AWS Certified Machine Learning - Specialty Practice Questions

Requirement: Share 65 ML Specialty practice questions.

Important Note: The practice questions should appropriately belong to ML Specialty in terms of exam objectives & difficulty level.

Delivery Timeline: < Date >

Question Response Types

There are two types of questions:

- Multiple Choice Single Response – **1** correct answer **3** incorrect responses (distractors).
- Multiple Choice Multiple Response – **2** or more correct answers out of **5** or more options.

Important Note

- Do write Question Number for quick identification. Q# 1, Q# 2 …. & so on.
- Please mention Domain (based on ML Specialty exam blueprint), Topic & Sub-Topic (If Applicable) with every question.
- Note that we're expecting standard scenario based questions & NOT straight-forward definition kind of questions.
- The options should not have any obviously incorrect option. We need to word the options so that all of them should appear correct for the students, but a subtle point should mark the correct answer without any ambiguity. So, one answer should be the best choice without any doubt.
- The answer / explanation section should contain explanations on why the answer is correct and others are incorrect. It should also contain the relevant resource link (for details) preferably from AWS documentation.

    - Example
        - Option A is incorrect because..
        - Option B is CORRECT because…
        - Option C is incorrect because..
        - Option D is incorrect because..

- Try to balance the domains based on weightage % defined in the exam blueprint.
- Any AWS service or feature must be approximately 6 months old to figure out in Practice Tests. Put a note in the explanation for any latest service or feature that might be on the borderline of appearing in the real exam.

- **Plagiarism** in any form - Question or in Explanation will be **rejected.** Questions & Explanations should reflect your own professional experience & AWS skills. Author's who indulge in plagiarism will be **blacklisted** & dropped from our author's list.
- The ownership of the questions once approved & published on Whizlabs LMS platform, lies solely with Whizlabs Software Pvt. Ltd. You can't share or publish it elsewhere in any circumstances.

Sample Format of Questions

Question : #
Main Topic : < >
Sub Topic : [optional]
Domain: < >

Question text:
<Scenario based. Should be clear in terms of requirements. No ambiguity. No duplicate options. In case of multiple answers, at the end, you should include number of expected answers. e.g. **Choose 2 answers**, choose 3 answers etc. For single answers this is NOT required>

A) Option A...
B) Option B...
C) Option C...
D) Option D...

Answer: A and C

Explanation:
Option A is CORRECT because...
Option B is incorrect because...
Option C is CORRECT because...
Option D is incorrect because...

[Insert the explanation in clear and lucid language here.]

Diagram: [Optional] [Insert the architectural or conceptual diagram here.]

Reference: [Insert the references here - which may include links to AWS Documentation, Blog, re:Invent video, Authority YouTube video].

ML Specialty has 4 Domains

S. No.	Name of the Domain	Weight	Estimated No. of Questions (out of 65 As per weightage %)
1	Data Engineering	20%	13
2	Exploratory Data Analysis	24%	16
3	Modeling	36%	23
4	ML Implementation and Operations	20%	13

--------------------------------------Question Section Starts--

Question: 1

Main Topic : Machine Learning

Sub Topic : **Create data repositories for machine learning**

Domain: Data Engineering

Question text:

You are a machine learning expert working for a marketing firm. You are supporting a team of data scientists and marketing managers who are running a marketing campaign. Your data scientists and marketing managers need to answer the question “Will this user subscribe to my campaign?” You have been given a dataset in the form of a CSV file which is formatted as such:

UserId, jobId, jobDescription, educationLevel, campaign, duration, willRespondToCampaign

When you build your schema for this dataset, which of the following data descriptors would you use to define the willRespondToCampaign attribute? Please choose 2 answers.

A) CATEGORICAL
B) targetAttributeName
C) TEXT
D) BINARY
E) Numeric
F) rowId

Answer: B and D

Explanation:

Option A is incorrect because you choose the CATEGORICAL data type for an attribute that holds a limited set of unique strings. For example, a user name, the region, and a product code are categorical values. The willRespondToCampaign attribute takes on either ‘yes’ or ‘no’ values, which are binary in nature.

Option B is correct because for each user observation you are trying to discern “Will this user subscribe to my campaign?” You assign the targetAttributeName field value to the name of the attribute that you are trying to predict. You must assign a targetAttributeName when you create or evaluate your model.

Option C is incorrect because you choose the TEXT data type for an attribute that is a string, or a set of words. Amazon ML converts text attributes into tokens and uses white space as a delimiter. For example, document title becomes document and title, and document-title here becomes document-title and here.

Option D is correct because you choose the BINARY data type for an attribute that only has two possible values, such as yes or no, or true or false. The attribute willRespondToCampaign has only two possible answers: yes or no.

Option E is incorrect because you choose the NUMERIC data type for an attribute that holds a quantity as a number. For example, count, height, and acceleration rate are numeric values.

Option F is incorrect because you choose the rowId field value as an optional flag associated with an attribute in the input data. If you specify an attribute as the rowId, it is included in the prediction output. This attribute allows you to associate each prediction with its observation. The willRespondToCampaign attribute would make a poor identifier for each observation since it only takes two values: yes, or no.

Reference:
Please see the AWS Developer Guide titled **Creating a Data Schema for Amazon ML** (https://docs.aws.amazon.com/machine-learning/latest/dg/creating-a-data-schema-for-amazon-ml.html#assigning-data-types) for a complete description of the schema attributes.

