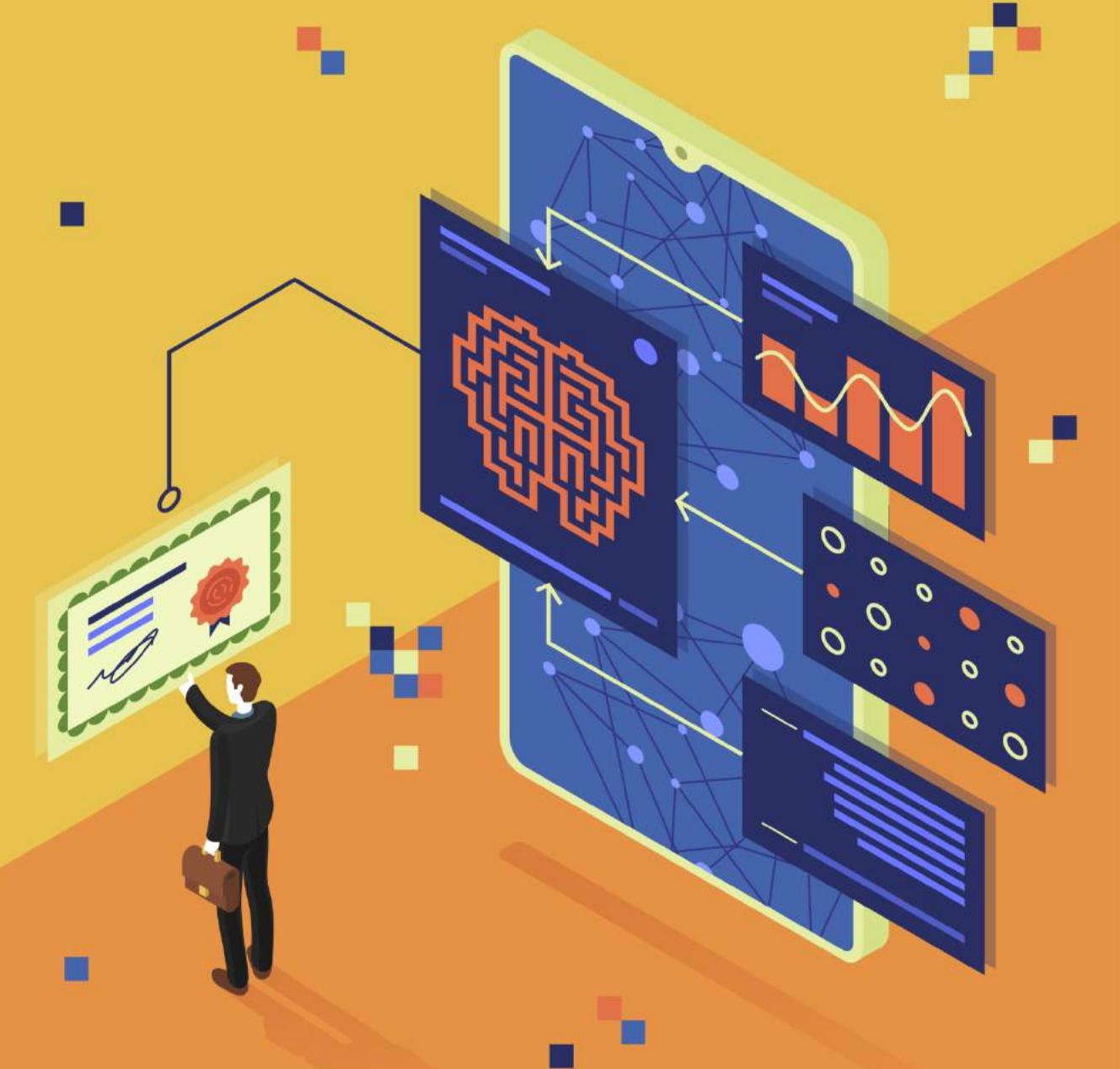


AWS MACHINE LEARNING CERTIFICATION



MODULE #1: DATA ENGINEERING (20% EXAM)



AWS ML CERTIFICATION EXAM DOMAINS



Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #1 OVERVIEW:

SECTION #1: INTRODUCTION, DATA/ML LINGO, AWS DATA STORAGE

- What is Machine Learning and Artificial Intelligence?
- What is Amazon Web Services (AWS)?
- Artificial Intelligence and Machine learning Lingo (data types, Labeled vs. unlabeled, sagemaker groundtruth)
- structured vs. unstructured and database vs. data lake vs. data storage
- AWS Data Storage (Redshift, RDS, S3, DynamoDB)

SECTION #2: AMAZON S3

- Amazon S3 in Depth (partitions, tags)
- Amazon S3 Storage Tiers and Lifecycles
- Amazon S3 Encryption and Security
- Amazon S3 Encryption and Security – Part #2 (ACL, CloudWatch, CloudTrail, VPC)
- Additional Notes (Elasticsearch, ElastiCache, and Database vs. data warehouse)

DOMAIN #1 OVERVIEW:

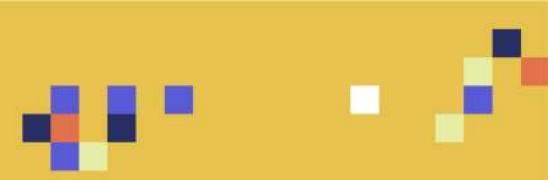
SECTION #3: AWS DATA MIGRATION, GLUE, PIPELINE, STEP AND BATCH

- AWS Glue (crawlers, features, built-in transformations etc)
- AWS Data pipeline
- AWS Data Migration Service (DMS)
- AWS Batch
- Step Function

SECTION #4: DATA STREAMING & KINESIS

- Kinesis Overview
- Kinesis Video Streams
- Kinesis Data Streams
- Kinesis Firehose
- Kinesis Analytics and Random Cut Forest

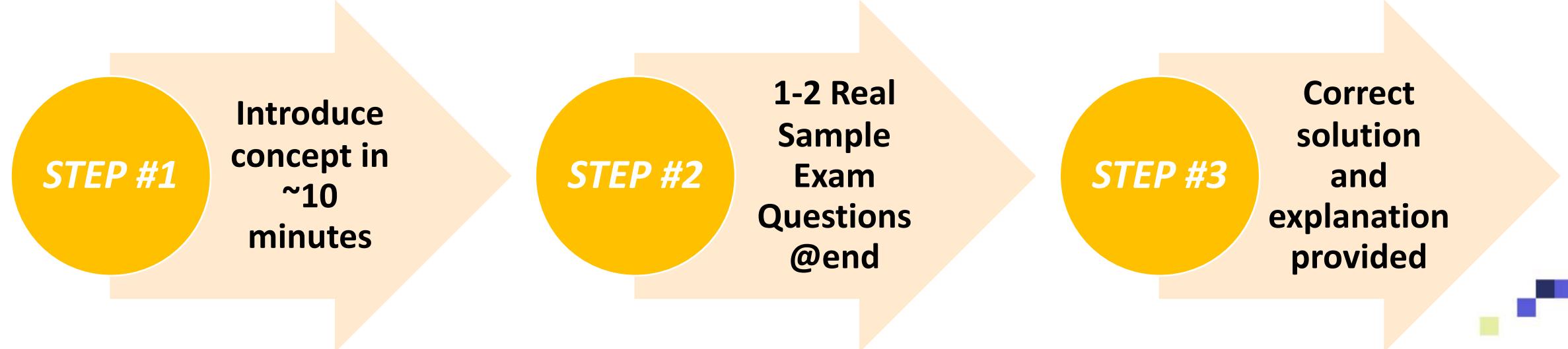
LECTURE DESIGN



- We know how hard it is to study for an exam especially if you have a busy schedule.
- This course is designed to be extremely on point and optimized to pass the exam.

No boring content. Zero unnecessary information.

- Here's the lecture structure that we will follow:

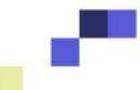


VALUABLE PRIZE!



- For those of you who will successfully complete the entire section and watch the videos till the end, they will receive a valuable prize!

10 NEW SAMPLE EXAM QUESTIONS + COMPLETE ANSWER KEY



GAME AND MINI CHALLENGES!



- Unfortunately, you can't skip the videos.
- You have to collect a code throughout the lectures to unlock the exam.
- Special characters will appear at random moments throughout the video.
- You will need to collect the code and enter it to a website to access the material.
- That's what the final code might look like!

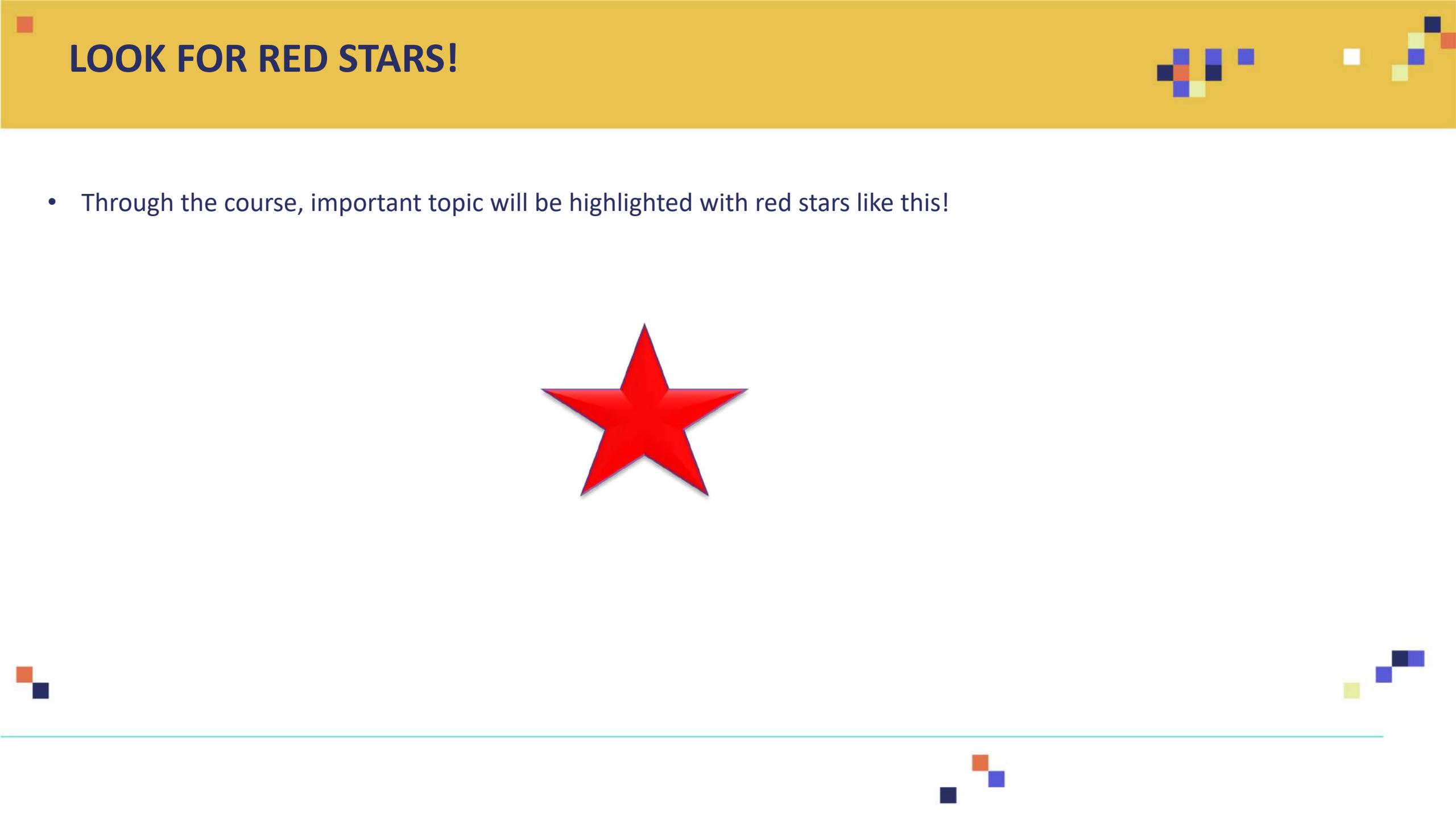
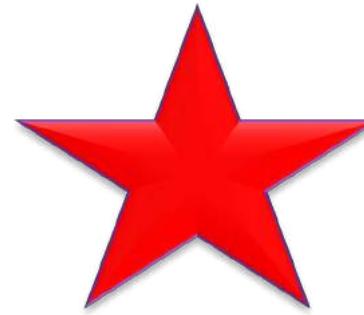
F 2 @ 9 & B



