ASSIGNMENT-2

Q1. What is AWS ML?

- Amazon Machine Learning is a service that allows to develop predictive applications by using algorithms, mathematical models based on the user's data.
- Amazon Machine Learning reads data through Amazon S3, Redshift and RDS, then visualizes the data through the AWS Management Console and the Amazon Machine Learning API. This data can be imported or exported to other AWS services via S3 buckets.
- It uses "industry-standard logistic regression" algorithm to generate models.

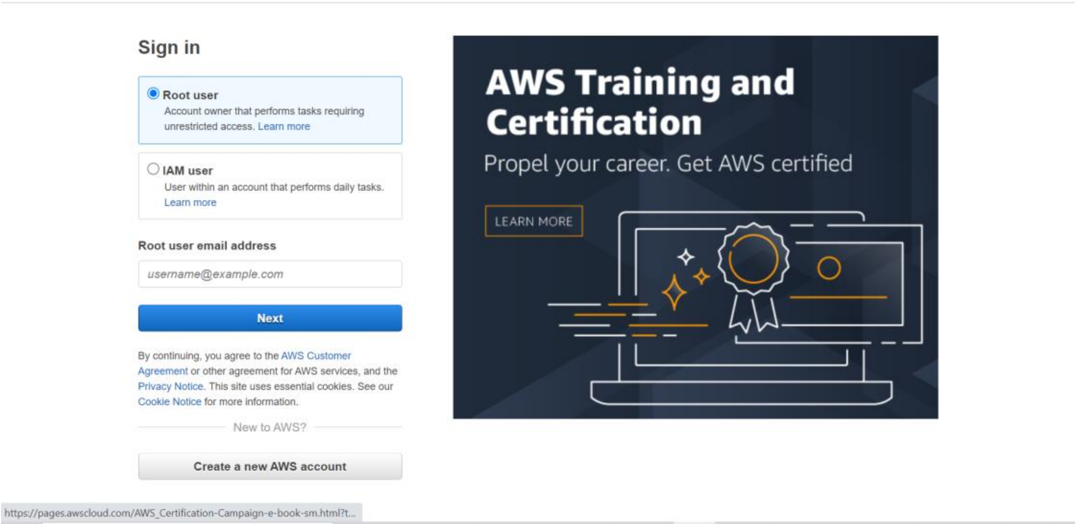
Three different types of tasks can be performed by Amazon Machine learning service –

- A binary classification model can predict one of the two possible results, i.e., either yes or no.
- A multi-class classification model can predict multiple conditions. For example, it can track a customer's online orders.
- A regression model results in an exact value. Regression models can predict the best-selling price for a product or the number of units that will sell.

Q2. Create your AWS ML account. Write steps for AWS ML.

AWS ML is now a deprecated service. Now AWS Sagemaker is used instead of AWS ML.

Step-1: Visit the AWS homepage using the following link: https://aws.amazon.com/
Step-2: Click on the "Sign-in to console" option in the top right corner of the screen.
Step-3: Select "Create a new AWS account" option as shown below.



Sign in

● Root user
Account owner that performs tasks requiring unrestricted access. Learn more

○ IAM user
User within an account that performs daily tasks. Learn more

Root user email address

username@example.com

By continuing, you agree to the AWS Customer Agreement or other agreement for AWS services, and the Privacy Notice. This site uses essential cookies. See our Cookie Notice for more information.

https://pages.awscloud.com/AWS_Certification-Campaign-e-book-sm.html?t...

Next
New to AWS?
Create a new AWS account
AWS Training and Certification
Propel your career. Get AWS certified
LEARN MORE

Step-4: Enter the required details (email, username, credit-card details etc), choose the Free-Tier plan and then verify your email address.
Step-5: You may now sign-in to your account.
Step-6: Search for AWS Sagemaker service in the search bar and then select the option.
Step-7: Setup the Sagemaker domain. For Execution role, choose Create an IAM role. In the dialog box that appears, choose Any S3 bucket and choose Create role.
Step-8: Now ML related services can be used from Sagemaker.

Q3. List 5 real time Use case for AWS ML with problem statement.

IDENTIFY FRAUDULENT ONLINE ACTIVITIES
Globally each year, tens of billions of dollars are lost to online fraud. Traditionally, companies used rule-based fraud detection applications that aren't accurate enough and can't keep up with the changing behaviours of fraudsters. With AWS Fraud Detection machine learning solutions, companies can proactively and more

accurately detect and prevent online fraud. These solutions will help reduce revenue losses, avoid brand damage, and provide a frictionless customer online experience while adapting to changing threat patterns.
Use cases:
1) Payment and transaction fraud detection
2) New account fraud
3) Account takeover
4) Authentication

INCREASE CUSTOMER SATISFACTION WITH CONVERSATIONAL AI INTERFACES

Conversational AI interfaces add human-like conversation capabilities to the business applications by combining different natural language technologies like natural language processing (NLP), natural language understanding (NLU) and natural language generation (NLG). Conversational interfaces continue to grow as one of the preferred ways for users to interact with businesses. Covid-19 has further accelerated the adoption of these interfaces given social distancing rules and shelter in place orders. Enterprises are developing conversational interfaces to engage with users in new ways like interactive chatbots or virtual assistants capable of understanding customer needs, gathering required user information, and integrating back-end services to complete the required task.