Question: 2
Main Topic : Machine Learning
Sub Topic : **Identify and implement a data-ingestion solution**

Domain: Data Engineering

Question text:
You work for an energy company that buys and sells energy to customers. To get the best prices for their energy customers, your company trades financial energy derivative futures contracts. The trading of these futures contracts requires accurate forecasting of energy prices. You need to build a model that compares spot prices (current commodity price) to future commodity prices (price that a commodity can be bought or sold in the future). Your model needs to assist your company's futures traders in hedging against future energy price changes based on current price predictions. To source the model with appropriate data you need to gather and process the energy price data automatically.

The data pipeline requires two sources of data:

1) Historic energy spot prices
2) Energy consumption and production rates

Based on the company analysts' requirements, you have decided you need multiple years of historical data. You also realize you'll need to update the data feed daily as the market prices change. You can gather the required data through APIs from data provider vendor systems. Your company's traders require a forecast from your model multiple times per day to help them form their trading strategy. So your pipeline needs to call the data provider APIs multiple times per day. Your data-ingestion pipeline needs to take the data from the API calls, perform preprocessing, and then store the data in an S3 data lake from which your forecasting model will access the data.

Your data-ingestion pipeline has three main steps:

1) Data ingestion
2) Data storage
3) Inference generation

Assuming you have written a lambda function that interacts with the data provider APIs and stores the data in CSV format, which of the following python libraries are the best option to perform the data preprocessing to transform the data by changing raw feature vectors into a format best suited for a SageMaker batch transform job to generate your forecast?

A) matplotlib and plotly
B) boto3 and moto
C) pandas and scikit-learn
D) NLTK and scrapy

Answer: C

Explanation:
Option A is incorrect because matplotlib and plotly are data visualization python libraries which contain no data transformation functions (see https://matplotlib.org and https://plot.ly/python/).

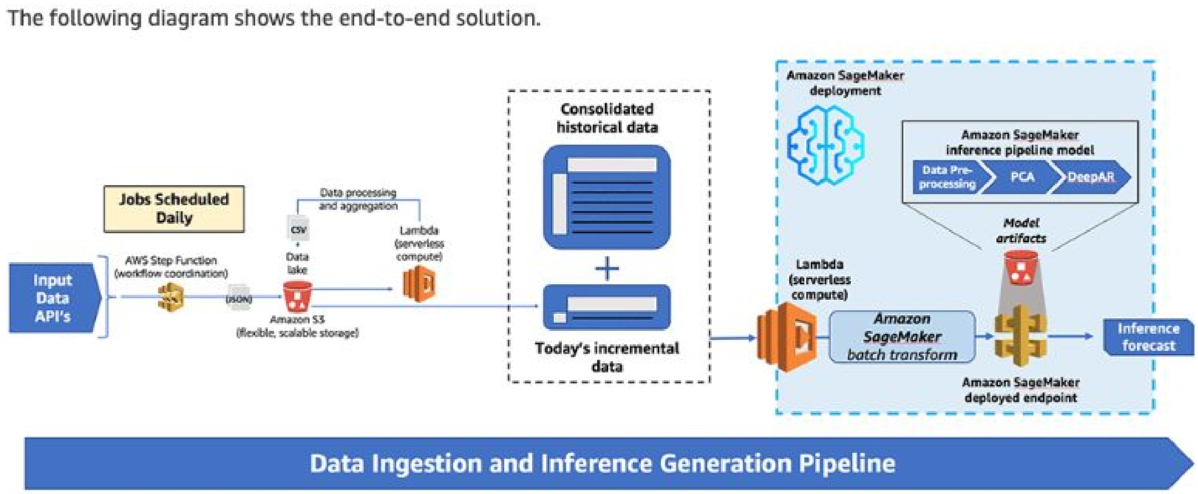
Option B is incorrect because boto3 is a python library that is used to interface with AWS services such as S3, DynamoDB, SQS, etc. Boto3 has no data transformation functions (see https://aws.amazon.com/sdk-for-python/). Moto is a python library used to mock interfaces to AWS services such as S3, DynamoDB, SQS, etc. The moto library also contains no data transformation functions (see https://pypi.org/project/moto/).

Option C is correct because pandas is the best choice for data wrangling and manipulation of tabular data such as CSV formatted data (see https://pypi.org/project/pandas/). Scikit-learn is the best python package to transform raw feature vectors into a format suited to downstream estimators (see https://scikit-learn.org/stable/modules/preprocessing.html).

Option D is incorrect because Natural Language Toolkit (NLTK) is best suited to text tagging, classification, and tokenizing, not manipulation of tabular data (see https://www.nltk.org). Scrapy is best suited to crawling functionality used to gather structured data from websites, not manipulation of tabular data (see https://scrapy.org).

Diagram:

Here is a screen shot from the AWS Machine Learning blog depicting the solution:



The following diagram shows the end-to-end solution.

Jobs Scheduled Daily
Input Data API's
AWS Step Function (workflow coordination)
JSON
CSV
Data lake
Amazon S3 (flexible, scalable storage)
Data processing and aggregation
Lambda (serverless compute)
Consolidated historical data
+
Today's incremental data
Amazon SageMaker deployment
Amazon SageMaker inference pipeline model
Data Pre-processing
PCA
DeepAR
Model artifacts
Lambda (serverless compute)
Amazon SageMaker batch transform
Amazon SageMaker deployed endpoint
Inference forecast
Data Ingestion and Inference Generation Pipeline

Reference:

Please see the scikit-learn preprocessing data documentation: https://scikit-learn.org/stable/modules/preprocessing.html, and a detailed pandas example: https://towardsdatascience.com/why-and-how-to-use-pandas-with-large-data-9594dda2ea4c

Question: 3

Main Topic : Machine Learning

Sub Topic : Perform hyperparameter optimization

Domain: Modeling

Question text:

You work for a retail firm that wishes to conduct a direct mail campaign to attract new customers. Your marketing manager wishes to get answers to questions that can be put into discrete categories, such as "using historical customer email campaign responses, should this customer receive an email from our current campaign?" You decide to use the SageMaker Linear Learner algorithm to build your model. Which hyperparameter setting would you use to get the algorithm to produce discrete results?