LOOK FOR RED STARS!



- Through the course, important topic will be highlighted with red stars like this!



WHAT IS ARTIFICIAL INTELLIGENCE? AND WHAT IS MACHINE LEARNING? – Part #1



INTRODUCTION



- Artificial Intelligence/Machine learning does not only mean robots or Sci-Fi movies!
- Machine and deep learning applications are everywhere!
- Google search engine, amazon recommender systems, Facebook facial recognition (tagging), Siri.

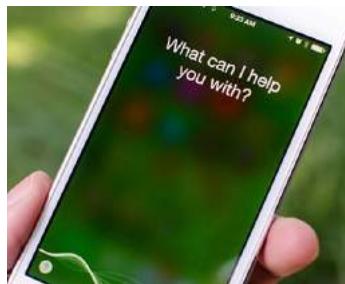
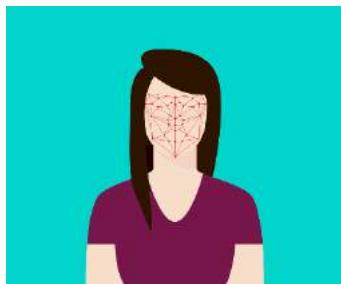


Photo Credit: https://commons.wikimedia.org/wiki/File:Waymo_self-driving_car_front_view.gk.jpg

Photo Credit: <http://blog.etonic.net/index.php?entry=entry110316-081129>

Photo Credit: <https://www.flickr.com/photos/topgold/8325104250>

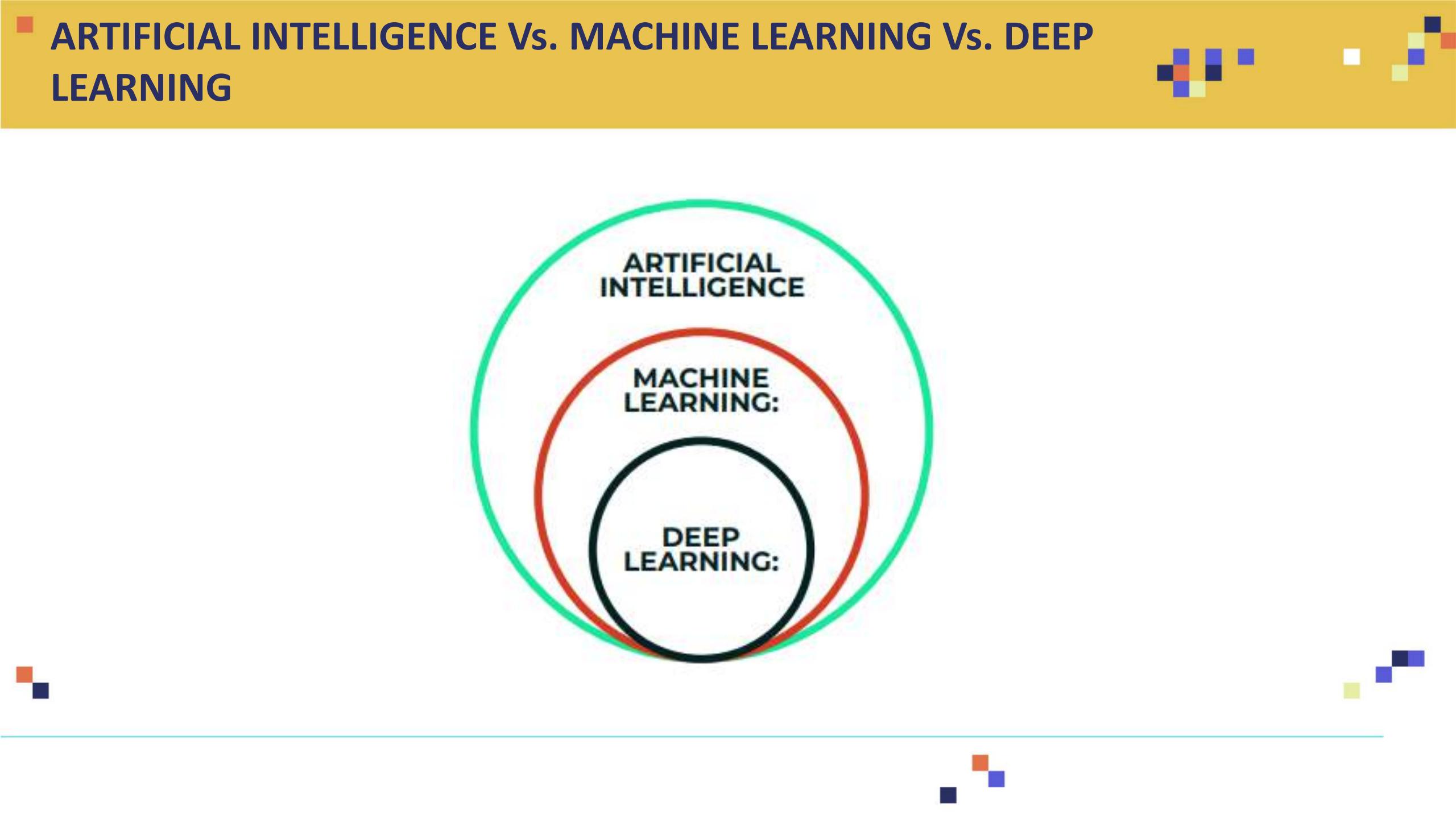
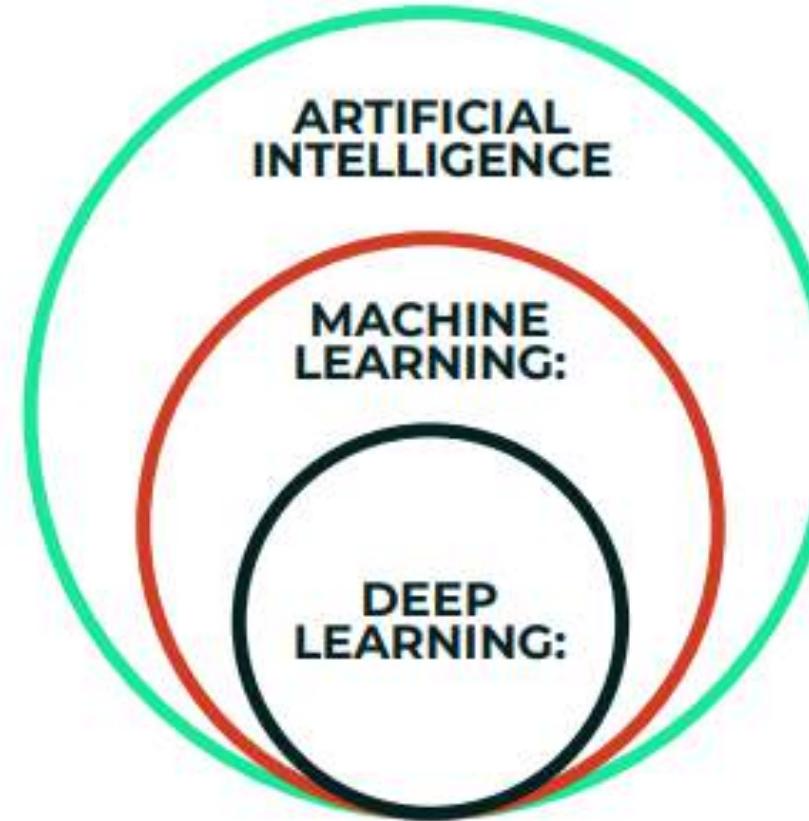
Photo Credit: <https://picryl.com/media/google-search-engine-magnifying-glass-computer-communication-00b825>

Photo Credit: <https://www.flickr.com/photos/iphonedigital/26988770454>

Photo Credit: <https://pixabay.com/illustrations/flat-recognition-facial-face-woman-3252983/>



ARTIFICIAL INTELLIGENCE Vs. MACHINE LEARNING Vs. DEEP LEARNING



1. ARTIFICIAL INTELLIGENCE



- Science that empowers computers to mimic human intelligence such as decision making, text processing, and visual perception.
- AI is a broader field (i.e.: the big umbrella) that contains several subfield such as machine learning, robotics, and computer vision.



Photo Credit: <https://pixabay.com/illustrations/artificial-intelligence-brain-think-4111582/>



2. MACHINE LEARNING



- Machine Learning is a subfield of Artificial Intelligence that enables machines to improve at a given task with experience.
- It is important to note that all machine learning techniques are classified as Artificial Intelligence ones. However, not all Artificial Intelligence could count as Machine Learning since some basic Rule-based engines could be classified as AI but they do not learn from experience therefore they do not belong to the machine learning category.