AUTOMATE DATA PROCESSING FROM DOCUMENTS

Documents come in various file types, varied formats, and contain valuable information. In most cases, you are manually processing the documents which is time consuming, prone to error, and costly. Not only do you want this information quickly but likely need to use the information within those documents for downstream applications.

To help overcome these challenges, AWS Machine Learning (ML) now provides you choices when it comes to extracting information from complex content in any document format such as insurance claims, mortgages, healthcare claims, contracts, and legal contracts.

FIND ACCURATE INFORMATION FASTER

Nearly half the time, knowledge workers fail to find the information they need to excel because information is scattered across their organization in the form of unstructured data.

Not only is this data hard to find, but your employees are often using search tools that miss the mark because they return long lists of documents that employees have to sift through. This extends the time it takes to find what they are looking for, and that’s if they even find it at all.

AWS offers Amazon Kendra, an intelligent search service powered by machine learning, to address this issue. Kendra uses natural language search capabilities to help your organization quickly return accurate answers from unstructured content.

PERSONALIZE YOUR CUSTOMER EXPERIENCES

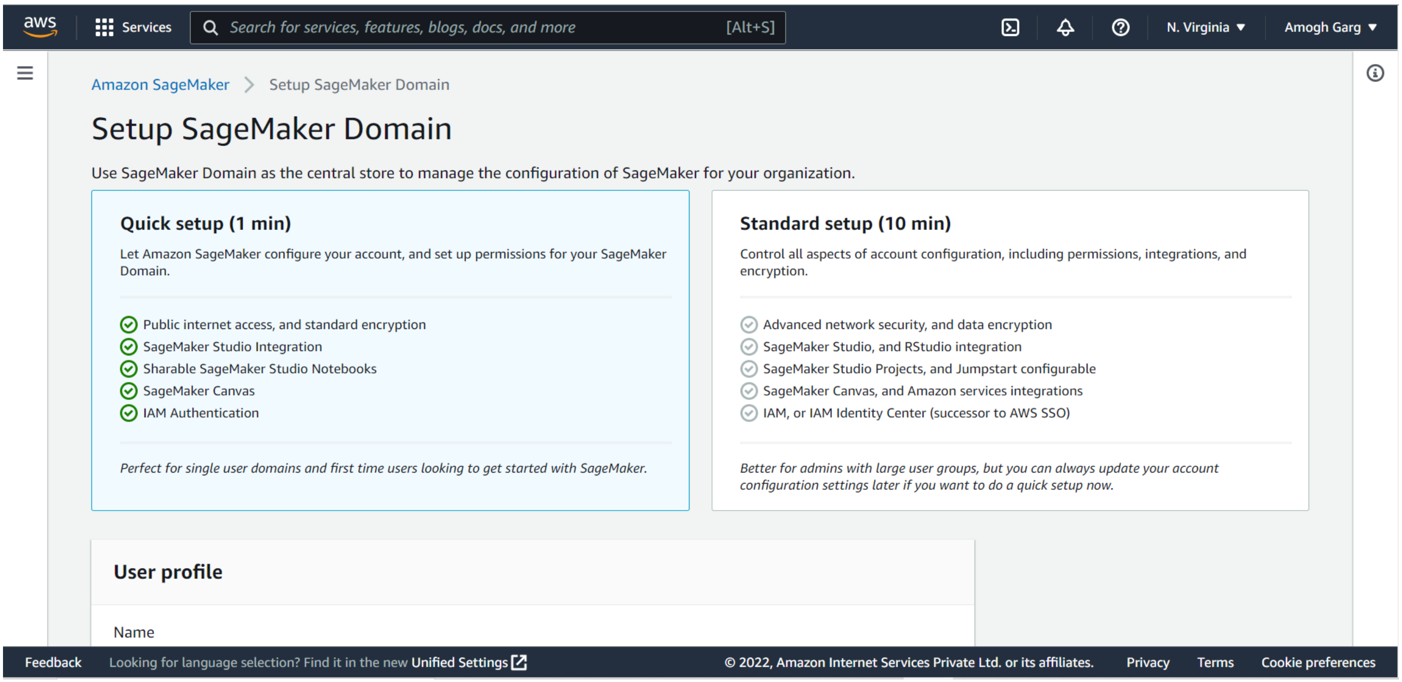
As the ability to deliver more sophisticated digital experiences has evolved over time, so has the expectation and demand from customers to receive a more personalized experience from the brands they engage with across retail, media and entertainment, travel and hospitality and more. Consumers today expect real-time, curated experiences across digital channels as they consider, purchase, and use products and services.