A) set the objective hyperparameter to reg:logistic.
B) set the predictor_type hyperparameter to binary_classifier.
C) set the predictor_type hyperparameter to regressor.
D) set the objective hyperparameter to reg:linear.

Answer: B

Explanation:
Option A is incorrect because the objective hyperparameter is set to reg:logistic when you are using the XGBoost algorithm (See the AWS SageMaker developer documentation: https://docs.aws.amazon.com/sagemaker/latest/dg/xgboost_hyperparameters.html).

Option B is correct because the AWS SageMaker documentation states that for this type of discrete classification problem, when using the Linear Learner algorithm, you set the predictor_type hyperparameter to binary_classifier (See the AWS SageMaker documentation: https://sagemaker.readthedocs.io/en/stable/linear_learner.html).

Option C is incorrect because the predictor_type hyperparameter is set to regressor when you are using the Linear Learner algorithm for answers that are quantitative, not discrete (See the AWS SageMaker documentation: https://sagemaker.readthedocs.io/en/stable/linear_learner.html).

Option D is incorrect because the objective hyperparameter is set to reg:linear when you are using the XGBoost algorithm for answers that are quantitative in nature (See the AWS SageMaker developer documentation: https://docs.aws.amazon.com/sagemaker/latest/dg/xgboost_hyperparameters.html).

Reference:
Please see the AWS SageMaker developer guide titled **Using Amazon SageMaker Built-in Algorithms**: https://docs.aws.amazon.com/sagemaker/latest/dg/algos.html) for a complete description of the SageMaker hyperparameter settings.

Question: 4
Main Topic : Machine Learning
Sub Topic : **Select the appropriate model(s) for a given machine learning problem**

Domain: Modeling

Question text:
You work for the information security department of a major corporation. You have been asked to build a solution that detects web application log anomalies to protect your organization from fraudulent activity. The system needs to have near-real-time updates to the model where log entry data points dynamically change the underlying model as the log files are updated. Which AWS service component do you use to implement the best algorithm based on these requirements?

A) SageMaker Random Cut Forest
B) Kinesis Data Streams Naive Bayes Classifier

C) Kinesis Data Analytics Random Cut Forest
D) Kinesis Data Analytics Nearest Neighbor

Answer: C

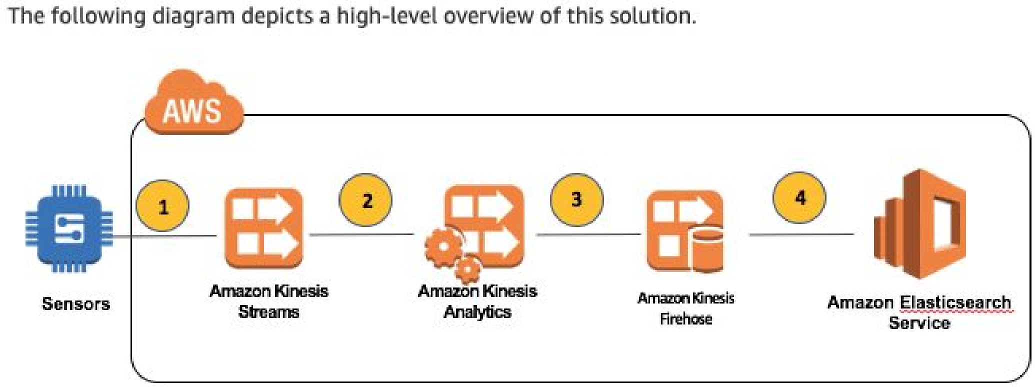
Explanation:
Option A is incorrect because SageMaker Random Cut Forest is best used for large batch data sets where you don't need to update the model frequently (See AWS Kinesis Data Analytics documentation:
https://docs.aws.amazon.com/kinesisanalytics/latest/sqlref/sqlrf-random-cut-forest.html).

Answer B is incorrect because the Naive Bayes Classifier is used to find independent data points. The Kinesis Data Streams service does not have machine learning algorithm capabilities (See the AWS Kinesis Streams developer documentation:
https://docs.aws.amazon.com/streams/latest/dev/introduction.html).

Option C is correct. The Kinesis Data Analytics Random Cut Forest algorithm works really well for near-real-time updates to your model (See the AWS Kinesis Data Analytics documentation:
https://docs.aws.amazon.com/kinesisanalytics/latest/sqlref/sqlrf-random-cut-forest.html).

Option D is incorrect because Kinesis Data Analytics provides a hotspots function that detects higher than normal activity using the distance between a hotspot and its nearest neighbor, but it does not provide ML model update capabilities (See AWS Kinesis Data Analytics documentation:
https://docs.aws.amazon.com/kinesisanalytics/latest/sqlref/sqlrf-hotspots.html).

Diagram:
Here is a screen shot from the AWS Big Data blog:



The following diagram depicts a high-level overview of this solution.