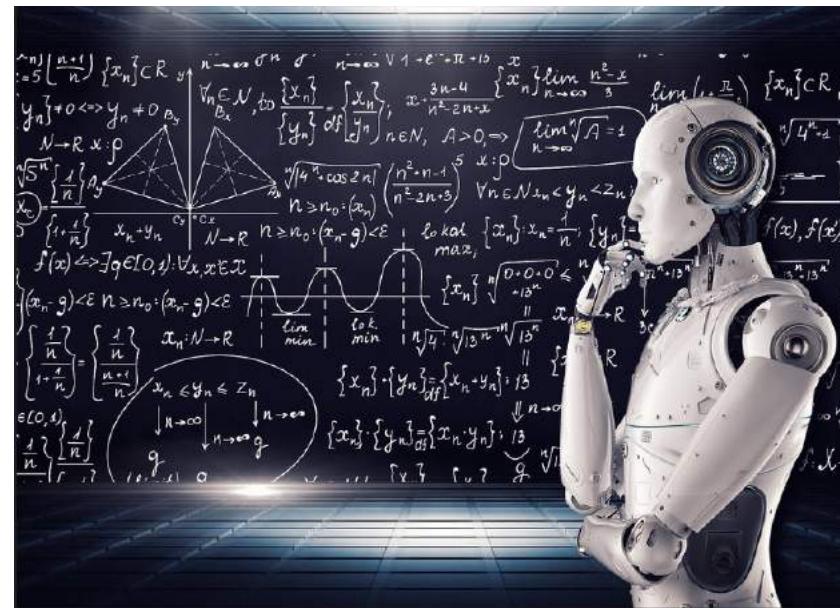


Photo Credit: <https://www.flickr.com/photos/mikemacmarketing/30212411048>



3. DEEP LEARNING

- Deep Learning is a specialized field of Machine Learning that relies on training of Deep Artificial Neural Networks (ANNs) using large dataset such as images.
- ANNs are information processing models inspired by the human brain.
- The human brain consists of billions of neurons that communicate to each other using electrical and chemical signals and enable humans to see, feel, and make decision.

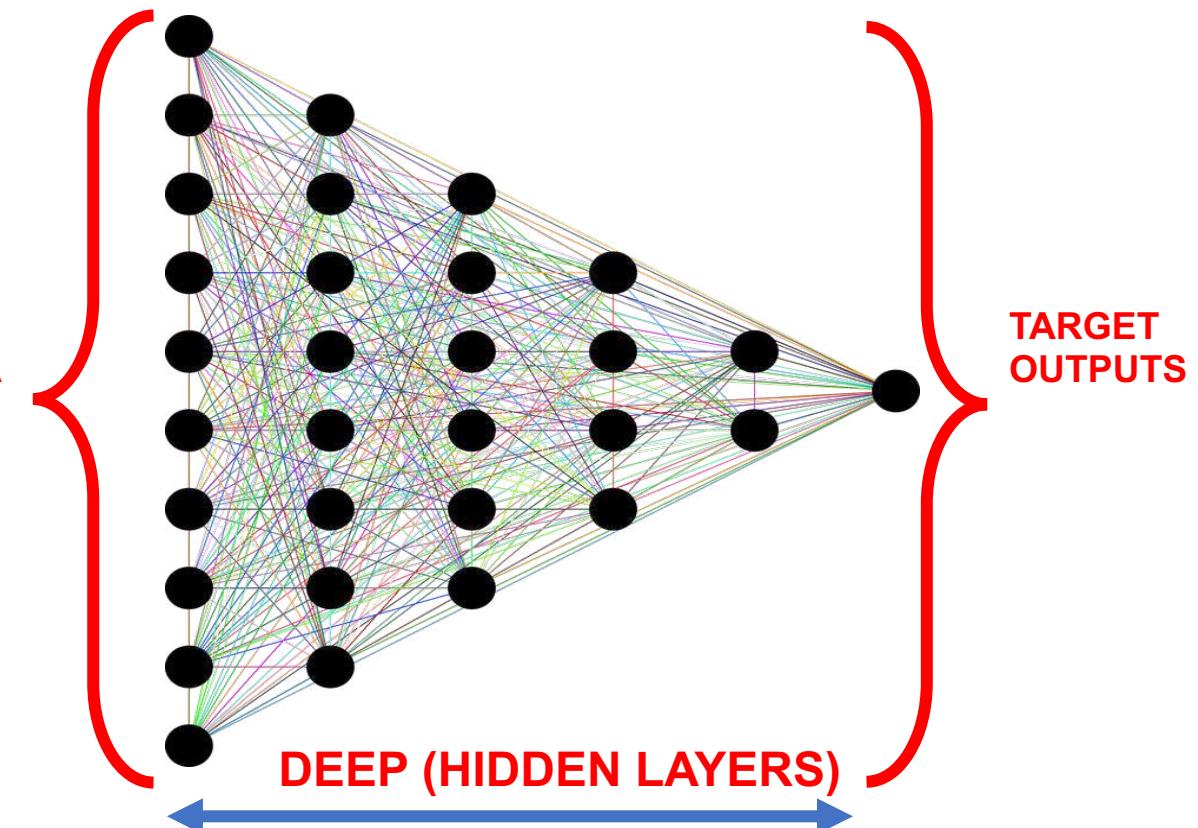
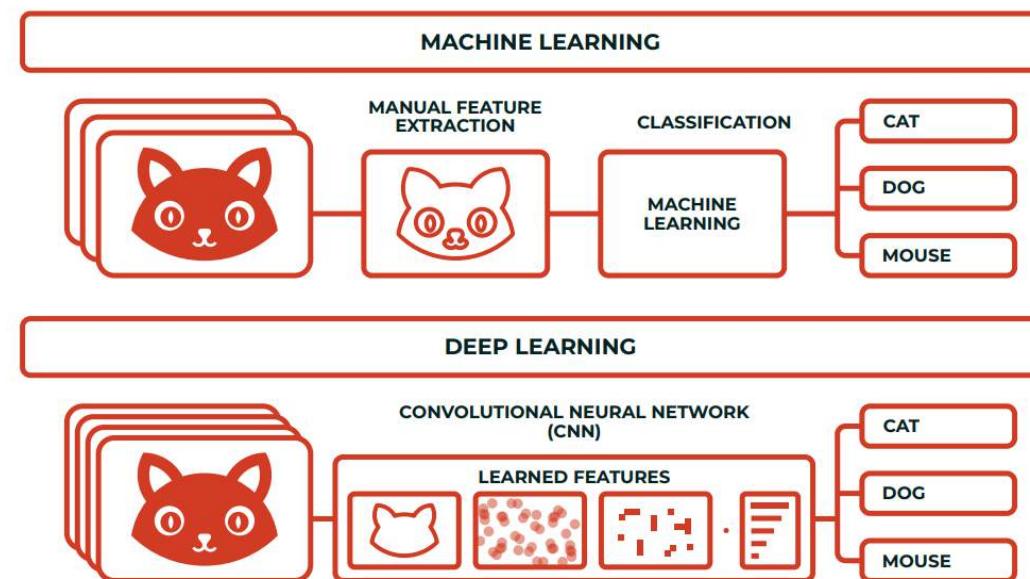


Photo Credit: <https://pixabay.com/en/neural-network-thought-mind-mental-3816319/>

MACHINE VS. DEEP LEARNING



- What differentiates deep learning from machine learning techniques is in their ability to extract features automatically:
 - Machine learning Process: (1) select the model to train, (2) manually perform feature extraction.
 - Deep Learning Process: (1) Select the architecture of the network, (2) features are automatically extracted by feeding in the training data (such as images) along with the target class (label).



WHAT IS ARTIFICIAL INTELLIGENCE? AND WHAT IS MACHINE LEARNING? – Part #2

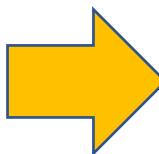


MACHINE LEARNING: BIG PICTURE



ARTIFICIAL INTELLIGENCE
Science that enables computers to mimic human intelligence.
Subfields: Machine Learning, robotics, and computer vision

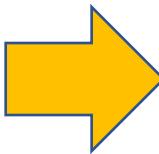
MACHINE LEARNING
Subset of AI that enable machines to improve at tasks with experience



SUPERVISED LEARNING
Training algorithms using labeled input/output data.

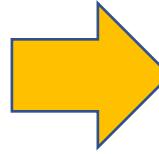


CLASSIFICATION



UNSUPERVISED LEARNING
Training algorithms with no labeled data. It attempts at discovering hidden patterns on its own.