Machine learning (ML) can help organizations deliver highly personalized experiences, resulting in improvements in customer engagement, conversion, revenue, and margin and create differentiation in a digital world.

AWS offers machine learning solutions that deliver higher-quality personalized experiences for your customers across digital channels, all tailored to your business needs.

Q4. Implement one real time Use Case in AWS ML.

Problem Statement: You assume the role of a developer working at a bank. You have been asked to develop a machine learning model to predict whether a customer will enrol for a certificate of deposit (CD). The model will be trained on the marketing dataset that contains information on customer demographics, responses to marketing events, and external factors.


aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia ▼
Amogh Garg ▼

Amazon SageMaker > Setup SageMaker Domain

Setup SageMaker Domain

Use SageMaker Domain as the central store to manage the configuration of SageMaker for your organization.

Quick setup (1 min)

Let Amazon SageMaker configure your account, and set up permissions for your SageMaker Domain.

- ✓ Public internet access, and standard encryption
- ✓ SageMaker Studio Integration
- ✓ Sharable SageMaker Studio Notebooks
- ✓ SageMaker Canvas
- ✓ IAM Authentication

Perfect for single user domains and first time users looking to get started with SageMaker.

Standard setup (10 min)

Control all aspects of account configuration, including permissions, integrations, and encryption.

- ✓ Advanced network security, and data encryption
- ✓ SageMaker Studio, and RStudio integration
- ✓ SageMaker Studio Projects, and Jumpstart configurable
- ✓ SageMaker Canvas, and Amazon services integrations
- ✓ IAM, or IAM Identity Center (successor to AWS SSO)

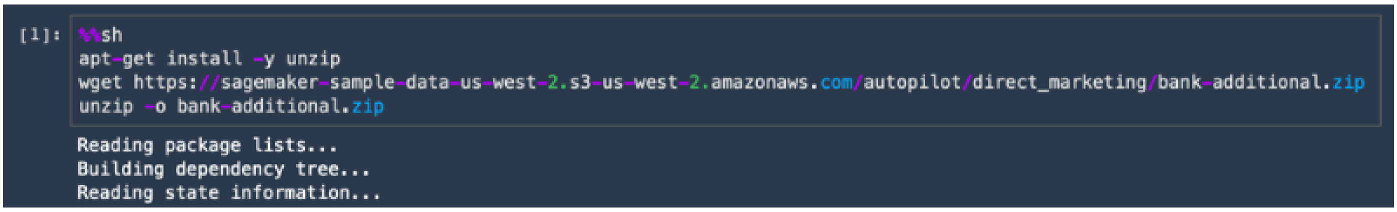
Better for admins with large user groups, but you can always update your account configuration settings later if you want to do a quick setup now.

User profile

Name

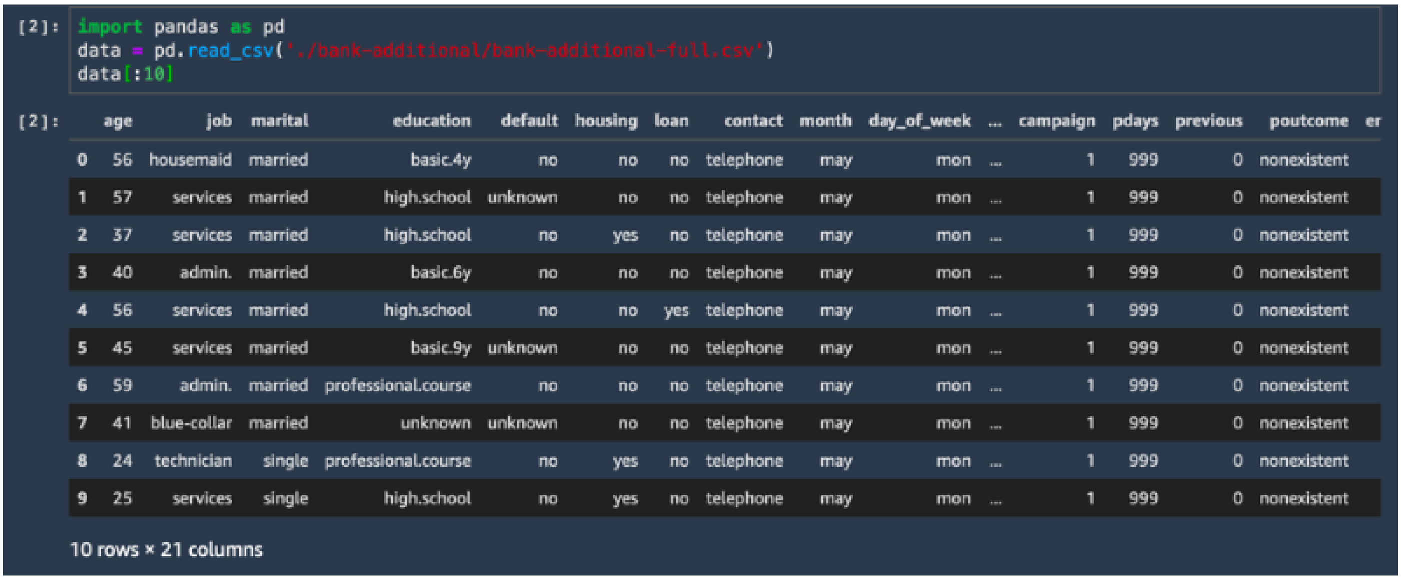
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences

Step-1: Download the dataset.



```
[1]: %%sh
     apt-get install -y unzip
     wget https://sagemaker-sample-data-us-west-2.s3-us-west-2.amazonaws.com/autopilot/direct_marketing/bank-additional.zip
     unzip -o bank-additional.zip

     Reading package lists...
     Building dependency tree...
     Reading state information...
```

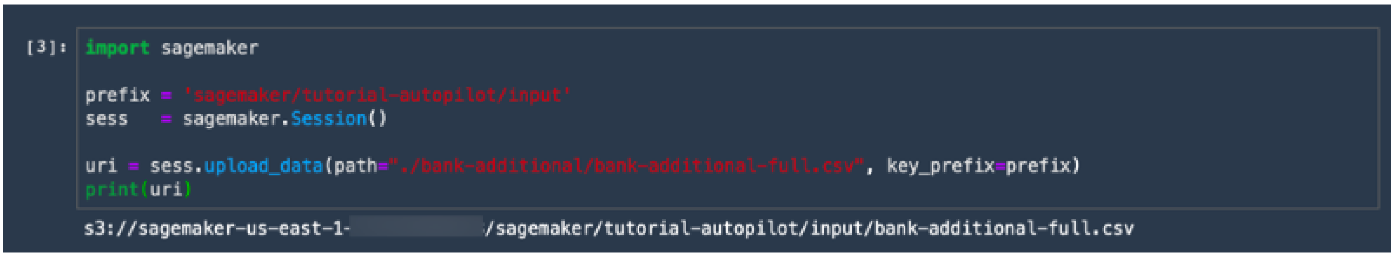


```
[2]: import pandas as pd
     data = pd.read_csv('./bank-additional/bank-additional-full.csv')
     data[:10]
```

[2]:

	age	job	marital	education	default	housing	loan	contact	month	day_of_week	...	campaign	pdays	previous	poutcome	er
0	56	housemaid	married	basic.4y	no	no	no	telephone	may	mon	...	1	999	0	nonexistent	
1	57	services	married	high.school	unknown	no	no	telephone	may	mon	...	1	999	0	nonexistent	
2	37	services	married	high.school	no	yes	no	telephone	may	mon	...	1	999	0	nonexistent	
3	40	admin.	married	basic.6y	no	no	no	telephone	may	mon	...	1	999	0	nonexistent	
4	56	services	married	high.school	no	no	yes	telephone	may	mon	...	1	999	0	nonexistent	
5	45	services	married	basic.9y	unknown	no	no	telephone	may	mon	...	1	999	0	nonexistent	
6	59	admin.	married	professional.course	no	no	no	telephone	may	mon	...	1	999	0	nonexistent	
7	41	blue-collar	married	unknown	unknown	no	no	telephone	may	mon	...	1	999	0	nonexistent	
8	24	technician	single	professional.course	no	yes	no	telephone	may	mon	...	1	999	0	nonexistent	
9	25	services	single	high.school	no	yes	no	telephone	may	mon	...	1	999	0	nonexistent	