Reference:
For an example, please see the AWS Big Data blog post titled **Perform Near Real-time Analytics on Streaming Data with Amazon Kinesis and Amazon Elasticsearch Service**: https://aws.amazon.com/blogs/big-data/perform-near-real-time-analytics-on-streaming-data-with-amazon-kinesis-and-amazon-elasticsearch-service/) for a complete description of the use of Kinesis Data Analytics and the random cut forest algorithm.

Question: 5
Main Topic : Machine Learning
Sub Topic : Sanitize and prepare data for modeling

Domain: Exploratory Data Analysis

Question text:
You work in the data analytics department of a ride sharing software company. You need to use the K-means machine learning algorithm to separate your company's optimized ride data into clusters based on ride coordinates. How would you best use AWS Glue to build the data tables needed to classify the ride data?

A) Use Glue crawlers together with a K-means classifier to classify the ride data based on coordinates
B) Use Glue FindMatches to find and remove duplicate records in you data
C) Use Glue to automatically generate code to classify the ride data based on coordinates
D) Use Glue to transform and flatten your data so you can classify the ride data based on coordinates

Answer: A

Explanation:
Option A is correct. The best way to classify your optimized data is to use a Glue crawler that applies the K-means algorithm. See the AWS Machine Learning documentation (See the AWS SageMaker https://docs.aws.amazon.com/sagemaker/latest/dg/k-means.html and AWS Glue crawler https://docs.aws.amazon.com/glue/latest/dg/add-crawler.html documentation).

Answer B is incorrect because there is no stated need to remove duplicates from the data.

Option C is incorrect because you don't need to automatically generate code since Glue will classify your data based on a prioritized list of classifieds without custom code (See the AWS Glue developers guide: (https://docs.aws.amazon.com/glue/latest/dg/add-classifier.html).

Option D is incorrect because there is no stated requirement to flatten the ride data.

Reference:
For an example, please see the AWS Machine Learning blog post titled **Serverless unsupervised machine learning with AWS Glue and Amazon Athena**: https://aws.amazon.com/blogs/machine-learning/serverless-unsupervised-machine-learning-with-aws-glue-and-amazon-athena/.

Question: 6
Main Topic : Machine Learning
Sub Topic : Build machine learning solutions for performance, availability, scalability, resiliency, and fault tolerance

Domain: Machine Learning Implementation and Operations

Question text:
You work in the security department of your company's IT division. Your company has decided to try to use facial recognition to improve security on their campus. You have been asked to design a system that augments your company's building access security by scanning the faces of people entering their buildings and recognizing the person as either an employee/contractor/consultant, who is in the company's database, or visitor, who is not in their database.

Across their many campus locations worldwide your company has over 500,000 employees and over 100,000 contractors and consultants. These workers are all registered in their HR database. Each of these workers has five images of their face stored (straight on, face turned left, face turned right, face tilted down, face tilted up). Based on this large set of data to process, which of the following design choices is the quickest and simplest to build while also scaling to perform at the required level to implement a real-time access decision based on the video image of a person arriving at the entrance to a company building?

A) Build a facial recognition model using the cascade classifier technique implemented with the OpenCV and face_recognition python libraries
B) Build a facial recognition model using the Multi-task Cascaded Convolutional neural network framework implemented in the MTCNN Keras python library
C) Use the Amazon Rekognition API to implement the facial recognition algorithm
D) Use the AWS Apache MXNet Gluon interface to build a facial recognition model using a linear regression, convolutional network and recurrent LSTM

Answer: C

Explanation:
Option A is incorrect because you are trying to quickly build a model that scales to you required data volumes. Creating your model using open source python libraries will not give you a performant model that scales massively without a lot of effort. (See this example: https://www.pyimagesearch.com/2018/06/18/face-recognition-with-opencv-python-and-deep-learning/)

Option B is incorrect for the same reason option A is incorrect. Creating your model using the MTCNN python library will require significant effort and will not give you a performant system at the scale needed for your solution. (See the MTCNN documentation: https://pypi.org/project/mtcnn/)

Option C is correct. Amazon Rekognition allows you to create a facial recognition system that scales massively across AWS's resources. Building a system using Rekognition requires no machine learning expertise (See the Amazon Rekognition documentation: https://aws.amazon.com/rekognition/)

Option D is incorrect because the Apache MXNet AWS service is used mainly for object detection, speech recognition, recommendations, and personalization. It is not the best choice available via AWS services for a facial recognition system. (See the Apache MXNet on AWS documentation: https://aws.amazon.com/mxnet/)

Reference:
Please see the AWS Deep Learning on AWS documentation: https://aws.amazon.com/deep-learning/.

Question: 7
Main Topic : Machine Learning

Sub Topic : Recommend and implement the appropriate machine learning services and features for a given problem

Domain: Machine Learning Implementation and Operations

Question text:
Your marketing department wishes to understand how their products are being represented in the various social media services in which they have active content streams. They would like insights into the reception of a current product line so they can plan for the roll out of a new product in the line in the new future. You have been tasked with creating a service that organizes the social media content by sentiment across all languages so that your marketing department can determine how best to introduce the new product.

How would you quickly and most efficiently design and build a service for your marketing team that gives insight into the social media sentiment?

A) Use the scikit-learn python library to build a sentiment analysis service to provide insight data to the marketing team's internal application platform. Build a dashboard into the application platform using React or Angular.
B) Use the DetectSentiment Amazon Comprehend API as a service to provide insight data to the marketing team's internal application platform. Build a dashboard into the application platform using React or Angular.
C) Use the Amazon Lex API as a service to implement the to provide insight data to the marketing team's internal application platform. Build a dashboard into the application platform using React or Angular.
D) Use Amazon Translate, Amazon Comprehend, Amazon Kinesis, Amazon Athena, and Amazon QuickSight to build a natural-language-processing (NLP)-powered social media dashboard

Answer: D

Explanation:
Option A is incorrect since this option is not the quickest to implement nor is it the most efficient, since developers will have to code a react UI and build an end-point to connect the sentiment service to the React or Angular UI.