CLUSTERING

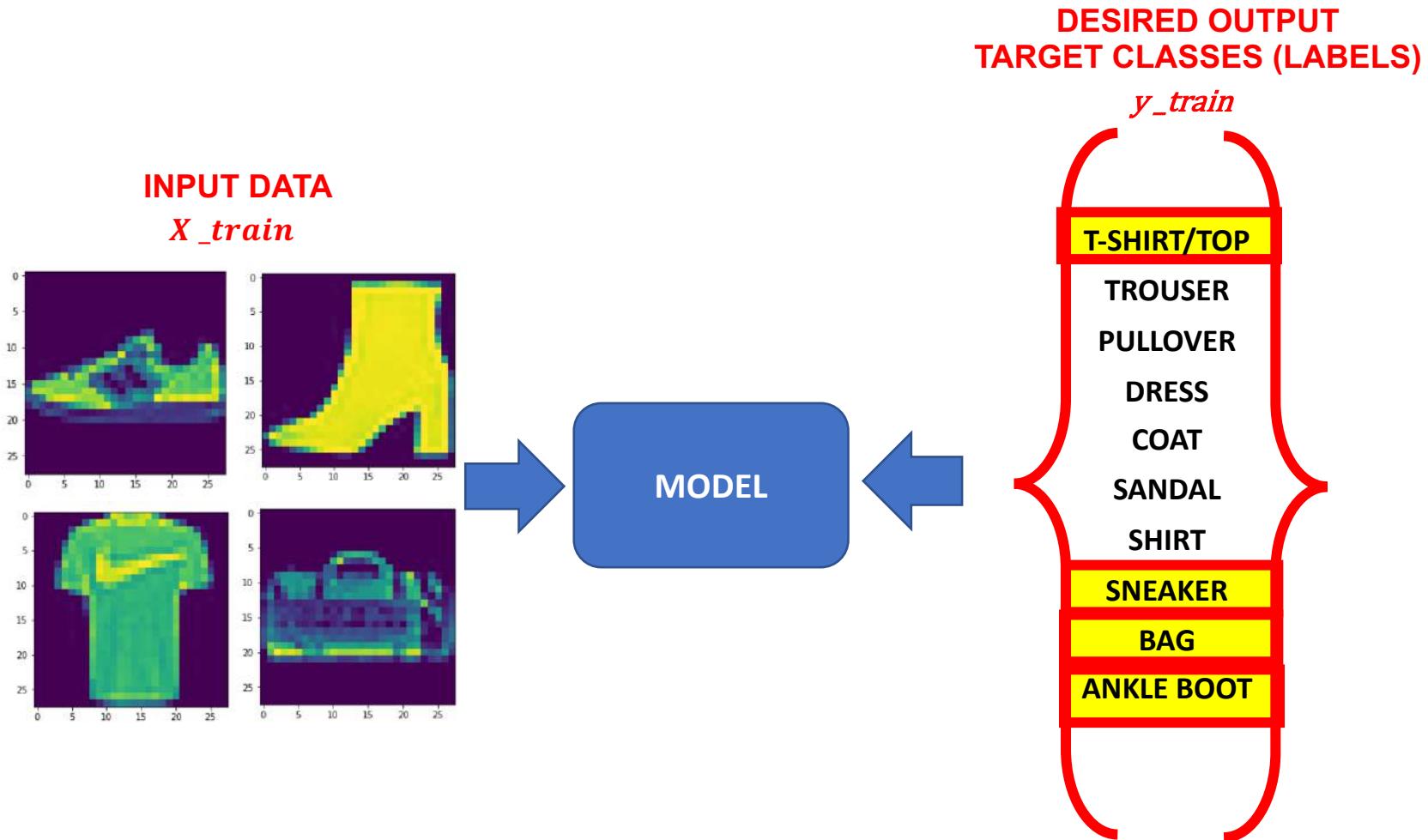


REINFORCEMENT LEARNING
Algorithm take actions to maximize cumulative reward.

MACHINE LEARNING: SUPERVISED LEARNING



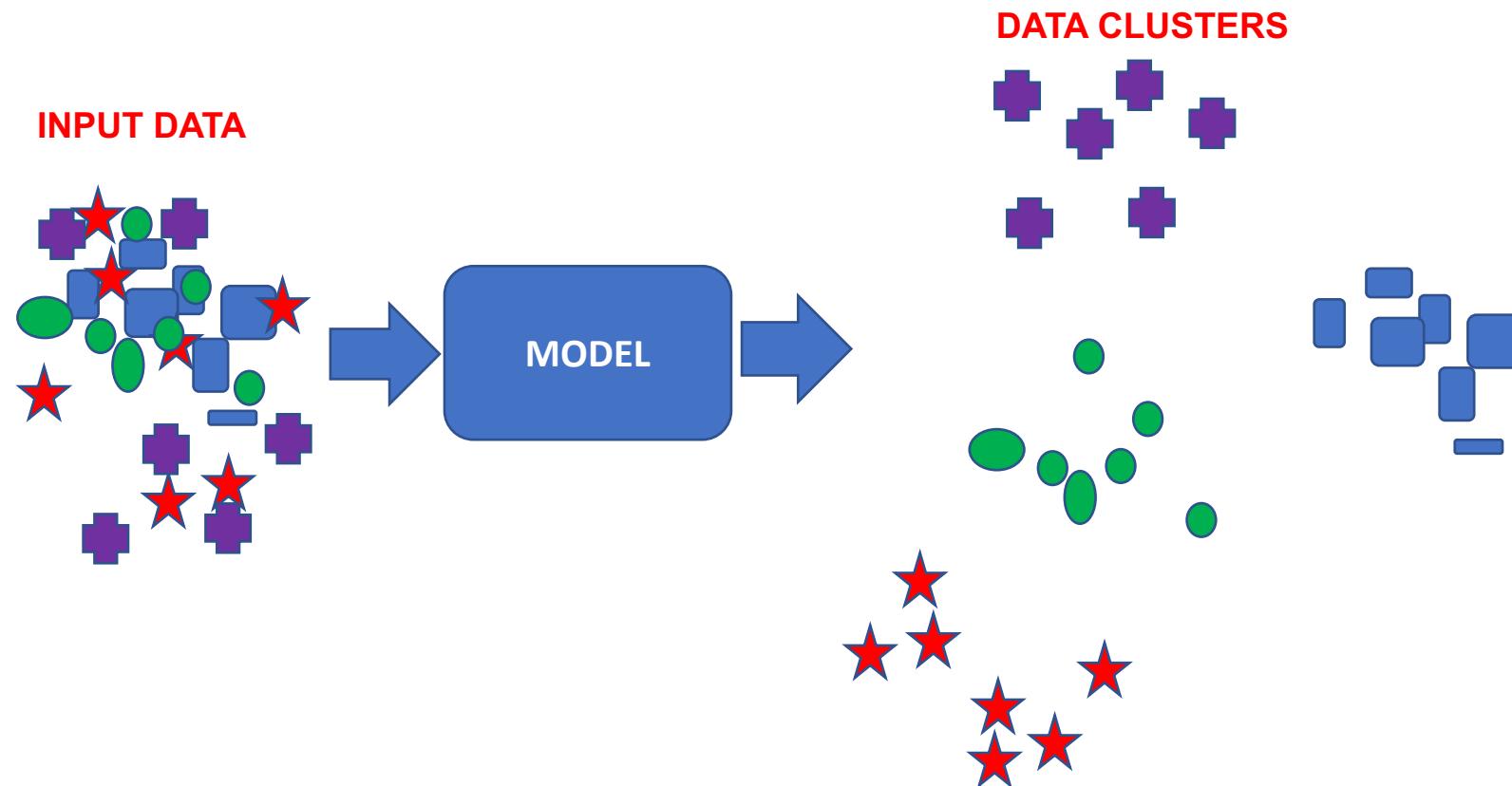
- **Supervised:** used to train algorithms using labeled input and output data.
- Performance is assessed by comparing trained model prediction vs. real output.



MACHINE LEARNING: UNSUPERVISED LEARNING



- **Unsupervised learning:** provides the algorithm with no labeled data.
- The algorithm attempts at discovering hidden patterns within the training data.
- Unsupervised learning methods can analyze complex data that humans might find difficult to interpret.
- No feedback!



MACHINE LEARNING: REINFORCEMENT LEARNING



- Reinforcement learning allows machines take actions to maximize cumulative reward.
- Reinforcement algorithms learn by trial and error through reward and penalty.
- Two elements: environment and learning agent.
- The environment rewards the agent for correct actions.
- Based on the reward or penalty, agent improves its environment knowledge to make better decision.

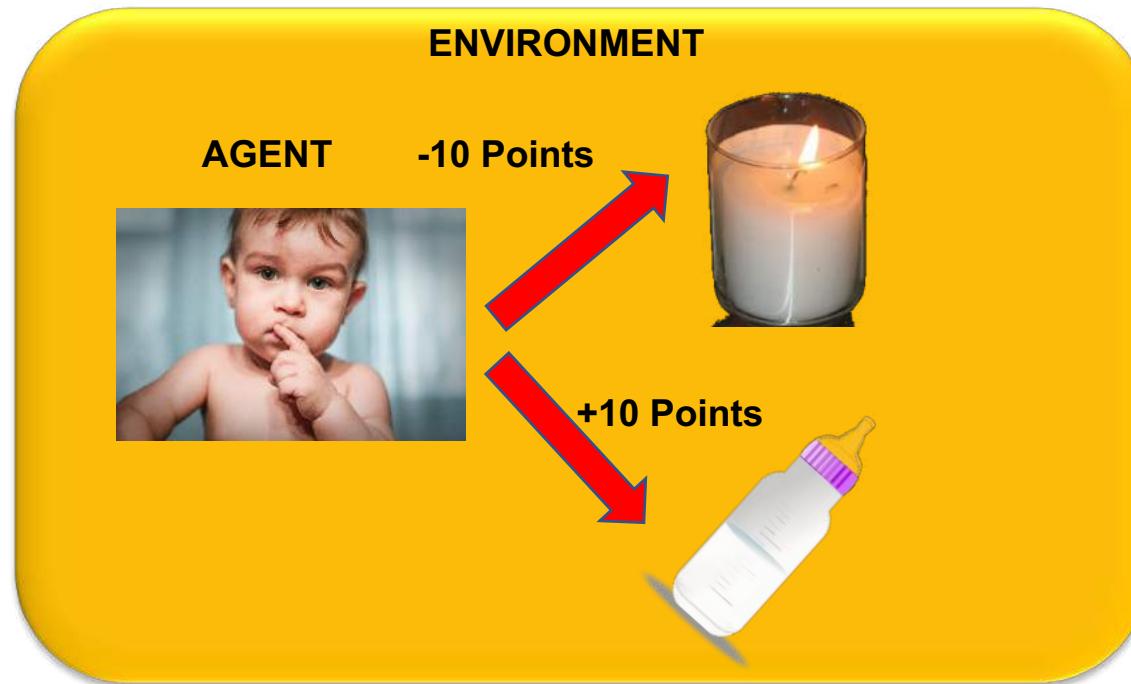
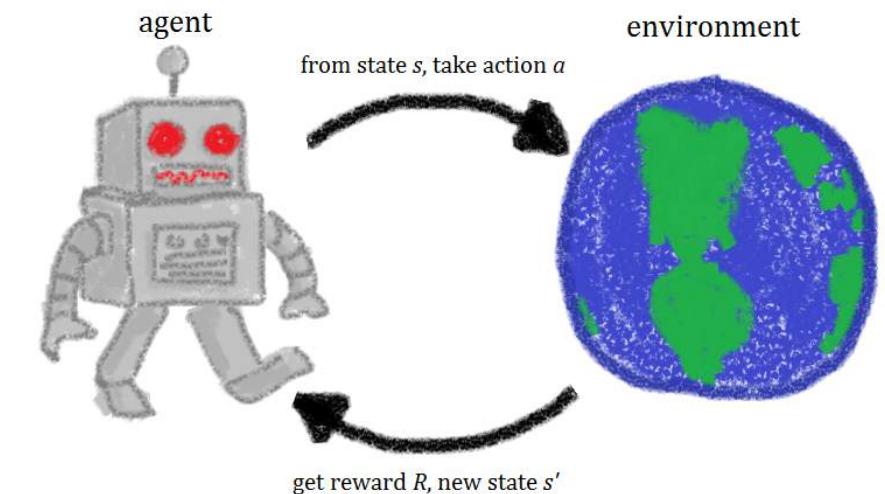


Photo Credit: https://commons.wikimedia.org/wiki/File:RL_agent.png



AMAZON WEB SERVICES (AWS)