10 rows × 21 columns



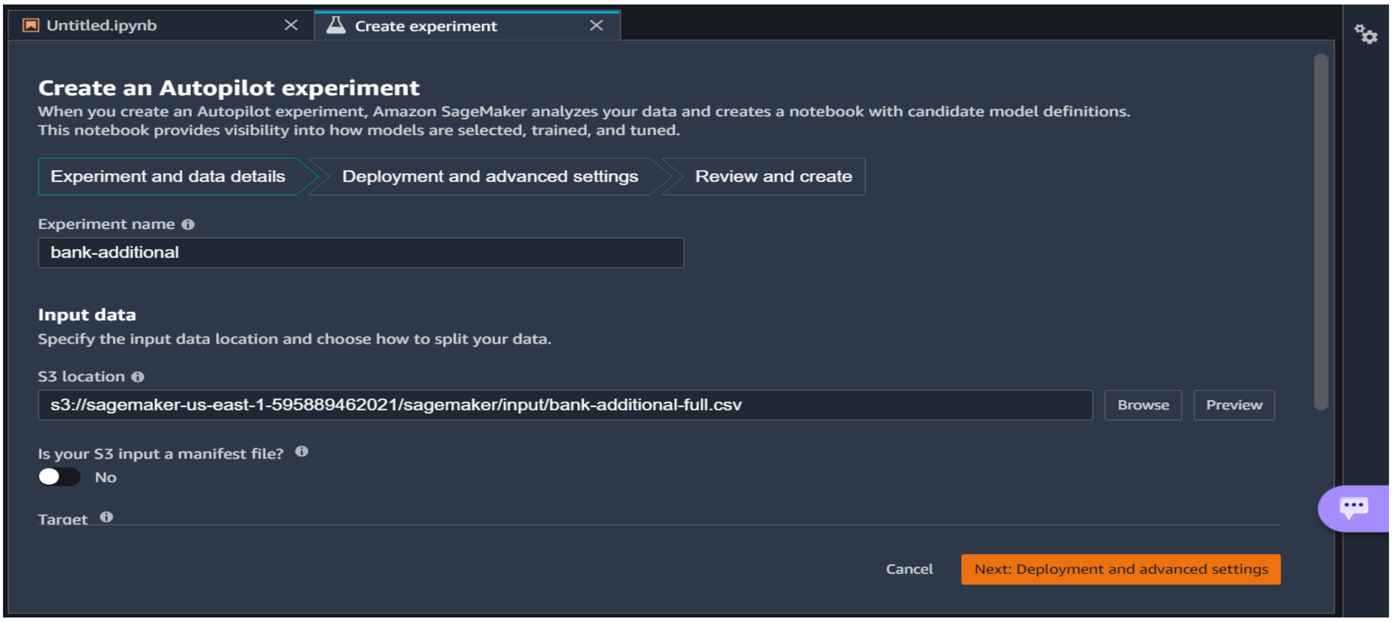
```
[3]: import sagemaker

     prefix = 'sagemaker/tutorial-autopilot/input'
     sess   = sagemaker.Session()

     uri = sess.upload_data(path="./bank-additional/bank-additional-full.csv", key_prefix=prefix)
     print(uri)

     s3://sagemaker-us-east-1-[redacted]/sagemaker/tutorial-autopilot/input/bank-additional-full.csv
```

Step-2: Create a Sagemaker Autopilot experiment.



Create an Autopilot experiment

When you create an Autopilot experiment, Amazon SageMaker analyzes your data and creates a notebook with candidate model definitions. This notebook provides visibility into how models are selected, trained, and tuned.

Experiment name: bank-additional

Input data

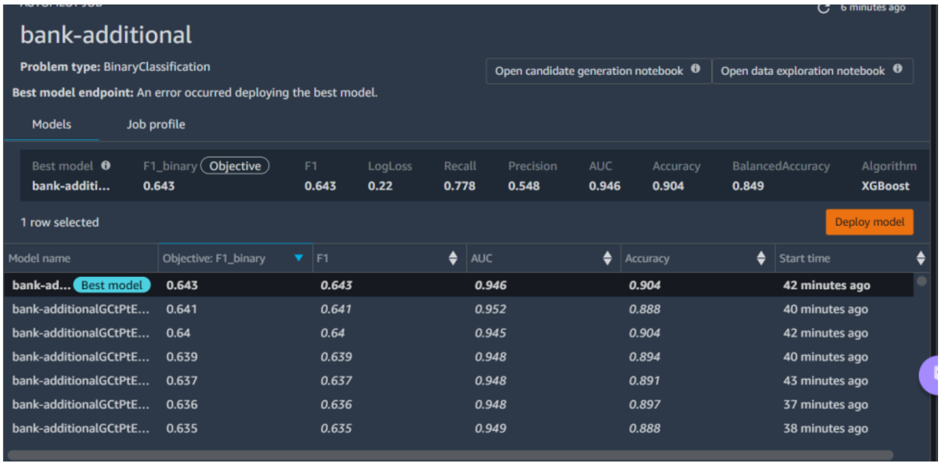
Specify the input data location and choose how to split your data.

