**Module 1**

**Recommended AWS Knowledge**

The successful candidate likely has 1–2 years of hands-on experience developing, architecting, or running ML/deep learning workloads on the AWS Cloud, along with:

* The ability to express the intuition behind basic ML algorithms
* Experience performing basic hyperparameter optimization
* Experience with ML and deep learning frameworks
* The ability to follow model-training best practices
* The ability to follow deployment and operational best practices

**Exam validation points**

1. Select and justify the appropriate ML approach for a given business problem.
2. Identify appropriate AWS services to implement ML solutions.
3. Design and implement scalable, cost-optimized, reliable, and secure ML solutions.

**Exam Details**

* It is 170 minutes
* ~65 questions
* There are 5,000 test centers.
* There are 2 test Vendors:
  + PSI
  + Pearson VUE:
    - [What to expect in a Pearson Vue test center](https://home.pearsonvue.com/Test-takers/Resources.aspx#what-to-expect)
    - [What to expect when taking a Pearson VUE Remote Proctored Exam](https://home.pearsonvue.com/Test-takers/OnVue-online-proctoring.aspx)
* Two forms of IDs are required. One must be a government-issued ID with a photo.

**Exam Mechanism**

**Questions preview**

**Response Types**

* There are two types of questions on the examination:
  + Multiple choice: Has one correct response and three incorrect responses (distractors).
  + Multiple response: Has two or more correct responses out of five or more options.
* Select one or more responses that best complete the statement or answer the question.
* Distractors, or incorrect answers, are response options that an examinee with incomplete knowledge or skill would likely choose. However, they are generally plausible responses that fit in the content area defined by the test objective.
* Questions are designed to include only the information you need.
* Questions won't be tested on:
  + Services and features that are new for less than 6 months.
  + Information that's likely to change, like pricing and performance metrics.
* Unanswered questions are scored as incorrect; there is no penalty for guessing.

**Scoring**

* Unscored Content:
  + Your examination may include unscored items that are placed on the test to gather statistical information. These items are not identified on the form and don ot affect your score.
* Exam Results:
  + It is a pass or fail exam.
  + Exam score from 100–1,000, with a minimum passing score of 750.
  + Scores will be available immediately upon completion.
  + Scaled scoring models are used to equate scores across multiple exam forms that may have slightly different difficulty levels.
  + score report contains a table of classifications of your performance at each section level, to identify which domain needs improvement and which meets compliance.
  + No need to pass each individual domain

**Content Outline**

* It is divided to 4 domains
  + 36% Modeling: Designing algorithms, tweaking hyperparameters
  + 20% Data engineering
  + 24% Exploratory data analysis
  + 20% ML implementation and operations
* In a Foundational or Associate Level Exams, questions are based on Understanding and Recall. While in Specialty Exams, questions requires evaluation, analyzes and applying knowledge to the situation.

**Examination strategy**

* During the exam, strategies to keep in mind:
  + Read and understand the question before reading answer options.
  + Identify the key phrases (Important words) and qualifiers.
  + Anticipate the answer before even looking at the answer choices
  + Eliminate options based on what you know about the question/topic
  + Compare and contrast remaining answer options
  + Consider flagging question and moving on to easier questions.

**Module 2**

**Domain 1: Data Engineering**

**1.1 Create data repositories for machine learning**

**Services related to this domain**

* AWS Lake Formation
* Amazon S3 (as a data lake)
* Amazon FSx for lustre
* Amazon EFS
* Amazon EBS volumes
* Amazon S3 life cycle configurations
* Amazon S3 data storage options

**Data Repository**

* Data Lake is a centralized, secured repository that stores structured and unstructured data.
* Data Lake formation helps to build Data Lake

**Amazon S3 data storage options**

* S3 Standard:
  + Active, frequently accessed data
  + Milliseconds access
  + >= 3 Availability Zone
  + Most expensive
* S3 Intelligent-Tiering (S3 Int):
  + Data with changing access patterns
  + Milliseconds access
  + >= 3 Availability Zone
* S3 Standard-IA (S3 S-IA):
  + Infrequently accessed data
  + Milliseconds access
  + >= 3 Availability Zone
* S3 One Zone-IA (S3 1Z-IA):
  + Re-creatable, less accessed data
  + Milliseconds access
  + 1 Availability Zone
* S3 Glacier (Amazon Glacier):
  + Archive Data
  + Select minutes or hours
  + >= 3 Availability Zone
  + Least expensive

**Amazon S3 and SageMaker**

Amazon S3 integrates with SageMaker as a storage solution

* Training data input
* Storing Model artifacts

**Amazon S3 and Amazon FSx for Lustre**

Amazon FSx for Lustre speeds up training jobs by serving data to Amazon SageMaker

* It is designed to be high performance
* It used to store training data from S3 to provide SageMaker with a very high throughput.