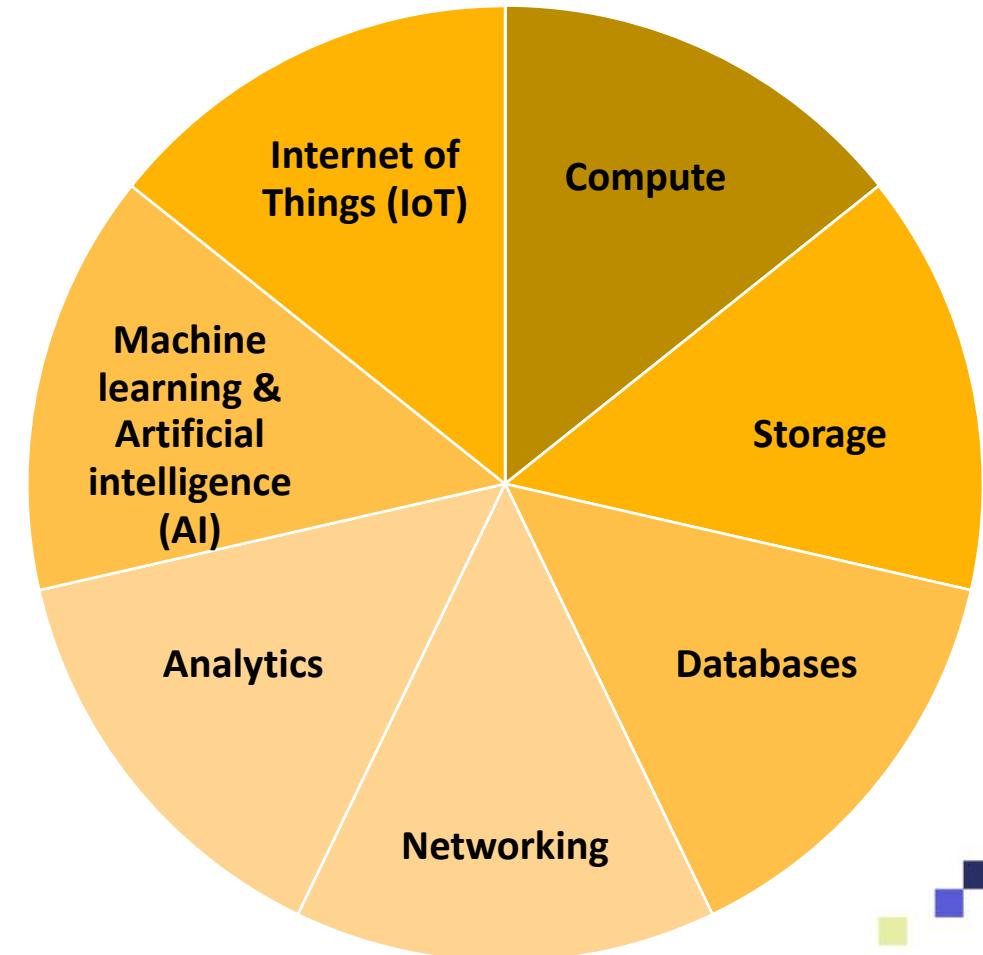


WHAT IS AWS?

- Amazon Web Services (AWS) is the world's top cloud platform.
- AWS offers more than 165 fully featured services (40 of them are not offered anywhere else!).
- AWS is adopted by millions of customers globally including small and large scale enterprises.
- AWS enables companies to be more agile, flexible, secure at a fraction of the cost.
- AWS provides services for broad range of applications such as:



Photo Credit: https://commons.wikimedia.org/wiki/File:AmazonWebservices_Logo.svg



MACHINE LEARNING COMPONENTS?



1. DATA



2. MODEL



3. COMPUTE



AWS SERVICES: STORAGE AND COMPUTE



S RyanAhn

History

Amazon SageMaker

Console Home

Amazon Polly

Amazon Rekognition

Amazon Comprehend

Amazon Translate

Compute

EC2

Lightsail

ECR

ECS

EKS

Lambda

Batch

Elastic Beanstalk

Serverless Application Repository

Storage

S3

EFS

FSx

S3 Glacier

Storage Gateway

AWS Backup

Database

RDS

DynamoDB

ElastiCache

Neptune

Amazon Redshift

Amazon QLDB

Amazon DocumentDB

Analytics

Athena

EMR

CloudSearch

Elasticsearch Service

Kinesis

QuickSight

Data Pipeline

AWS Data Exchange

AWS Glue

AWS Lake Formation

MSK

Customer Enablement

AWS IQ

Support

Managed Services

Blockchain

Amazon Managed Blockchain

Satellite

Ground Station

Management & Governance

AWS Organizations

CloudWatch

AWS Auto Scaling

CloudFormation

CloudTrail

Config

OpsWorks

Service Catalog

Systems Manager

Trusted Advisor

Control Tower

AWS License Manager

AWS Well-Architected Tool

Personal Health Dashboard

AWS Chatbot

Launch Wizard

Business Applications

Alexa for Business

Amazon Chime

WorkMail

End User Computing

WorkSpaces

AppStream 2.0

WorkDocs

WorkLink

Internet Of Things

IoT Core

Amazon FreeRTOS

IoT 1-Click

IoT Analytics

IoT Device Defender

IoT Device Management

IoT Events

IoT Greengrass

IoT SiteWise

IoT Things Graph

Game Development

Amazon GameLift

3

1

The screenshot shows the AWS Services dashboard. A search bar at the top has 's' typed into it. Below the search bar, there's a navigation menu with 'Services' and 'Resource Groups'. On the far right, a user profile for 'RyanAhn' is shown with a notification bell icon. The main content area displays various service categories: Compute, Storage, Analytics, Customer Enablement, Business Applications, End User Computing, Internet Of Things, and Game Development. The 'Compute' and 'Storage' sections are circled in red and labeled with yellow circles containing the numbers '3' and '1' respectively. The 'Compute' section includes services like EC2, Lambda, and ECR. The 'Storage' section includes S3, EFS, and FSx.

AWS SERVICES: MACHINE LEARNING



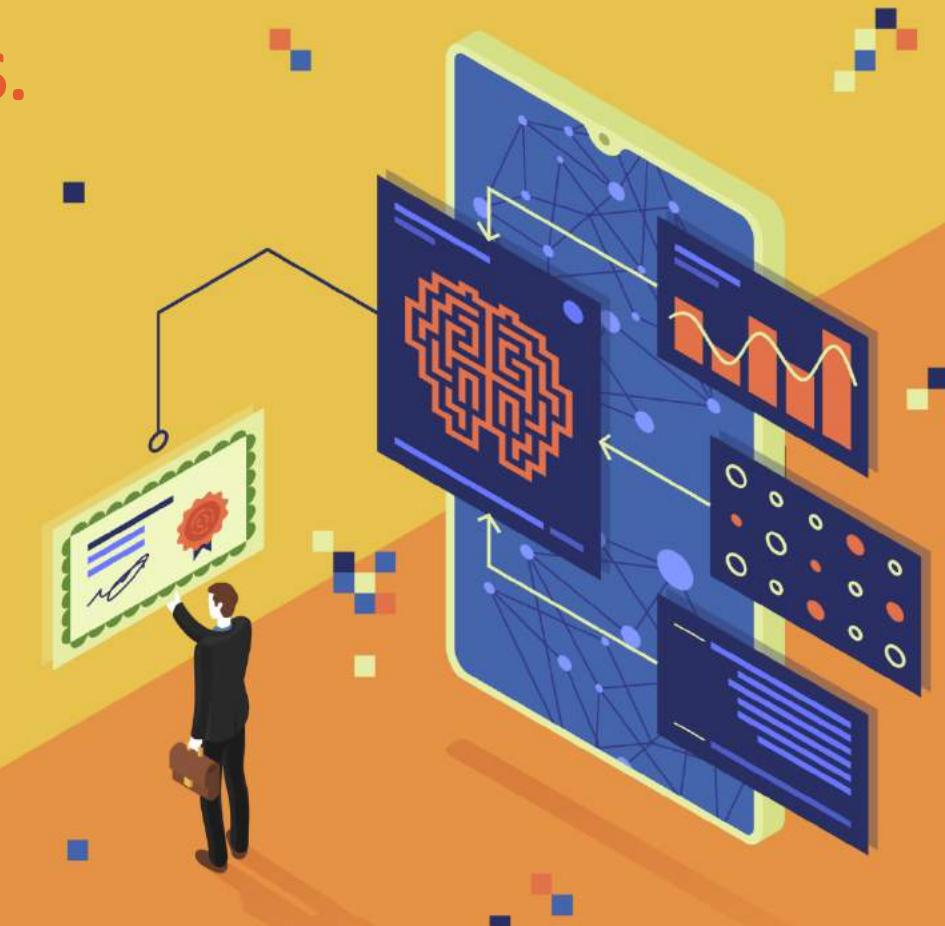
Screenshot of the AWS Services Catalog showing the Machine Learning section highlighted.

The screenshot shows the AWS Services Catalog interface. On the left, there's a sidebar with 'History' and links to various AWS services like S3, Amazon SageMaker, and Amazon Comprehend. The main area lists services under several categories:

- Migration & Transfer**: AWS Migration Hub, Application Discovery Service, Database Migration Service, Server Migration Service, AWS Transfer for SFTP, Snowball, DataSync.
- Networking & Content Delivery**: VPC, CloudFront, Route 53, API Gateway, Direct Connect, AWS App Mesh, AWS Cloud Map, Global Accelerator.
- Developer Tools**: CodeStar, CodeCommit, CodeBuild, CodeDeploy, CodePipeline, Cloud9, X-Ray.
- Robotics**: AWS RoboMaker.
- Machine Learning** (highlighted with a red box):
 - Amazon SageMaker
 - Amazon Comprehend
 - Amazon Forecast
 - Amazon Lex
 - Amazon Machine Learning
 - Amazon Personalize
 - Amazon Polly
 - Amazon Rekognition
 - Amazon Textract
 - Amazon Transcribe
 - Amazon Translate
 - AWS DeepLens
 - AWS DeepRacer
- Media Services**: Elastic Transcoder, Kinesis Video Streams, MediaConnect, MediaConvert, MediaLive, MediaPackage, MediaStore, MediaTailor, Elemental Appliances & Software.
- Mobile**: AWS Amplify, Mobile Hub, AWS AppSync, Device Farm.
- AR & VR**: Amazon Sumerian.
- Application Integration**: Step Functions, Amazon EventBridge, Amazon MQ, Simple Notification Service, Simple Queue Service, SWF.
- AWS Cost Management**: AWS Cost Explorer, AWS Budgets, AWS Marketplace Subscriptions.
- Customer Engagement**: Amazon Connect, Pinpoint, Simple Email Service.

A yellow circle with the number '2' is overlaid on the 'Machine Learning' section.