S3 location: s3://sagemaker-us-east-1-595889462021/sagemaker/input/bank-additional-full.csv

Is your S3 input a manifest file? No

Target

Step 3: Choose the best tuning model and deploy the model to an endpoint managed by Amazon SageMaker.



bank-additional

Problem type: BinaryClassification

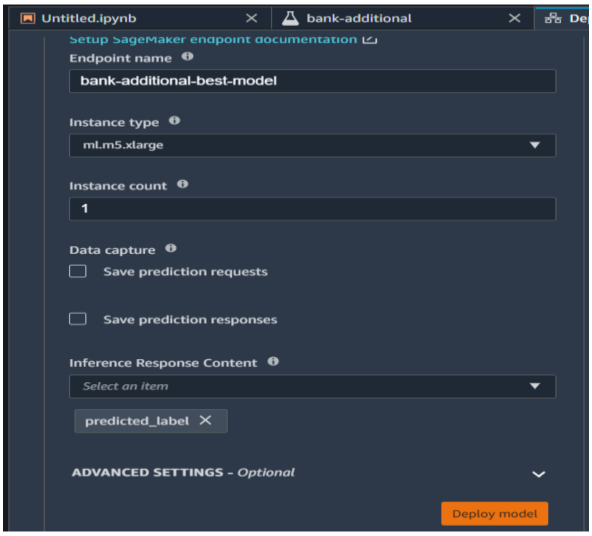
Best model endpoint: An error occurred deploying the best model.

Open candidate generation notebook
Open data exploration notebook
Models
Job profile

Best model	F1_binary Objective	F1	LogLoss	Recall	Precision	AUC	Accuracy	BalancedAccuracy	Algorithm
bank-additi...	0.643	0.643	0.22	0.778	0.548	0.946	0.904	0.849	XGBoost

1 row selected

Model name	Objective: F1_binary	F1	AUC	Accuracy	Start time
bank-ad... Best model	0.643	0.643	0.946	0.904	42 minutes ago
bank-additionalGCtPtE...	0.641	0.641	0.952	0.888	40 minutes ago
bank-additionalGCtPtE...	0.64	0.64	0.945	0.904	42 minutes ago
bank-additionalGCtPtE...	0.639	0.639	0.948	0.894	40 minutes ago
bank-additionalGCtPtE...	0.637	0.637	0.948	0.891	43 minutes ago
bank-additionalGCtPtE...	0.636	0.636	0.948	0.897	37 minutes ago
bank-additionalGCtPtE...	0.635	0.635	0.949	0.888	38 minutes ago



