1 = Negative, 0=Neutral, 1=Positive)
```

You used Amazon SageMaker Autopilot to automatically find the best model, hyper-parameters, and feature-engineering scripts for our dataset. Autopilot uses a uniquely-transparent approach to AutoML by generating re-usable Python scripts and notebooks.

Upload the notebook into S3 bucket for grading purposes.

Note: you may need to click on "Save" button before the upload.

```
In [ ]: !aws s3 cp ./C1_W3_Assignment.ipynb s3://$bucket/C1_W3_Assignment_Learner
```

Please go to the main lab window and click on **Submit** button (see the **Finish the lab** section of the instructions).

```
In [ ]:
```