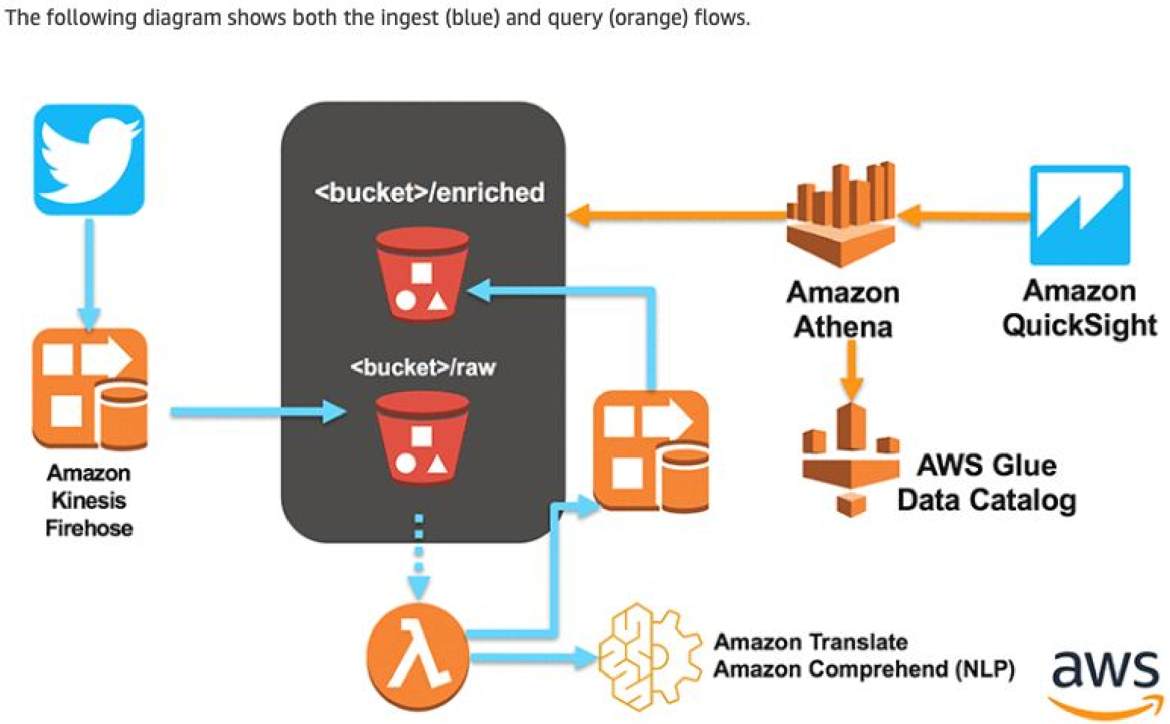
Option B is incorrect since this option is also not the quickest to implement nor is it the most efficient, since developers will have to code a react UI and build an end-point to connect the sentiment service to the React or Angular UI.

Option C is incorrect since Amazon Lex is used primarily for building conversational interfaces into an application using voice or text. This would not give you the most efficient solution to the problem.

Option D is correct since it is the most efficient and quickest way to implement the solution. Amazon Kinesis Data Firehose is used to capture and prepare the social media content. A lambda function can be used to analyze the social media content using Amazon Translate and Amazon Comprehend. Amazon Athena to query the data produced by the lambda function. Use Amazon QuickSight to produce the dashboard. (See the AWS Machine Learning blog post titled: **Build a social media dashboard using machine learning and BI services:https://aws.amazon.com/blogs/machine-learning/build-a-social-media-dashboard-using-machine-learning-and-bi-services/**)

Diagram:

Here is a screen shot from the AWS Machine Learning blog that shows the desired solution:



The following diagram shows both the ingest (blue) and query (orange) flows.

Reference:

Please see the Amazon Comprehend documentation: https://aws.amazon.com/comprehend/.

Question: 8

Main Topic : Machine Learning

Sub Topic : Perform feature engineering

Domain: Exploratory Data Analysis

Question text:

You work for a financial services firm that wishes to further enhance their fraud detection capabilities. The firm has implemented fine grained transaction logging for all transactions their customers make using their credit cards. The fraud prevention department would like to use this data to produce dashboards to give them insight into their customer's transaction activity and to provide real-time fraud prediction.

You plan to build a fraud detection model using the transaction observation data with Amazon SageMaker. Each transaction observation has a date-time stamp. In its raw form, the date-time stamp is not very useful in your prediction model since it is unique. Can you make use of the date-time stamp in your fraud prediction model, and if so how?

A) No, you cannot use the date-time stamp since this data point will never occur again. Unique features like this will not help identify patterns in your data.
B) Yes, you can use the date-time stamp data point. You can just use feature selection to deselect the date-time stamp data point, thus dropping it from the learning process.
C) Yes, you can use the date-time stamp data point. You can transform the date-time stamp into features for the hour of the day, the day of the week, and the month.
D) No you cannot use the date-time feature since there is no way to transform it into a unique data point.

Answer: C

Explanation:
Option A is incorrect since you can use the date-time stamp if you use feature engineering to transform the data point into useful form.