Endpoint name: bank-additional-best-model

Instance type: ml.m5.xlarge

Instance count: 1

Data capture

□ Save prediction requests

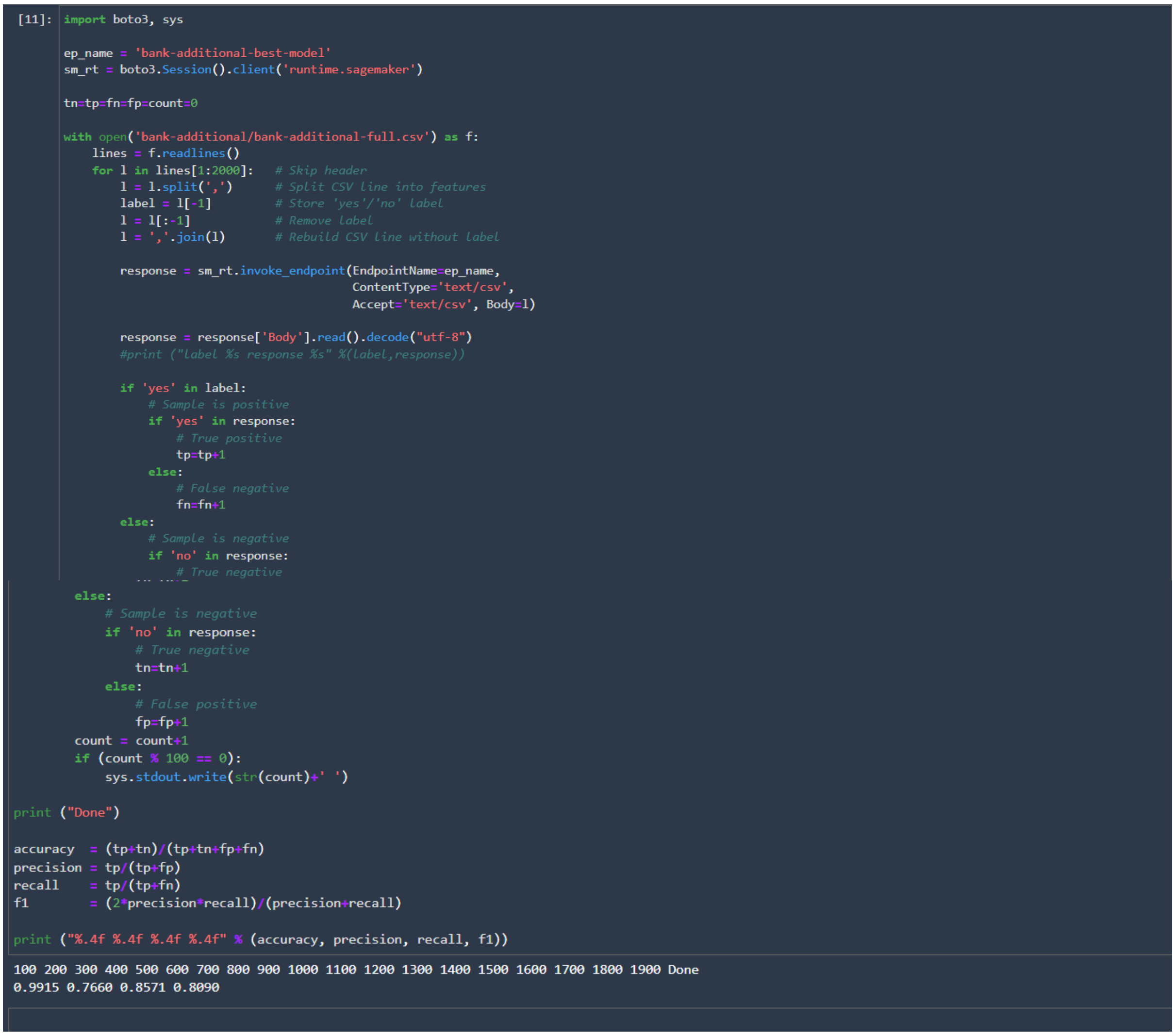
□ Save prediction responses

Inference Response Content: Select an item

predicted_label ✕

ADVANCED SETTINGS - *Optional*

Step-4: Predict the result



```
[11]: import boto3, sys

      ep_name = 'bank-additional-best-model'
      sm_rt = boto3.Session().client('runtime.sagemaker')

      tn=tp=fn=fp=count=0

      with open('bank-additional/bank-additional-full.csv') as f:
          lines = f.readlines()
          for l in lines[1:2000]:   # Skip header
              l = l.split(',')      # Split CSV line into features
              label = l[-1]         # Store 'yes'/'no' label
              l = l[:-1]            # Remove label
              l = ','.join(l)       # Rebuild CSV line without label

              response = sm_rt.invoke_endpoint(EndpointName=ep_name,
                                                ContentType='text/csv',
                                                Accept='text/csv', Body=l)

              response = response['Body'].read().decode("utf-8")
              #print ("label %s response %s" %(label,response))

              if 'yes' in label:
                  # Sample is positive
                  if 'yes' in response:
                      # True positive
                      tp=tp+1
                  else:
                      # False negative
                      fn=fn+1
              else:
                  # Sample is negative
                  if 'no' in response:
                      # True negative
        else:
            # Sample is negative
            if 'no' in response:
                # True negative
                tn=tn+1
            else:
                # False positive
                fp=fp+1
        count = count+1
        if (count % 100 == 0):
            sys.stdout.write(str(count)+' ')

print ("Done")

accuracy  = (tp+tn)/(tp+tn+fp+fn)
precision = tp/(tp+fp)
recall    = tp/(tp+fn)
f1        = (2*precision*recall)/(precision+recall)

print ("%.4f %.4f %.4f %.4f" % (accuracy, precision, recall, f1))
```

```
100 200 300 400 500 600 700 800 900 1000 1100 1200 1300 1400 1500 1600 1700 1800 1900 Done
0.9915 0.7660 0.8571 0.8090
```

Q5. List 5 startup using AWS ML. Give problem statement.

ICONY: Detecting and Handling Fake Accounts with Amazon Fraud Detector: In our digital world, as the popularity of online dating platforms and websites continues to grow, so too does the threat from scammers, bots, and other bad actors. ICONY GmbH, a white-label dating platform based in Germany, helps address this issue by rigorously validating users — allowing its business partners to launch their services with a database of reputable and up-to-date profiles already in place.

Carsome: Leveraging Automatic Car Plate Masking on Amazon SageMaker to Focus on Growth: Carsome is Southeast Asia's largest integrated car ecommerce platform. With operations across Malaysia, Indonesia, Thailand, and Singapore, we aim to digitize the region's used car industry by reshaping and elevating the car buying and selling experience. They provide end-to-end solutions to consumers and used car dealers, from car inspection to ownership transfer to financing, promising a service that is trusted, convenient and efficient. Carsome currently transacts around 100,000 cars annually and has more than 2,000 employees across all our offices.Typically, a car takes up to 30 minutes of inspection where our professional inspector will generate 175 points and take pictures annotating various notes about the car condition and appearance. As our business has grown rapidly, it's critical to streamline this process while maintaining high quality.We ran our experiment on Amazon SageMaker, which is a fully managed service for the whole workflow from training a deep neural network model into inference. It provides us with just what we need. SageMaker helps data scientists and developers to prepare, build, train, and deploy ML models quickly by bringing together a broad set of capabilities purpose-built for ML. We can build, train, tune, and deploy our model without having to think much about managing the infrastructure. The diagram below shows the high-level architecture.

