notes.md

Introduction

About the exam

Recommended

- At least an Associate certification or equivalent experience
- 2 years of hands-on experience in AWS
- Some experience with general machine learning topics and algorithms

Exam:

- 180 minutes, ~65 questions
- Multiple choirce and multiple-resonse
- No partial credit
- No points for unanswered questions
- Scores between 100 and 1000 with min passing score 750
- Scaled scoring models used

Domains:

1. Data Engineering - 20%
2. Exploratory Data Analysis - 24%
3. Modeling - 36%
4. Maching Learning Implementation and Operations - 20%

AWS will try to trick you to expose gaps in your knowledge.

About Course

- Deep knowledge of Sagemaker, Glue, and Kinesis

- Focus on service capabilities and what problems they solve, don't memorize service limits, prices, version numbers.

- Study to become and Expert, don't study to pass the Exam

- Make it your own

- Review the Supplemental Material

Data Collection

- Machine Learning Cycle - Fetch -> Clean -> Prepare -> Train Model -> Evaluate Model -> Deploy to Production -> Monitor & Evaluate -> Fetch ...

- Goal

    - Understand the problem at hand.
    - Understand the parts of our input data

    - Map our data into AWS

Data Collection Concepts

Before you begin, ask a few questions:

- What type of generalization are we seeking?
    - Are we trying to forecast a number? See if the customer will pick option A or B?
- Do we really need machine learning?
    - Could it just be done with if/then logic?
- How will my ML generalization be consumed?
    - Will we need to return results in real time, or just in some batch process?
    - Will others need to talk to this using an API?
- What do we have to work with?
    - What kind of data is out there/in house that we can use with our machine learning process?
- How can I tell if the generalization is working?

Traits of Good Data	Traits of bad data	Why
Large datasets	Small datasets (less than 100 rows).	Generally, more data means better model training
Precise attribute types, feature rich	Useless attributes, not needed for solving problem at hand.	Models need to train on importan features
Complete fields, no missing values	Missing values, null fields.	Modes can skewresults when data points are missing.
Values are consistent.	Inconsistent values	Models like clean and consistent data
Solid distribution of outcomes	Lots of positive outcome, few negative outcomes	Models cannot learn with skewed distribution of outcomes
Fair sampling	Biased sampling	Models will skew results with biased data

You should have at least 10 times as many data points as the total number of features.

No matter how you get your data, you are building a data repository. We must find a way to congregate the data into a single data repository.

General Data Terminology

Data is an integral part of machine learning. Understanding the terminology allows us to define and describe our data.

Datasets - The data we use in machine learning is usually defined as a dataset. Datasets are a collection of data.

- dataset = input data = training/testing data
- column = attribute = feature
- row = observation = sample = data point

Structured Data - has a defined schema and a schema is the information needed to interpret the data, including attribute names and their assigned data types.

Unstructured Data - has no defined schema or structural properties. Makes up majority of data collected.

Semi-structured Data - too unstructured for relational data, but has some organizational structure. Usually in the form of csv, json, or xml.

Database

- Traditional relational databases
- Transactional
- Strict defined schema

Data Warehouse

- Processing done on import (schema-on-write)
- Data is classified/stored with user in mind
- Ready to use with BI tools (query and analysis)

Data Lakes

- Processing done on export (schema-on-read)
- Many different sources and formats
- Raw data may not be ready for use

Machine Learning Data Terminology

Labeled Data - data where we already know what the target attribute is

Unlabeled Data - data that has been collected with no target attribute

Supervised learning is done with labeled data, unsupervised learning is done with unlabeled data.

Categorical features

- Values that are associated with a group
- Qualitative
- Discrete

Continuous features

- Values that are expressed as a measurable number
- Quantitative
- Infinite

Text Data (Corpus Data) - datasets collected from text. Used in Natural Language Processing (NLP), speech recognition, text to speech, and more.

Ground Truth - refers to factual data that has been observed or measured. This data has successfully been labeled and can be trusted as "truth" data.

Amazon SageMaker Ground Truth

- Tool that helps build ground truth datasets by allowing different types of tagging/labeling processes.
- Easily create labeled data.