**Amazon EFS**

* If the data is already stored in EFS, it doesn't required to be moved to S3 to be used in SageMaker or Notebooks

**File systems load time**

* Amazon S3: < 1.00
* Amazon EFS: 1
* Amazon EBS: 1.29
* Amazon FSx: > 1.6

Comparisons of the relative to Amazon EFS images per second that each file system can load

**1.2 Identify and implement a data-ingestion solution**

How to ingest the data into the data lakes

1. AWS Glue Batching:
   * Batch processing periodically collects and groups source data
   * Several services can help with batch processing into AWS cloud.
   * Unstructured Data ingested into an S3 bucket then data inserted to AWS Glue to run a SPARK job that cleans the data.
2. Kinesis:
   * Amazon Kinesis is a platform for streaming data on AWS
   * It can be used for ingesting and processing stream data.
   * Kinesis Video Streams:
     + Securely stream video from connected devices to AWS for analytics, machine learning, playback, and other processing.
   * Kinesis Data Streams:
     + Continuously capture and streamed in near real-time.
   * Kinesis Data Firehose:
     + Batch and compress the data to generate incremental views
   * Kinesis Data Analytics:
     + Process the data that is streaming through Kinesis Data Streams or Kinesis Data Firehose using SQL
   * Amazon Kinesis Client Library (KCL):
     + Reads the data from Amazon Kinesis
   * Amazon Kinesis Producer Library (KPL):
     + Writes the data to other services
3. Amazon Managed Streaming for Apache Kafka (Amazon MSK)

**1.3 Identify and implement a data-transformation solution**

* Raw ingested data is not ML-ready, so it requires to a middle service for data preparation ETL.

1. Amazon Glue:
   * Glue can be used in this spot, but for traditional Apache Spark jobs.
2. Amazon EMR:
   * It is mainly used for Hadoop environments. So, if there is already existing workloads coded in Hadoop using mass produce or even for Spark.
   * It can be used for preparing, cleansing, partitioning.

**Domain 2: Exploratory Data Analysis**

**2.1 Sanitize and prepare data for modeling**

* Data is ingested and transformed, but it's still messy and misunderstood

**Dataset generation**

* Amazon SageMaker Ground Truth
* Amazon Mechanical Turk

**Data augmentation**

* Standardize language and grammar
* Make sure the data is on the same scale
* Make sure a column doesn't include multiple features
* Outliers are data points that lie at abnormal distance from other values
  + Outliers can:
    - Add richness to the data.
    - Make accurate predictions more difficult
    - Indicate that the data point belongs to another column

**Descriptive statistics**

* Use descriptive statistics to gain insights into your data before cleaning the data.
* Overall stats helps to understand the dimensions of the data.
  + Number of instances (rows)
  + Number of attributes (columns)
* Attribute stats help to understand the shape of attributes
  + Numeric attributes (mean, variance, etc)
  + Categorical attributes (histograms, mode, most/least frequent values, percentage, number of unique values)
  + Target attributes (class distribution)
* Categorical stats identify frequency of values and class imbalances

**Informative statistics**

* Multivariate stats identify relationships between attributes
  + Correlation:
    - Identifying it is important, bec. they can impact model performance.
    - Correlation matrices measures the linear dependance between features.
    - They can be visualized with heat maps
  + Contingency tables
  + Scatter plots visualize relationships between numerical variables

**Handling missing values/outliers**

* Missing data makes it hard to interpret a feature/target relationship
  + Techniques for missing data:
    - Removing rows or columns
    - Impute missing values with Mean, Zero, regression

**2.2 Perform feature engineering**

* Feature engineering gives your model stronger prediction power
* Feature engineering may be necessary because of the data dimensionality
* Data dimensionality reduction techniques include:
  + Principle component analysis (PCA)
  + t-distributed stochastic neighbor embedding
* For numerical features, data transformation can be applied
* Categorical data can be converted to numerical data.
  + Binary class can be encoded to 0s and 1s
  + Ordinal can mapped to numerical values
  + For non-ordinal, one-hot encoding is used.