- **AI/ML DATA LINGO – LABELED VS.
UNLABELED**



MACHINE LEARNING DATA



- Machine Learning models require data to train.
- There are generally two types of data that we could use to train machine learning models.

UNLABELED DATASET

Unlabeled data consists of data that does not have explanation (class or tag) associated with it.



LABELED DATASET

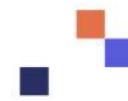
Labeled data consists of unlabeled data but with a “class” or “tag” associated with it.



LABEL = “CAT”



LABEL = “DOG”



WHERE DOES THIS DATA COME FROM?



- Data can come from so many sources such as images, audio, video, and text.
- Collecting, structuring and analysing this data is critical for companies to gain customers insights and set their marketing and product strategies.

IMAGE/VIDEO



TEXT (CORPUS)



AUDIO/SOUND



TIMESERIES/SIGNALS



Photo Credit: <https://pxhere.com/en/photo/1454351>

Photo Credit: <https://www.flickr.com/photos/29881930@N00/2086641598>

Photo Credit: https://commons.wikimedia.org/wiki/File:Mobile_phone_text_messages.jpg

Photo Credit: https://en.wikipedia.org/wiki/File:Messages_Yosemite.svg

Photo Credit: <https://www.pexels.com/photo/blue-and-yellow-graph-on-stock-market-monitor-159888/>



GOOD Vs. BAD DATA



GOOD DATA

- Many samples (large number of data points)
- Not Biased
- Does not contain missing data points
- Only contains (relevant) important features
- Does not contain duplicate samples

BAD DATA

- Few samples (small number of data points)
- Biased
- Contains missing data points
- Contains many irrelevant (useless) features
- Contains duplicate samples



WHERE DOES THIS DATA COME FROM?

- Data could also come from multiple sources such as Kaggle, UCI, AWS Dataset, and ImageNet.
- ImageNet is an open source repository of images consisting of 21,841 subcategories (classes) and over 14 million images.

The image displays three side-by-side screenshots of data repositories:

- Kaggle Datasets:** A screenshot of the Kaggle website showing a search bar and a list of public datasets. Examples include "UFC Fight Historical data from 1993 to 2009", "Forest Fires In Brazil", and "GK Mobile Strategy Games".
- UCI Machine Learning Repository:** A screenshot of the UCI ML Repository homepage. It features a search bar and a table titled "Browse Through: 488 Data Sets". The table includes columns for Name, Data Type, Default Task, Attribute Types, Instances, Attributes, and Year. Some entries shown are "Abalone", "Adult", "Mushrooms", "Lenses", "Artificial Characters", "Auditory/Visual", "Androgen/Strained", "Aortic MEG", "Aneurysm", and "Bacteria".
- Registry of Open Data on AWS:** A screenshot of the AWS Registry of Open Data. It shows a search bar and a table titled "Search datasets (currently 119 matching datasets)". The table has columns for Name, Description, and Details. An example entry is "Amazon Customer Reviews (a.k.a. Product Review)".

Check out website here: <https://archive.ics.uci.edu/ml/datasets.php>
Check out website here: <https://www.kaggle.com/datasets>

HOW TO OBTAIN LABELED DATA USING AWS? SAGEMAKER GROUNDTRUTH

- AWS SageMaker GroundTruth is a service offered by AWS to label data.
- In machine learning terminology, Ground truth means “gold standard”!
- Ground Truth indicates the “true” or “real” class that you would like your model to learn how to predict.



AMAZON
SAGEMAKER
GROUNDTRUTH

The screenshot shows the AWS SageMaker GroundTruth console interface. At the top, there are navigation links for 'Services' and 'Resource Groups'. Below that, a 'Task category' dropdown is set to 'Image'. The main area is titled 'Task selection' and describes the type of data being labeled. It lists four options:

- Image classification:** Get workers to categorize images into specific classes. (Info) Basketball Soccer. Below this is a thumbnail image of a basketball game.
- Bounding box:** Get workers to draw bounding boxes around specified objects in your images. (Info) Below this are two thumbnail images of birds with green bounding boxes drawn around them.
- Semantic segmentation:** Get workers to draw pixel level labels around specific objects and segments in your images. (Info) Below this is a thumbnail image of a woman walking on a street with a blue car.
- Label verification:** Get workers to verify existing labels in your dataset. (Info) Correct label Incorrect label. Below this is a thumbnail image of a green car with a white bounding box around it.

HOW TO OBTAIN LABELED DATA USING AWS? SAGEMAKER GROUNDTRUTH



The screenshot shows the Amazon SageMaker console with the 'Ground Truth' section highlighted. Below, a diagram titled 'How it works' illustrates the machine learning pipeline: Label, Build, Train, Tune, and Deploy. The 'Label' section is specifically highlighted with a red box and a red arrow pointing from the explanatory text below.

YOU CAN CREATE LABELING JOBS USING AMAZON GROUND TRUTH SERVICE

Amazon SageMaker
Build, train, and deploy machine learning models at scale
The quickest and easiest way to get ML models from idea to production.

How it works

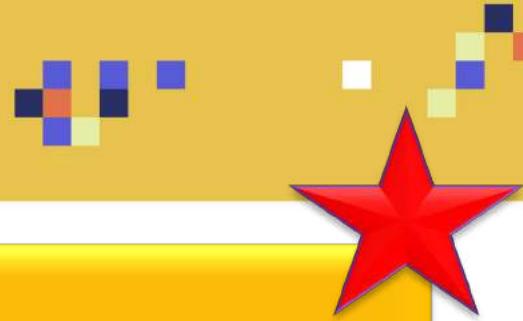
- Label**
Set up and manage labeling jobs for highly accurate training datasets within Amazon SageMaker, using active learning and human labeling
- Build**
Connect to other AWS services and transform data in Amazon SageMaker notebooks
- Train**
Use Amazon SageMaker's algorithms and frameworks, or bring your own, for distributed training
- Tune**
Amazon SageMaker automatically tunes your model by adjusting multiple combinations of algorithm parameters
- Deploy**
Once training is completed, models can be deployed to Amazon SageMaker endpoints, for real-time predictions

<https://aws.amazon.com/sagemaker/groundtruth/>

- **AI/ML DATA LINGO – DATA TYPES**



DATA TYPES



1. QUANTITATIVE (NUMERICAL)

- Numerical data also known as quantitative data represents a measurement or count.
- Examples: weight, blood pressure, and dollars count.
- Either discrete or continuous

2. QUALITATIVE (CATEGORICAL)

- Categorical (Qualitative) data represents data that could be divided into groups.
- Examples: race, sex, age group, and educational level.
- Categorical data can take numerical values but they do not have mathematical meaning so you can't multiply them together for example.

3. ORDINAL

- Ordinal data represents a mix between numerical and categorical data.
- Example: course ratings on Udemy!
- Data consists of categories such as numbers between 1 and 5, in which:
 - 1 star means poor quality course
 - 5 star means great quality course

1. QUANTITATIVE (NUMERICAL) DATA



- Numerical data also known as quantitative data represents a measurement or count.
- Examples: weight, blood pressure, and dollars count.
- Numerical data consists of two types as follows: (1) discrete and (2) continuous.

DISCRETE

- Includes data that are distinct and separable.
- Discrete data could be counted as integers.
- Examples: How many cats do you have? How many products sold?

CONTINUOUS

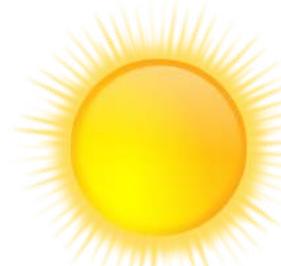
- Represent measurements that are uncountably infinite
- Continuous data can only be described using intervals on the real number line.
- Examples: How much gas did you put in a car? Values could be anywhere between 0 gallons to 20 gallons. There are infinite possibilities; 8.40 gallons, or 8.41, or 8.414863 gallons.



2. QUALITATIVE (CATEGORICAL) DATA



- Categorical data represents data that could be divided into groups.
- Examples: race, sex, age group, and educational level.
- Categorical data can take numerical values but they do not have mathematical meaning so you can't multiply them together for example (and order does not mean anything).
 - **Example:** 1 = single, 2 = married
- Binary data is a type of qualitative data that consists of only two values such as:
 - **Example:** 1 or 0 ON or OFF True or False
- Binary data is very common in machine learning classification algorithms in which the outcome could be one of two options: healthy or sick, malignant or benign.



SUNNY = 1



CLOUDY = 2



RAINY = 3

Photo Credit: <https://publicdomainvectors.org/en/free-clipart/Vector-graphics-of-weather-forecast-color-symbol-for-brightly-sunny-sky/18972.html>

Photo Credit: <https://pixabay.com/vectors/clouds-weather-rain-drops-308682/>

Photo Credit: <https://publicdomainvectors.org/en/free-clipart/Vector-drawing-of-weather-forecast-color-symbol-for-sunny-to-cloudy-sky/18971.html>



3. ORDINAL DATA

- Ordinal data represents a mix between numerical and categorical data.
- Example: course ratings on Udemy!
- Data consists of categories such as numbers between 1 and 5, in which:
 - 1 star means poor quality course
 - 5 star means great quality course
- The numbers in each category have mathematical meaning.
- This what differentiates ordinal data from categorical data.
- For example, if you take the average of the 1000 reviews on Udemy per course, you will end up with an answer that have a meaning.
- This does not work if you have categorical data, you cannot average single and married and get meaningful results.



THERE IS ONE MORE DATA TYPE: USELESS DATA!



- Useless data is a type of data that is discrete and has no relationship whatsoever with the output.
- We usually drop the useless features from the dataset before training the model
- Example include: random customer IDs at a store or a bank

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	Nan	S
1	2	1	Cumings, Mrs. John Bradley (Florence Briggs Th... Heikkinen, Miss. Laina	female	38.0	1	0	PC 17599 STON/O2. 3101282	71.2833 7.9250	C85 NaN	C S
2	3	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	26.0	0	0	113803	53.1000	C123	S
3	4	1	Allen, Mr. William Henry	male	35.0	1	0	373450	8.0500	Nan	S
4	5	0									

USELESS
INFORMATION



DATABASE Vs. DATA LAKE Vs. DATA WAREHOUSE



STRUCTURED Vs. UNSTRUCTURED DATA



- In order to understand the difference between database, data warehouse and data lake, we need to cover the difference between structured and unstructured data.

STRUCTURED

Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
Cumings, Mrs. John Bradley (Florence Briggs Th... Heikkinen, Miss. Laina	female	38.0	1	0	PC 17599 STON/O2. 3101282	71.2833 7.9250	C85 NaN	C S
Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	26.0	0	0	113803	53.1000	C123	S
Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

UNSTRUCTURED

- Unstructured data has no schema or structure.