Fintech Startup Floryn: Floryn is a fast-growing Dutch fintech startup helping small and medium sized enterprises (SMEs) grow by providing business loans. In two minutes, businesses can create an account, upload their bank statements, and get a credit decision within 24 hours. This all powered by artificial intelligence (AI) and machine learning (ML) technologies leveraging historic payment transactions to and from a business.

PulpoAR Uses Machine Learning to Build an Augmented Reality Shopping Experience for Beauty Product: PulpoAR performs 100 percent of its operations with AWS. Photos taken by users are uploaded and analyzed with Lambda functions before being sent back to the customer's browser. "Cloud solutions are very important for us because our technology requires serious device processing power," Bayat explains. "With AWS, we were able to access the same processing power on every device, and thus our users achieved better results."

Olive Builds the Internet of Healthcare and an AI Workforce on AWS: Today, the healthcare industry is flooded with software. Any given hospital has an EMR, billing software, different portals for every insurance partner, and individual medical tools each with their own interfaces, just to name a few. None of these systems work together, and the downstream effects dehumanizes the care experience. In effect, healthcare has used people as routers, forcing workers to toggle between numerous systems. They copy, they paste, and they endlessly toggle between software interfaces, becoming data entry, data transfer, and data interpretation robots. Doctors spend almost six hours on data entry during a typical workday. You can digitally trade stocks, transfer money, buy any product, sign a lease, and more instantly. Yet, patients are expected to constantly "log in" by filling out forms, regurgitating medical histories, showing paper cards, faxing claims and scheduling appointments over the phone with long wait times. Healthcare is essentially operating on dialup, and it's negatively affecting all parties involved. Similar to the Internet of Things (IoT), our products create the IoH. In the literal sense, the IoH means connecting networks – health systems, payers, software, data, patient information, and history. Essentially, we're transforming healthcare from a series of antiquated "fax machines" and human routers, to an advanced, secure internet connected by artificial intelligence. The result? Our autonomous interoperability platform enables humans to refocus on patients and care. The platform's workforce is a network, a brain. Each brain, or Olive, works 24/7, shares best practices and gets better at doing her job and helping her human counterparts every day. Today, Olive uses various technologies inside AWS including but not limited to AWS EKS, Amazon SageMaker, data lakes, AWS Lambda, AWS Lambda Step Functions, Amazon Redshift, and many more services to do her job.

Q6. What are some challenges in adopting AWS ML/Sagemaker?

- SageMaker does not allow you to schedule training jobs.
- SageMaker does not provide a mechanism for easily tracking metrics logged during training.
- We often fit feature extraction and model pipelines. We can inject the model artifacts into AWS-provided containers, but we cannot inject the feature extractors. We could provide our own container to SageMaker instead, but this is tantamount to serving the model ourselves.
- Amazon SageMaker is a great tool for a data scientist, although surprisingly, comparing different machine learning models with SageMaker is not as easy as one would think. I think Amazon needs to team up with a data scientist who does ensemble modeling.
- Because SageMaker is targeted for machine learning models, other models a data scientist might use require more effort to get them incorporated. My guess is Amazon is moving to make SageMaker a more complete tool.
- SageMaker can take a long time to run on larger data sets.
- Lack of Flexibility: Sometimes, SageMaker feels like a tool designed for companies to get started with machine learning that don't necessarily have any background in machine learning. It's not necessarily a tool for companies that have perspective and knowledge of the kind of machine learning problems and processes they want to adopt to their custom field. This is because it routinely sacrifices flexibility in favor of speed in performing basic operations. For example, while training a standard ML algorithm can be really easy, doing your own custom training can be a complete pain, because the SageMaker API's for it are underbaked and poorly documented. Additionally, the rush to be comprehensive on the part of SageMaker, rather than well-

documented, hints at an adoption strategy focused on IT and engineering architects, rather than developers themselves. These kinds of people tend to make such decisions at bigger companies.

- Unclear Costs: Keeping costs associated with machine learning down is a crucial objective. Managing costs with SageMaker as your needs scale is not intuitive, particularly given how its a wrapper around other AWS services. I'm not sure that it's a very cost effective solution in the long run for any company that plans to seriously spend money on building machine learning models.
- Lack of Community: Community is crucial for the uptake of a technical product. Often times, issues you encounter with a product are solved by the community (i.e. in Stack Overflow threads), not in official documentation. Many other machine learning products have thriving communities, like Comet.ml, DVC, TensorFlow, etc. Comparatively, I've been a little bit disappointed in the quality of the community output that's there for SageMaker. Most of the people on Github threads and other forums are SageMaker employees themselves, which is not the same as a diverse, thriving community applying a tool to novel problem areas and generating knowledge.

Submitted By: Amogh Garg
2020UCO1688
COE-3