**Scaling**

* Numerical data can be scaled to ensure proportionate influence on the prediction
* Common Techniques for scaling:
  + Mean/variance standardization
  + MinMax scaling
  + MaxAbs scaling
  + Robust scaling
  + Normalizer

**2.3 Analyze and visualize data for machine learning**

* Visualization helps to better understand the features and their relationships
* Common visualization techniques for numerical features:
  + Density plot
  + Histogram
  + Box plot
* Scatterplot is used to visualize relationship between two variables
* Histograms show the overall behavior of a particular feature
* Plots for features:
  + Scatter
  + Box
  + Histogram
  + Scatter matrix
  + Correlation matrix
  + Heatmap
* Plots for target and prediction variable:
  + Confusion matrix

**Domain 3: Modeling**

**3.1 Frame business problems as machine learning problems**

* Supervised ML:
  + Binary Classification
  + Multi- class classification
  + Regression
* Unsupervised:
  + Clustering
  + Anomaly detection
* Reinforcement learning
* Deep Learning:
  + Perceptron
  + Components of a neuron

**3.2 Select the appropriate model(s) for a given machine learning problem**

**Options for choosing a training algorithm**

* Use Amazon SageMaker built-in algorithms
  1. Supervised Learning
     1. Classification
        1. Binary
           1. Linear learner
           2. XGBoost
        2. Multi-class
           1. XGBoost
           2. KNN
     2. Regression
        1. XGBoost: Set objective hyperparameter to re:linear
        2. KNN: Queries the k closest points to the sample point and returns the average of their feature values as the predicted value
        3. Linear learner: Set predictor\_type hyperparameter to regressor
        4. Factorization machines
     3. Recommendation
        1. Factorization machines
     4. NLP
        1. BLazing Text
        2. Sequence2sequence
        3. Object2Vec
     5. Computer Vision
        1. Image classification
        2. Semantic segmentation
        3. Object detection
  2. Unsupervised Learning
     1. Clustering
        1. K-means
        2. LDA
     2. Topic modeling
        1. LDA
     3. Embeddings
        1. Object2Vec
     4. Anomaly detection
        1. Random cut forest
        2. IP insights
     5. Dimensionality reduction
        1. PCA
* Submit custom code to train a model with a deep learning framework like TensorFlow or Apache MXNet
* Use your own custom algorithm and put the code together as a Docker image
* Subscribe to an algorithm from AWS Marketplace

**3.3 Train machine learning models**

* Modeling training, tuning, and evaluation is an iterative process.
* Split data to ensure a proper division between training and evaluation
  + Training set (80%): To train models to see patterns
  + Validation set (10%): To give an estimate of model performance and/or compare performance across different models.
  + Test set (10%): To evaluate the predictive quality of the trained model
* Cross-validation: compare the performance of multiple models. CV techniques will increase the computational power needed for training
  + K-fold CV is a common validation method
  + Leave-one-out CV for small datasets
  + Stratified K-fold CV for imbalanced data
* Creating a training job in Amazon SageMaker
  + Ingest training data into S3 bucket
  + Amazon SageMaker workflow for training jobs
    - It creates a container/ a docker image
    - It orchestrates ML training
    - Model training on ML computer instances
  + Amazon Elastic Container Registry
  + Build your own containers
  + Amazon P3 instances
  + Components of an ML training job for deep learning

**3.4 Perform hyperparameter optimization**

* Hyperparameters are the settings that can be tuned before training
* Hyperparameters methods
  + Manual
  + Grid Search:
    - For each possible combination of hyperparameters, a model is trained and scored.
    - Thorough but not efficient.
  + Random Search:
    - It uses random combinations of hyperparameters to train and score models
  + Bayesian Search
  + SageMaker automated hyperparameters tuning
* Common components to tune:
  + Momentum
  + Optimizers
  + Activation functions
  + Dropout
  + Learning rate

**3.5 Evaluate machine learning models**

* Evaluation determines how well the model will predict the target on future data
* Classification problems metrics:
  + A confusion matrix is often used, which compares true values to predicted outcomes.It gives a high-level comparison of how predicted classes match up against actual classes
  + Accuracy:
    - It is the ratio of correct predictions to total number of predictions.
    - TF+TPTotal
    - It is less effective when there are lots of true negatives (say, predicting fraud with little or no fraud data)
  + Precision:
    - It is the ratio of True Positives to total positive values.
    - TPTP+FP
    - Precision is a good metric when the cost of false positives are high.
  + Recall:
    - It is the ratio of True Positives to True Positives and False Negatives.
    - TPTP+FN
    - Recall is a good metric when the cost of false negatives are high.
  + F1 Score = 2.Precision.RecallPrecision+Recall
  + Area under the curve: Receiver operator curve (AUC-ROC)
* Regression problems metrics:
  + Sum of squared errors, RMSE
* Sensitivity
* Specificity
* Neural network functions like Softmax for the last layer.