Photo Credit: <https://pixabay.com/vectors/database-schema-data-tables-schema-1895779/>
<https://pixabay.com/vectors/newspaper-article-journal-headlines-154444/>
<https://www.flickr.com/photos/lolololori/2343992441>

DATABASE



- Databases are typically structured with a defined schema.
- Items are organized as a set of tables with columns and rows.
- Columns include attributes and rows indicate an object or entity.
- Database is designed to be transactional and they are not designed to perform data analytics.

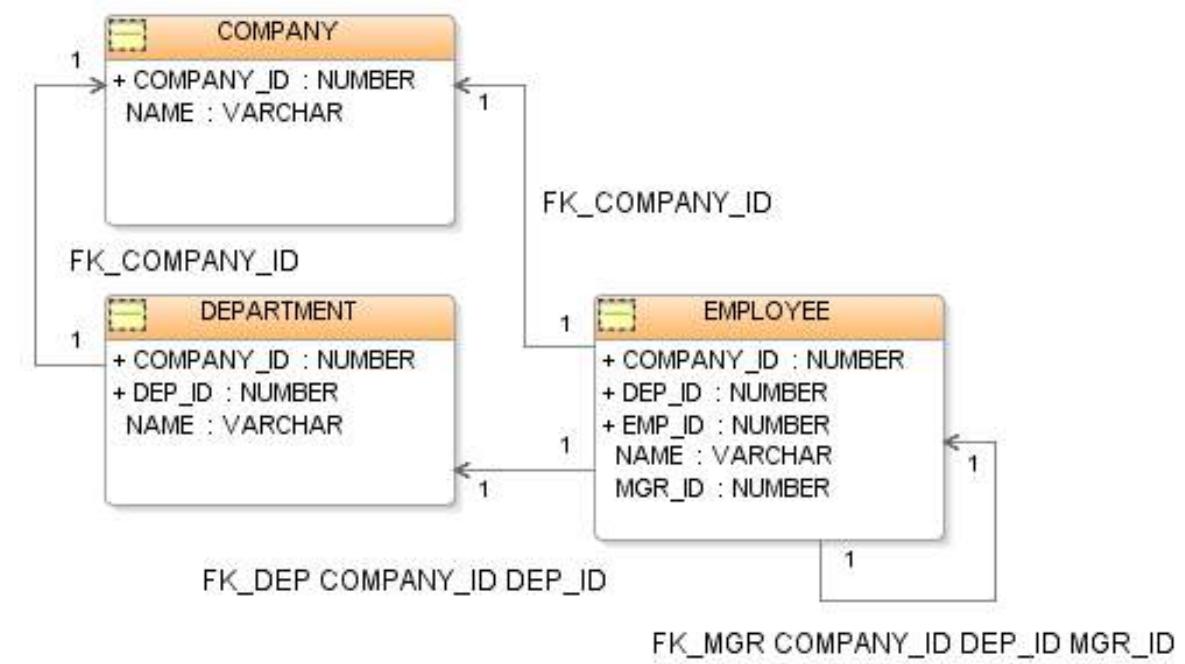
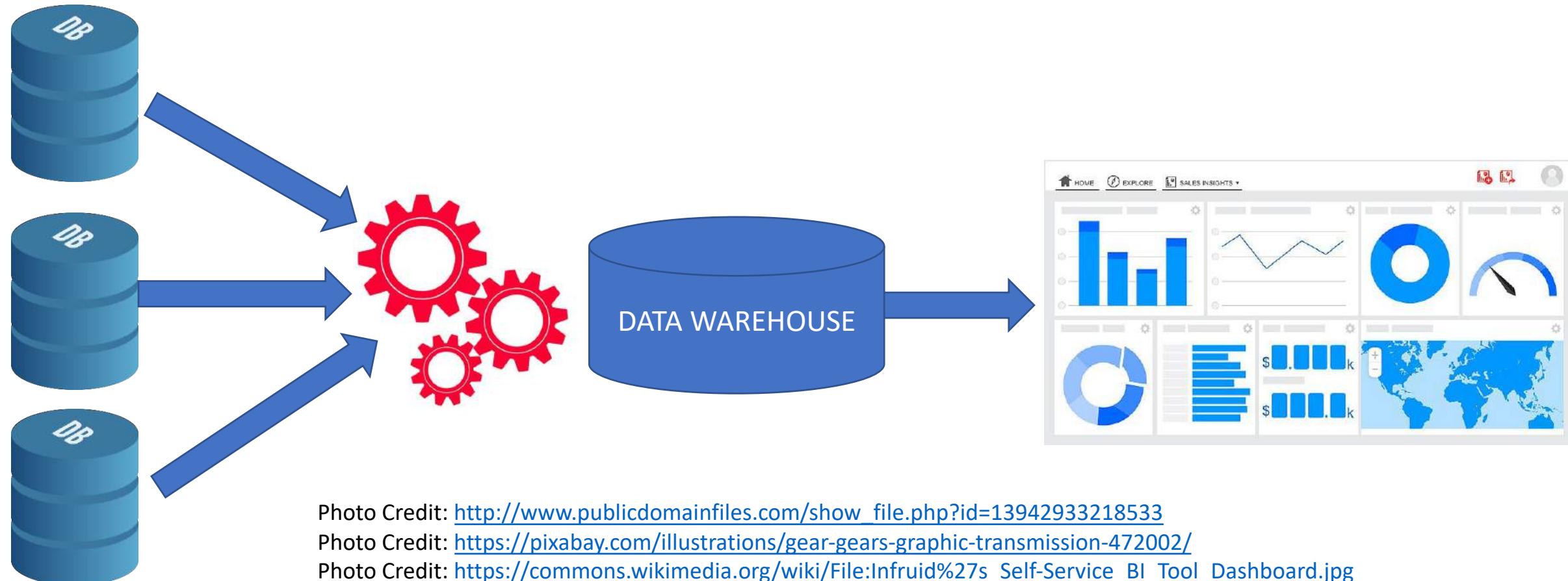


Photo Credit: <https://commons.wikimedia.org/wiki/File:Cascaded-keys.PNG>

DATA WAREHOUSE



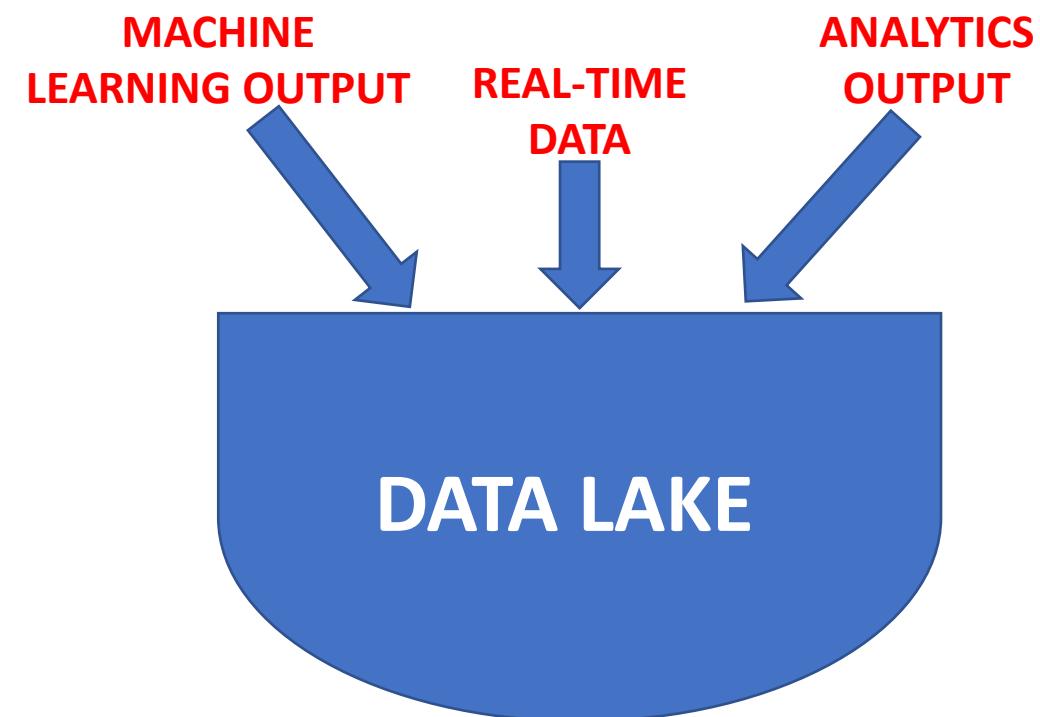
- A data warehouse exists on top of several databases and used for business intelligence.
- Data warehouse consumes data from all these databases and creates a layer optimized to perform data analytics.
- Schema is done on import.



DATA LAKE



- A data lake is a centralized repository for structured and unstructured data storage.
- Data lakes could be used to store raw data as is without any structure (schema).
- There is no need to perform any ETL or transformation jobs on it.
- You can store many types of data such images, text, files, videos.
- You can store machine learning models artifacts, retime data, and analytics outputs in data lakes.
- Processing could be done on export so schema is defined on read.



AWS KEY STORAGE SERVICES



AWS KEY STORAGE TYPES



S

History

Amazon SageMaker

Console Home

Amazon Polly

Amazon Rekognition

Amazon Comprehend

Amazon Translate

Compute

- EC2
- Lightsail
- ECR
- ECS
- EKS
- Lambda
- Batch
- Elastic Beanstalk
- Serverless Application Repository

Analytics

- Athena
- EMR
- CloudSearch
- Elasticsearch Service
- Kinesis
- QuickSight
- Data Pipeline
- AWS Data Exchange
- AWS Glue
- AWS Lake Formation
- MSK

Customer Enablement

- AWS IQ
- Support
- Managed Services

Business Applications

- Alexa for Business
- Amazon Chime
- WorkMail

Blockchain

- Amazon Managed Blockchain

Satellite

- Ground Station

End User Computing

- WorkSpaces
- AppStream 2.0
- WorkDocs
- WorkLink

Internet Of Things

- IoT Core
- Amazon FreeRTOS
- IoT 1-Click
- IoT Analytics
- IoT Device Defender
- IoT Device Management
- IoT Events
- IoT Greengrass
- IoT SiteWise
- IoT Things Graph

Management & Governance

- AWS Organizations
- CloudWatch
- AWS Auto Scaling
- CloudFormation
- CloudTrail
- Config
- OpsWorks
- Service Catalog
- Systems Manager
- Trusted Advisor
- Control Tower
- AWS License Manager
- AWS Well-Architected Tool
- Personal Health Dashboard
- AWS Chatbot
- Launch Wizard

Game Development

- Amazon GameLift

Storage

- S3
- EFS
- FSx
- S3 Glacier
- Storage Gateway
- AWS Backup

Database

- RDS
- DynamoDB
- ElastiCache
- Neptune
- Amazon Redshift
- Amazon QLDB
- Amazon DocumentDB

AWS KEY STORAGE TYPES

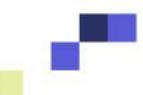


1. AMAZON S3

2. AURORA RDS

3. REDSHIFT

4. DYNAMODB



1. AMAZON S3

- Amazon Simple Storage Service (Amazon S3) is a storage service that allows enterprises/individuals to store and protect any amount of data.
- Amazon S3 is extremely easy to use and allows enterprises to organize their data and configure finely-tuned access controls.
- Amazon S3 extremely durable to 99.99999999% (11 9's).