Image Data - datasets with tagged images

Helpful image datasets:

- MNIST data - a collection of tagged handwritten characters to aid with handwriting analysis problems
- Image Net - a collection of tagged, searchable images to aid with image classification problems

Time Series Data - datasets that capture changes over time

Dataset Type	Example Cases	Format
Image Data	Facial recognition, action recognition, object detection, handwriting and character recognition	images, videos
Text Data	Reviews, news articles, messages, twitter and tweets, dialogs	text, csv
Sound Data	Speech, music, other sounds	mp3, text
Signal Data	Electrical signals, motion-tracking, chemical compounds	text
Physical Data	High-energy physics systems, astronomy, earth science	text
Biological Data	Human, animal, plants, microbes	text
Multi-variable Data	Financial, weather, census, transit, internet, games	csv, text

AWS Data Stores

Amazon Simple Storage Service (S3)

- Unlimited data storage that provides object based storage for any type of data
- Go to place for storing Machine Learning data
- Files can be from 0 bytes to 5 TB
- There is unlimited storage
- Files are stored into buckets (similar to folders)
- S3 is a universal namespace. That is, names must be unique globally
- Upload data through the console or cli/sdk

Relational Database Service (RDS) - SQL store for relational datasets

DynamoDB - NoSQL store for nonrelational datasets

Redshift - Data warehousing solution

Timestream - Allows BI acess to time series data

DocumentDB - somewhere to migrate mongodb data

AWS Migration Tools

Data Pipeline - Used to transfer data from other services to S3, also can be used as a transformation tool

DMS - Used to migrate data between different database platforms

AWS Glue - Fully Managed ETL (Extract, Transform, and Load) service

Datasource	Migration Tool	Why
PostgreSQL RDS instance with training data	AWS Data Pipeline	Specify SqlActivity query and places the output into S3
Unstructured log files in S3	AWS Glue	Create custom classifier and output results into S3
Clustered Redshift data	AWS Data Pipeline or AWS Glue	Use the unload command to return results of a query to CSV file in S3 or Create data catalog describing data and load it into S3

Datasource	Migration Tool	Why
On-premise MySQL instance with training data	AWS Database Migration Service	DMS can load data in CSV format onto S3

AWS Helper Tools

EMR - Distributed workloads over many EC2 systems. Hadoop cluster to process, transform, and analyze large amounts of data. Store petabytes of data in various platforms

Athena - Serverless SQL queries in S3 data

Redshift Spectrum

- Query S3 data
- Must have Redshift Cluster
- Made for existing customers

Athena

- Query S3 data
- No need for Redshift cluster
- New customers quickly want to query S3 data

Exam Tips

Before you Begin

- Understand that before we gather input data we must formulate the problem we are trying to solve
- Know how we can measure success and what your goals are
- Determine if Machine Learning is even necessary
- Understand what type of data is available to help solve problem.

Good Data

- Understand what makes up "good" data and why having good data is important
- Understanding what "good" and "bad" data looks like

Data Terminology

- Know how to identify columns/attributes and rows/observations within a dataset
- Know the difference in structured, semi-structured, and unstructured data
- Know the different types of data repositories (databases, data warehouses, data lakes)
- Understand the differences between labeled data and unlabeled data
- Be able to recognize categorical features and continuous features
- Know terms like corpus, ground truth, time series data, and image data

AWS Data Stores Tools

- Know the different AWS services where data can be stored
- Know what types of data is stored in different AWS services

AWS Migration Tools

- Know the Different AWS services we can use to migrate data

- Know when to use one migration tool over another

AWS Helper tools

- Know what EMR is and how we could use it as a migration tool
- Know what Amazon Athena is and how it differs from Redshift Spectrum.

Streaming Data Collection

Streaming Data Collection Concepts

Static Data - Data collected and stored

Free Dataset Resources

- Kaggle - large quantity of free datasets
- UCI - many OS datasets
- OpenData on AWS
- Google BigQuery

Streaming Data - data streamed in realtime

Kinesis tools used to handle streaming data

- Kinesis Data Streams
- Kinesis Data Firehose
- Kinesis Video Streams
- Kinesis Data Analytics

Kinesis Data Streams

Gets data from producers, the things that produce the data we want in AWS

Can use Kinesis Streams to get the streaming data into AWS using shards (container that holds data we want in aws)

Then use consumers to process that data and store in some storage location

Shard Components

- Partition Key - unique for each shard
- Sequence - each time we make a request, it creates a sequence associated with a shard
- Data

Shard Notes:

- Each shard consists of a sequence of data records. These can be ingested at 1000 records per second
- Default limit of 500 shards, but you can request to unlimited shards.
- A data record is the unit of data captured
    - sequence number
    - partition key
    - data blob (your payload, up to 1 MB)
- Transient Data Store - retention period for the data records are 24 hours to 7 days.