**Domain 4: Machine Learning Implementation and Operations**

**4.1 Build machine learning solutions for performance, availability, scalability, resiliency, and fault tolerance**

* One method of achieving high availability and fault tolerance is loose coupling
* Queues are used in loose coupling to pass messages between components.
* Amazon CloudWatch helps to monitor the system.
  + Amazon CloudWatch Logs: enables to monitor, store, and access your log files from EC2 instances.
  + Amazon CloudWatch Events: delivers a near real-time stream of system events that describe change in AWS resources.
  + Amazon CloudWatch alarms: allows to watch CloudWatch metrics and to receive notifications when the metrics fall outside of the levels (high or low thresholds) that is configured.
* AWS CloudTrail captures API calls and related events made by or on behalf of the AWS account and delivers the log files to a specified Amazon S3 bucket.
* Machine Learning and an ETL process have different compute and memory requirements.
* Amazon SageMaker EndPoints:
  + It ensure a highly available machine learning serving, When model is ready to be productionized.
  + It basically a load balancer
* Amazon SageMaker make it easy to run and scale containerized machine learning models
* AWS Auto Scaling
  + It is used to build scalable solutions by configuring automatic scaling.
  + Automatic scaling helps avoid errors and ensure high availability and lower latency.
  + Automatically scaling the endpoint avoids request errors and improves prediction throughput.
  + Use automatic scaling to ensure high availability during traffic fluctuations without having to constantly provision for peak traffic.
  + To determine the scaling policy for automatic scaling in Amazon SageMaker, test for how much load (RPS) the endpoint can sustain

**4.2 Recommend and implement the appropriate machine learning services and features for a given problem**

* For data ingestion:
  + Amazon Kinesis for streams
  + AWS Glue for batches
  + Amazon EMR for Hadoop
* For model building, training, tuning, and evaluation:
  + Amazon SageMaker
* Bottom tier of the stack:
  + It is for expert ML practitioners working at the framework level
  + Infrastructure:
    - EC2 P3 & P3DN
    - EC2 C5 & EC2 G4
    - FPGAs
    - DL containers & AMIs
    - DL containers & AMIs
    - Elastic Kubernetes Service
    - Greengrass
    - Elastic Inference
    - Inferentia
  + ML Frameworks:
    - TensorFlow
    - MXNet
    - PyTorch
  + Interfaces
    - Gluon
    - Keras
* Middle tier of the stack:
  + It is focused on Amazon SageMaker and making the pipeline easier
  + Amazon SageMaker:
    - Ground Truth
    - Notebooks
    - Algorithms + Marketplace
    - Reinforcement Learning
    - Training
    - Optimization
    - Deployment
    - Hosting
* Top tier of the stack
  + It includes services that abstract away the need to build and train ML models.
    - Vision:
      * Rekognition Image
      * Rekognition Video
      * Textract
    - Language:
      * Translate
      * Comprehend and Comprehend medical
    - Speech:
      * Polly
      * Transcribe
    - Recommendations
      * Personalize
    - Chatbots:
      * Amazon Lex
    - Forecasting:
      * Forecast

**4.3 Apply basic AWS security practices to machine learning solutions**

* Security is intrinsically built into SageMaker
* Authentication
  + IAM federation
* Authorization
  + Restrict access by IAM policy and condition keys
* Audit
  + API logs to AWS CloudTrial exception of InvokeEndpoint
* Data protection at rest
  + AWS KMS-based encryption for:
    - Notebooks
    - Training jobs
    - Amazon S3 location to store models
    - Endpoint
* Data protection at motion
  + HTTPs for:
    - API/console
    - Notebooks
    - VPC-enabled
    - Interface endpoint
    - Limit by IP
    - Training jobs/endpoints
* Compliance programs
  + PCI DSS
  + HIPAA-eligible with BAA
  + ISO

**4.4 Deploy and operationalize machine learning solutions**

* This ecosystem must be managed using cloud and software engineering practices
  + End-to-end and A/B testing
  + API versioning, if multiple versions of the model are used
  + Reliability and failover
  + Ongoing maintenance
  + Cloud infrastructure best practices, such as continuous integration/continuous deployment (Cl/CD)
* When an ML model is delivered into a production environment, keep the following in mind:
  + Apply software engineering disciplines
  + Add error recovery code
  + Perform unit testing, quality assurance, and user acceptance testing
  + Use common devops tools like AWS CodeBuild and AWS CodeCommit to automate the system
  + Track, identify, and account for changes in data sources
  + Perform ongoing monitoring and evaluation of results
  + Create methods to collect data from production inferences that can be used to improve future models
* Amazon SageMaker supports automatic scaling for production variants
* Define and apply a scaling policy that uses Amazon CloudWatch metrics
  + Automatic scaling uses the policy to adjust the number of instances up or down in response to actual workloads
  + You can use the AWS Management Console to apply a scaling policy based on a predefined metric
  + A predefined metric is defined in an enumeration so you can specify it by name in code or use it in the console
  + Always load-test your automatic scaling configuration to ensure that it works correctly before using it to manage production traffic

**References**

* [RE:Invent online session](https://virtual.awsevents.com/media/t/1_t19ybu7y)
* [Exam Guide PDF](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS-Certified-Machine-Learning-Specialty_Exam-Guide.pdf)