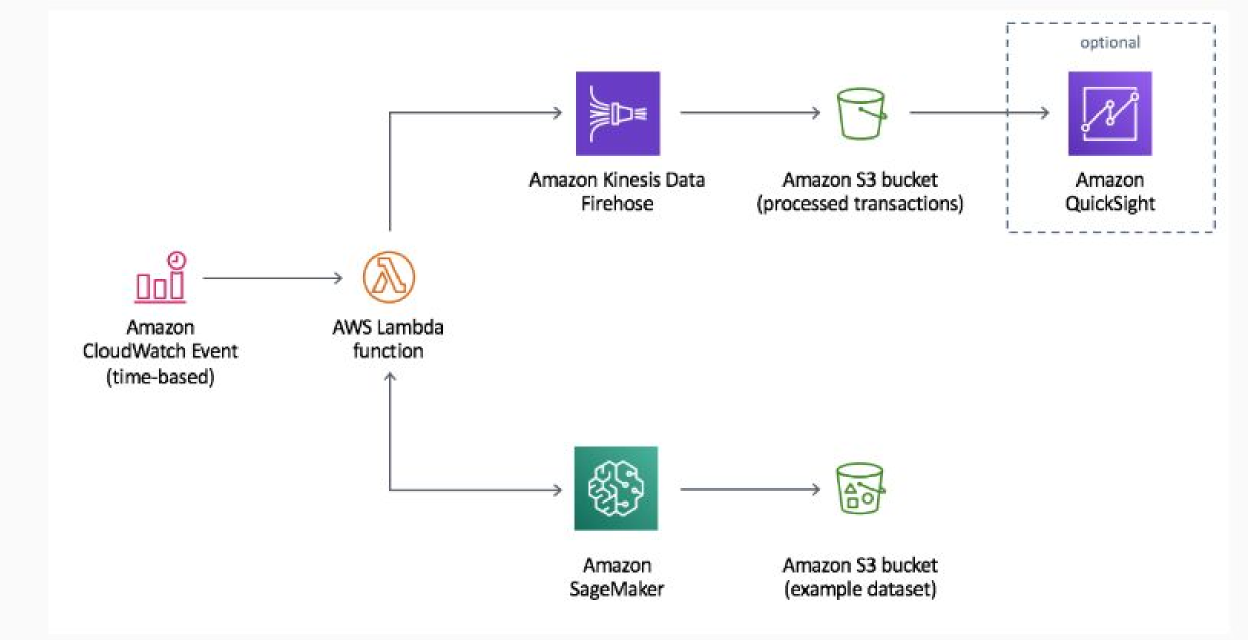
Option B is incorrect since this option is really just another way of ignoring, thus not using, the date-time stamp data point.

Option C is correct. You can transform the data point using feature engineering and thus gain value from it for the learning process of your model. (See the AWS Machine Learning blog post: **Simplify machine learning with XGBoost and Amazon SageMaker:** **https://aws.amazon.com/blogs/machine-learning/simplify-machine-learning-with-xgboost-and-amazon-sagemaker/)**

Option D is incorrect since we can transform the data point into unique features that represent the hour of the day, the day of the week, and the month, these variables could be useful to learn if the fraudulent activity tends to happen at a particular hour, day of the week, or month.

Diagram:

Here is a screen shot from the AWS Machine Learning documentation depicting a typical fraud detection machine learning solution:


optional
Amazon Kinesis Data Firehose
Amazon S3 bucket (processed transactions)
Amazon QuickSight
Amazon CloudWatch Event (time-based)
AWS Lambda function
Amazon SageMaker
Amazon S3 bucket (example dataset)

Reference:

Please see the Amazon Machine Learning developer documentation: https://docs.aws.amazon.com/machine-learning/latest/dg/feature-processing.html.

Question: 9

Main Topic : Machine Learning

Sub Topic : **Train machine learning models**

Domain: Modeling

Question text:

You work for a real estate company where you are building a machine learning model to predict the prices of houses. You are using a regression decision tree. As you train your model you see that it is overfitted to your training data and that it doesn't generalize well to unseen data. How can you improve your situation and get better training results in the most efficient way?

A) Use a random forest by building multiple randomized decision trees and averaging their outputs to get the predictions of the housing prices.

B) Gather additional training data that gives a more diverse representation of the housing price data.

C) Use the “dropout” technique to penalize large weights and prevent overfitting.
D) Use feature selection to eliminate irrelevant features and iteratively train your model until you eliminate the overfitting.

Answer: A

Explanation:
Option A is correct because the random forest algorithm is well known to increase the prediction accuracy and prevent overfitting that occurs with a single decision tree. (See these articles comparing the decision tree and random forest algorithms: https://medium.com/datadriveninvestor/decision-tree-and-random-forest-e174686dd9eb and https://towardsdatascience.com/decision-trees-and-random-forests-df0c3123f991)

Option B is incorrect since gathering additional data will not necessarily improve the overfitting problem, especially if the additional data has the same noise level of the original data.

Option C is incorrect since while the “dropout” technique improves models that are overfitted, it is a technique used with neural networks, not decision trees.

Option D is incorrect since it requires significantly more effort than using the random forest algorithm approach.

Reference:
Please see this overview of the random forest machine learning algorithm: https://medium.com/capital-one-tech/random-forest-algorithm-for-machine-learning-c4b2c8cc9feb

Question: 10
Main Topic : Machine Learning
Sub Topic : **Analyze and visualize data for machine learning**

Domain: Exploratory Data Analysis

Question text:
You work for the data analytics department of your company where you have been asked to build a visualization of the company’s corporate performance for the company’s annual report. The visualization needs to demonstrate company performance by showing how likely it is for a customer to recommend your company’s products, and how much profit a customer brings to the business net acquisition and retention costs. Which types of charts would you use to create this visualization? Choose 2 answers.