Interacting with Kinesis Data Streams

1. Kinesis Producer Library (KPL) - Easy to use library that allows you to write to a Kinesis Data Stream
2. Kinesis Client Libray (KCL) - Integrated directly with KPL for consumer applications to consume and process data from Kinesis Data Stream
3. Kinesis API (AWS SDK) - Used for low level API operations to send records to a Kinesis Data Stream

Kinesis Producer Library (KPL)

- Provides a layer of abstraction specifically for ingesting data
- Automatic and conrfigurable retry mechanism
- Additional processing delay can occur for higher packing efficiencies and better performance
- Java wrapper

Kinesis API

- Low-level API calls (PutRecords and GetRecords)
- Stream creations, resharding, and putting and getting records are manually handled
- No delays in processing
- Any AWS SDK

When should you use Kinesis Data Streams?

- Needs to be processed by consumers
- Real time analytics
- Feed into other services in real time
- Some action needs to occur on your data
- Storing data is optional
- Data retention is important

Use Cases

- Process and evaluate logs immediately
  - Example: Analyze system and application logs continuously and process within seconds.
- Real-time data analytics
  - Example: Run real-time analytics on click stream data and process it within seconds.

Kinesis Data Firehose

Data comes from Data Producers, can be preprocessed using AWS Lambda, and is sent to some storage solution (Redshift, S3, ...)

Used to stream data directly to a final data storage location / destination

When should you use Kinesis Data Firehose?

- Easily collect streaming data
- Processing is optional
- Final destination is S3 (or other data store)
- Data retention is not important

Use Cases

- Stream and store data from devices
  - Example: Capturing important data from IoT devices, embedded systems, consumer applications and storing it into a data lake
- Create ETL jobs on streaming data

    - Example: Running ETL jobs on streaming data before data is stored into a data warehousing solution

Kinesis Video Streams

Allows us to stream videos into the AWS cloud/images/audio files in real time.

Producers like webcams, microphones, live video feeds, etc. Consumers process data in fragments and frames from KVS to view, process and analyze it Then store in S3 if desired

When should you use Kinesis Video Streams?

- Needs to process real-time streaming video data (audio, images, radar)
- Batch-process and store streaming video
- Feed streaming data into other AWS services

Kinesis Data Analytics

Allows you to continuously read and process streaming data in real time using SQL queries. Gets input from Kinesis Data Streams/Kinesis Data Firehose, run real time SQL queries, and output the results in Redshift, S3, visualization/BI tools

When should you use Kinesis Data Analytics?

- Run SQL queries on streaming data
- Construct applications that provide insight on your data
- Create metrics, dashboads, monitoring, notifications, and alarms
- Output query results into S3 (or other AWS datasources)

Use Cases

- Responsive real-time analytics
    - Example: Send real-time alarms or notifications when certain metrics reach predefined threshold
- Stream ETL jobs
    - Example: Stream raw sensor data, then clean, enrich, organize, and transform it before it lands into data warehouse or data lake

Task at hand	Which Kinesis service to use?	Why?
Need to stream Apache log files directly from (100) EC2 instances and store them into Redshift	Kinesis Firehose	Firehose is for easily streaming data directly to a final destination. First the data is loaded into S3, then copied into Redshift
Need to stream live video coverage of a sporting event to distribute to customers in near real-time	Kinesis Video Streams	Kinesis Video Streams processes real-time streaming video data (audio, images, radar) and can be fed into other AWS services.
Need to transform real-time streaming data and immediately feed into a custom ML application	Kinesis (Data) Streams	Kinesis Streams allows for streaming hug amounts of data, process/transform it, and then store it or feed into custom applications or other AWS services.
Need to query real-time data, create metric graphs, and store output into S3	Kinesis Analytics	Kinesis Analytics gives you the ability to run SQL queries on streaming data, then store or feed the output into other AWS services

Exam Tips

Loading Data into AWS

- Understand how to get data from public or in house data sets and load it into AWS.
- Know the different ways to upload into S3 by using the console, the S3 API, or the AWS cli

The Kinesis Family

- Know what each service is and how it processes/handles streaming data
- Know what shards are, what a data record is, and the retention period for a shard
- Know the difference in the KPL, KCL, and Kinesis API
- For a given scenario, know which streaming Kinesis service to use.