A) Use a distribution scatter chart to show the customer recommendation rate

B) Use a Conversion Rate KPI chart to show conversion rate of customers
C) Use a Relative Market Share KPI chart to show competitive market share
D) Use a Net Promoter Score KPI chart to graph customer recommendations
E) Use a Customer Profitability Score KPI chart to show customer profitability

Answers: D and E

Explanation:
Option A is incorrect because a distribution scatter chart would show the size of recommendations but not the likelihood of a recommendation.

Option B is incorrect since a Conversion Rate KPI shows how many leads were converted to customers, not likelihood of recommendation or customer profitability.

Option C is incorrect since the Relative Market Share KPI shows how much market share your company owns versus your company's competitors.

Option D is correct since the Net Promoter Score KPI shows how likely it is a current customer would recommend you company's products.

Option E is correct since the Customer Profitability Score KPI shows how much profit a customer contributes to your company's profits after the expenses of acquiring the customer and the expenses associated with retaining the customer.

Reference:
Please see this AWS overview of analyzing and visualizing your data for business analytics: https://aws.amazon.com/data-visualization/

Question: 11
Main Topic : Machine Learning
Sub Topic : **Frame business problems as machine learning problems**

Domain: Modeling

Question text:
You work for a manufacturing company that produces retail apparel, such as shoes, dresses, blouses, etc. Your head of manufacturing has asked you to use your data science skills to determine which product, among a list of potential next products, your company should invest its resources to produce. You decide you need to predict the sales levels of each of the potential next products and select the one with the highest predicted purchase rate. Is this a problem that

you should solve using machine learning? If so, which type of machine learning approach should you use?

A) No, this is not a problem that requires machine learning. This problem can be solved using traditional programming techniques using predetermined steps.
B) No, this is not a problem that requires machine learning. All the rules needed to discern which product to produce next product are clearly evident.
C) Yes, this problem would be best solved using machine learning. You are trying to solve for the greatest number of sales across the potential next products. Therefore, you are solving a regression problem and you should use a linear regression model.
D) Yes, this problem would be best solved using machine learning. You are trying to solve for the greatest number of sales across the potential next products. Therefore, you are solving a binary classification problem and you should use a logistic regression model.

Answer: C

Explanation:
Option A is incorrect because this problem is a classic numeric target regression problem. The problem would be solved much more accurately using machine learning versus classic programming using predetermined steps. (See the AWS Machine Learning developer guide: https://docs.aws.amazon.com/machine-learning/latest/dg/when-to-use-machine-learning.html and https://docs.aws.amazon.com/machine-learning/latest/dg/formulating-the-problem.html)

Option B is incorrect for the same reason as option A. The problem would be solved much more accurately using machine learning versus versus using simple rules based programming. (See the AWS Machine Learning developer guide: https://docs.aws.amazon.com/machine-learning/latest/dg/when-to-use-machine-learning.html and https://docs.aws.amazon.com/machine-learning/latest/dg/formulating-the-problem.html)

Option C is correct. You are trying to solve for a numeric result: the number of purchases customers will make for each next potential product. This numeric result case calls for the use of a regression model such as the linear regression model.

Option D is incorrect. The problem is best solved using machine learning, however this is not a classification problem, where you're solving for a binary (yes or no) result, so a logistic regression model would be the wrong choice for your machine learning.

Reference:
Please see this AWS overview of machine learning concepts:
https://docs.aws.amazon.com/machine-learning/latest/dg/machine-learning-concepts.html

Question: 12

Main Topic : Machine Learning
Sub Topic : Apply basic AWS security practices to machine learning solutions

Domain: Machine Learning Implementation and Operations

Question text:
You work as a manager of data scientists at a large financial services firm where your team is responsible for building machine learning solutions such as price prediction of equities, futures, and options. You need petabytes of data from dozens of sources internal and external to your organization. All of the external data sources are contractually constrained as to where the data is used and who has access to the data. Your machine learning models require storage of these data in a data lake to allow for quick retrieval of data to fuel your ML models. You have chosen to use S3 to house your data lake. How will you most efficiently protect this data lake, your machine learning data source, against internal threats to data confidentiality and security?

A) Create IAM resource-based policies for each data lake S3 bucket resource. Use bucket policies and Access Control Lists (ACLs) to control the resources at the bucket level and at the object level.
B) Create IAM user policies so that permissions to access your S3 data lake assets are linked to user roles and permissions. Place your data scientists into IAM groups and assign the user policies to those groups. These policies and permissions will define access to the data processing and analytics services which your data scientists will use.
C) Create an access key ID and a secret access key for each internal user of your S3 data lake. Your internal users will then only be able to gain access to your data lake using these keys.
D) Use the AWS CloudHSM cloud-based hardware security module (HSM) to secure your S3 data lake. Internal users of your data lake will use the encryption keys generated by the CloudHSM module to gain access to the data needed for their machine learning models.

Answer: B

Explanation:
Option A is incorrect because this is a very inefficient approach to the problem of securing a data lake. Most large data lakes contain large numbers of buckets and objects. Using resource-based policies would mean creating a very large set of policies to secure the data lake.

Option B is correct. Per the AWS white paper Building Big Data Storage Solutions (Data Lakes) for Maximum Flexibility - Securing, Protecting, and Managing Data, “for most data lake environments, we recommend using user policies, so that permissions to access data assets can also be tied to user roles and permissions for the data processing and analytics services and tools that your data lake users will use.”

Option C is incorrect because access keys are used primarily for applications running outside the AWS environment. Resources running inside AWS, as is the case in this scenario, the best practice is to use IAM roles and policies. (See AWS Security blog entry **Guidelines for protecting your AWS account while using programmatic access: https://aws.amazon.com/blogs/security/guidelines-for-protecting-your-aws-account-while-using-programmatic-access/**)

Option D is incorrect. The CloudHSM module is used to generate encrypted access keys. However, since we're dealing with users internal to AWS, IAM role-based security is the best practice for this scenario.

Reference:
Please see the AWS white paper Building Big Data Storage Solutions (Data Lakes) for Maximum Flexibility - Securing, Protecting, and Managing Data

Question: 13
Main Topic : Machine Learning
Sub Topic : **Sanitize and prepare data for modeling**

Domain: Exploratory Data Analysis

Question text:
You work for a financial services company where you are building a model to analyze equity futures prices to predict price movement for your firm's hedging strategy. You receive several data feeds, some of which contain missing values for some data points. The missing data points in your data feeds are of the categorical type, such as the expiration month or the exchange on which the futures contract is traded. Which strategy should you employ to deal with the missing data point values while attempting to maximize the accuracy of your model without introducing bias into the model?

A) Remove the observations that have the missing data.
B) Impute the missing values using the Mean/Median strategy.
C) Impute the missing values using the Most Frequent strategy.
D) Impute the missing values using a Deep Learning strategy

Answer: D

Explanation:
Option A is incorrect because this approach will lead to the loss of data points with potentially useful information.

Option B is incorrect because, by definition, it can only be used with numeric data. It is not advisable to use the Mean/Median approach with categorical data points.

Option C is incorrect because while working with categorical data, this method can introduce bias into your data.

Option D is correct. A deep learning approach, using a library such as the datawig python library, uses deep neural networks to impute missing data values and is the most accurate strategy, in the list of given options, at imputing categorical values. (See the datawig documentation: https://github.com/awslabs/datawig)

Reference:
Please see the article 6 Different Ways to Compensate for Missing Values In a Dataset (Data Imputation with examples)

Question: 14
Main Topic : Machine Learning
Sub Topic : **Evaluate machine learning models**

Domain: Modeling

Question text:
You are a data scientist working for a cancer screening center. The center has gathered data on many patients that have been screened over the years. The data is obviously skewed toward true negative results, as most screened patients don’t have cancer. You are evaluating several machine learning models to decide which model best predicts true positives when using your cancer screening data. You have split your data into a 70/30 ratio of training set to test set. You now need to decide which metric to use to evaluate your models.

Which metric will most accurately determine the model best suited to solve your classification problem?

A) ROC Curve
B) Precision
C) Recall

D) PR Curve

Answer: D

Explanation:
Option A is incorrect because it is best used when both outcomes have equal importance. Due to the importance of true negative in this equation, it will not differentiate models well for the cancer screening problem, since this data set is skewed to true negatives. The true negative cases are heavily weighted in the equation, thus amplifying the impact of the imbalance.

Option B is incorrect because it only takes into account the percentage of positive cases out of the total predicted positive.

Option C is incorrect because it only takes into account the percentage of positive cases out of the total actual positive.

Option D is correct because the PR Curve is best used to evaluate models on data sets where most of the cases are negative, as in the cancer screening data set. The true negative cases are not weighted heavily in the equation, thus reducing the impact of the imbalance.

Reference:
Please see the article Various ways to evaluate a machine learning model's performance

Question: 15
Main Topic : Machine Learning
Sub Topic : Perform Feature Engineering

Domain: Exploratory Data Analysis

Question text:
You work for a mining company where you are responsible for the data science behind identifying the origin of mineral samples. Your data origins are Canada, Mexico, and the US. Your training data set is imbalanced as such:

Canada	Mexico	US
1,210	120	68

You run a Random Forest classifier on the training data and get the following results for your test data set (your test data set is balanced):

Confusion matrix:

	Predicted			
Observed	Canada	Mexico	US	Accuracy
Canada	45	3	0	94%
Mexico	5	38	5	79%
US	19	8	21	44%

In order to address the imbalance in your training data you will need to use a preprocessing step before you create your SageMaker training job. Which technique should you use to address the imbalance?

A) Run your training data through a preprocessing script that uses the SMOTE (Synthetic Minority Over-sampling Technique) approach
B) Run your training data through a Spark pipeline in AWS Glue to one-hot encode the features
C) Run your training data through a preprocessing script that uses the feature-split technique
D) Run your training data through a preprocessing script that uses the min-max normalization technique

Answer: A

Explanation:

Option A is correct. The SMOTE sampling technique uses the k-nearest neighbors algorithm to create synthetic observations to balance a training data set. (See the article SMOTE Explained for Noobs)

Option B is incorrect because the Spark pipeline creates one-hot encoded columns in your data. One-hot encoding is a process for converting categorical data points into numeric form. This won’t do anything to address the imbalance in your training data. (See this explanation of one-hot encoding)

Option C is incorrect because it splits a feature (data point) in your observations into multiple features per observation. This also will have no impact on your imbalanced training data. (See the article Fundamental Techniques of Feature Engineering for Machine Learning)

Option D is incorrect because the min-max normalization technique is used to normalize data points into a range of 0 to 1, for example. (See the wikipedia article Feature Scaling)

Reference:

Please see the article How to Handle Imbalanced Classification Problems in machine learning