AMAZON S3

Photo Credit: https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg

1. AMAZON S3



CREATE A BUCKET AND SIMPLY UPLOAD DATA TO IT

The screenshot shows the AWS S3 console with a red arrow pointing to the '+ Create bucket' button. The main content area displays three steps: 'Create a new bucket' (with an icon of a bucket), 'Upload your data' (with an icon of a bucket and an upload arrow), and 'Set up your permissions' (with an icon of two people and a plus sign). Each step has a 'Learn more' link and a 'Get started' button.

S3 buckets

Search for buckets

All access types

+ Create bucket Edit public access settings Empty Delete

0 Buckets 0 Regions

You do not have any buckets. Here is how to get started with Amazon S3.

Create a new bucket

Buckets are globally unique containers for everything that you store in Amazon S3.

Learn more

Upload your data

After you create a bucket, you can upload your objects (for example, your photo or video files).

Learn more

Get started

Set up your permissions

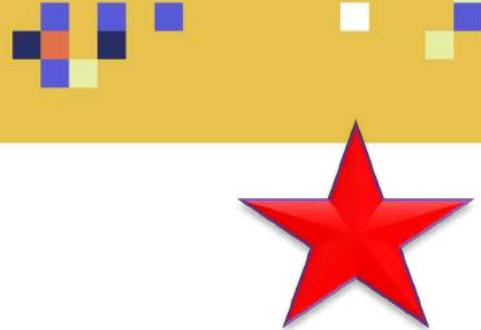
By default, the permissions on an object are private, but you can set up access control policies to grant permissions to others.

Learn more



2. RDS AURORA

- Amazon Aurora is a fully managed by Amazon Relational Database (RDS) service.
- Transactional style database.
- Amazon Aurora is a MySQL and PostgreSQL-compatible relational database.
- You do not have to deal with administration tasks such as hardware provisioning, creating backups and database setup.
- It features continuous backup to Amazon S3, and replication across three Availability Zones.
- Many engines available to create database



AMAZON RDS AURORA

- [Watch Video: https://aws.amazon.com/rds/aurora/](https://aws.amazon.com/rds/aurora/)

2. RDS AURORA



MANY ENGINE OPTIONS ARE
AVAILABLE TO CREATE
DATABASE

The screenshot shows the 'Create database' page in the AWS RDS console. At the top, a message says: 'We listened to your feedback! Now, create a database with a single click using our pre-built configurations! Or choose your own configurations.' Below this, the breadcrumb navigation shows 'RDS > Create database'. The main section is titled 'Create database' and contains a 'Choose a database creation method' section with two options: 'Standard Create' (selected) and 'Easy Create'. The 'Standard Create' option is described as setting all configuration options, including availability, security, backups, and maintenance. The 'Easy Create' option is described as using recommended best-practice configurations where some options can be changed after creation. A large red arrow points from the text 'MANY ENGINE OPTIONS ARE AVAILABLE TO CREATE DATABASE' to the 'Engine options' section. This section is titled 'Engine type' and includes icons and labels for six engines: Amazon Aurora (selected), MySQL, MariaDB, PostgreSQL, Oracle, and Microsoft SQL Server. Below the engine type section is an 'Edition' section with two options: 'Amazon Aurora with MySQL compatibility' (selected) and 'Amazon Aurora with PostgreSQL compatibility'.

We listened to your feedback!
Now, create a database with a single click using our pre-built configurations! Or choose your own configurations.
[Switch to your original interface.](#)

RDS > Create database

Create database

Choose a database creation method [Info](#)

Standard Create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.

Easy Create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type [Info](#)

Amazon Aurora

MySQL

MariaDB

PostgreSQL

Oracle

Microsoft SQL Server

Edition

Amazon Aurora with MySQL compatibility

Amazon Aurora with PostgreSQL compatibility

3. REDSHIFT

- Amazon Redshift is the fastest cloud **data warehousing** service that could be used to perform business analytics.
- Extremely fast and optimized performance since it relies on columnar storage and data compression.
- Queries are run against data stored in redshift storage or against data stored in S3.
- Redshift uses a unique data warehousing architecture that relies on Massively Parallel Processing (MPP).
- MPP parallelize and distribute SQL operations.
- Redshift uses machine learning to optimize performance.



AMAZON REDSHIFT

3. REDSHIFT



- Redshift is used on top of several databases and used for business intelligence.
- It consumes data from many sources and creates a layer optimized to perform data analytics.

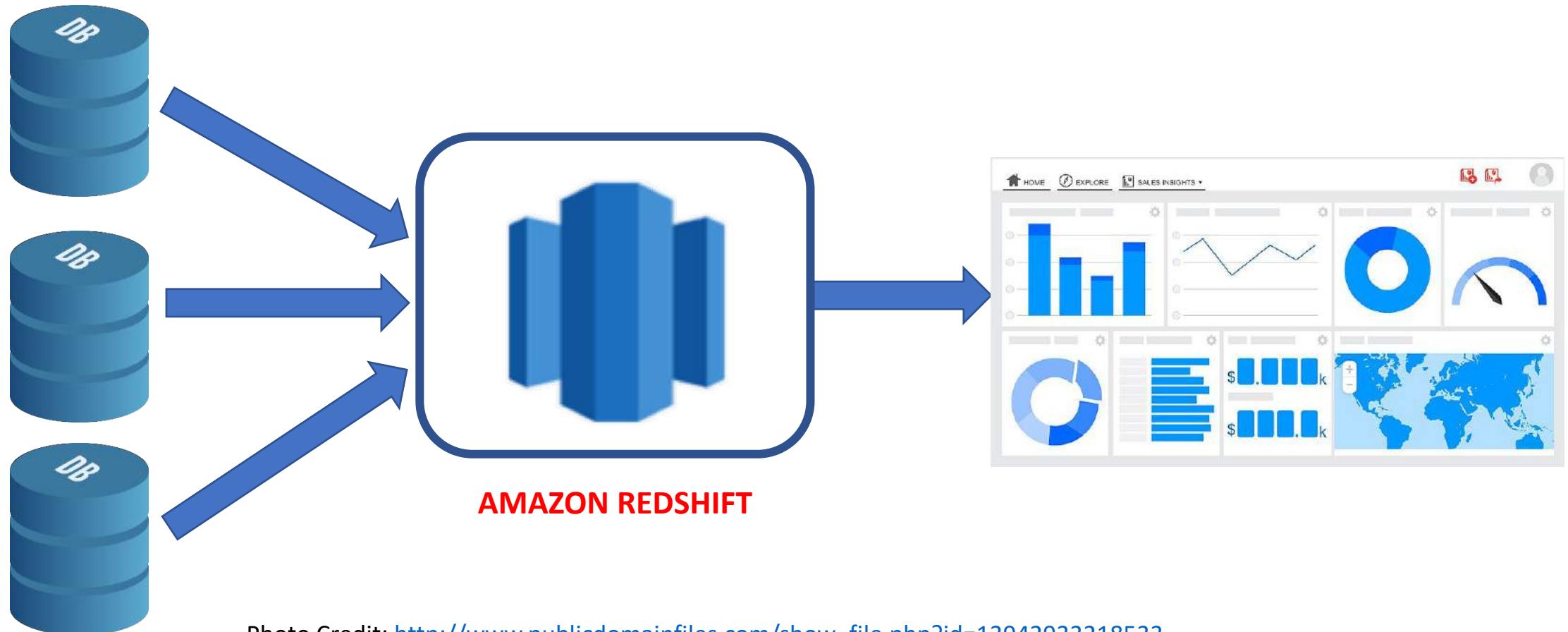


Photo Credit: http://www.publicdomainfiles.com/show_file.php?id=13942933218533

Photo Credit: https://commons.wikimedia.org/wiki/File:Infruid%27s_Self-Service_BI_Tool_Dashboard.jpg

3. REDSHIFT



AWS Services Resource Groups

Switch to the new Amazon Redshift console. We have been listening to your feedback to make several improvements. Try it out and tell us what you think!

Redshift dashboard

- Clusters
- Query editor
- Saved queries
- Snapshots
- Security
- Workload management New
- Reserved nodes
- Advisor
- Events
- Connect count
- What's new

Getting Started

- Getting Started with Amazon Redshift
- Overview and features
- Free Trial
- Evaluation and POC support
- Documentation
- Query your S3 data lake with Redshift Spe
- Pricing and Specs
- Purchase a Reserved Node

AWS Marketplace

- Matillion ETL for Amazon Redshift
 - By Matillion
 - Rating ★★★★☆
 - \$1.37 to \$5.48/hr for software + AWS usage
 - [View all Data Integration](#)
- TIBCO Spotfire® Analytics for AWS (Hourly)
 - By TIBCO Software Inc.
 - Rating ★★★★☆
 - Starting from \$1.20/hr or from \$8,400/yr (24 hrs)
 - [AWS usage fees](#)
 - [View all Infrastructure Software](#)
- Yellowfin 8.0.3 for AWS (12 Month, 3 User)
 - By Yellowfin
 - Rating ★★★★☆
 - Bring Your Own License + AWS usage fees
 - [View all Infrastructure Software](#)
- [Find more software on AWS Marketplace](#)

Launch cluster Learn more

With a few clicks, you can create your first Amazon Redshift cluster in minutes.

Quick launch cluster

Query Editor Learn more

Amazon Redshift console now supports writing, running, and saving queries.

Open Query Editor

Find the best cluster configuration for your needs Learn more

What is your uncompressed data size? GB TB PB

0 250 500 750 1000 20 GB

dc2.large

Best throughput at the lowest cost

On-demand: \$0.50/hour
Reserved (1 yr): \$0.32/hour
Reserved (3 yr): \$0.18/hour

Nodes: 2 Compressed storage: 0.32 TB

Launch this cluster

If you're doing a Proof-of-Concept on Redshift, follow this guide or reach the Amazon Redshift team for help.

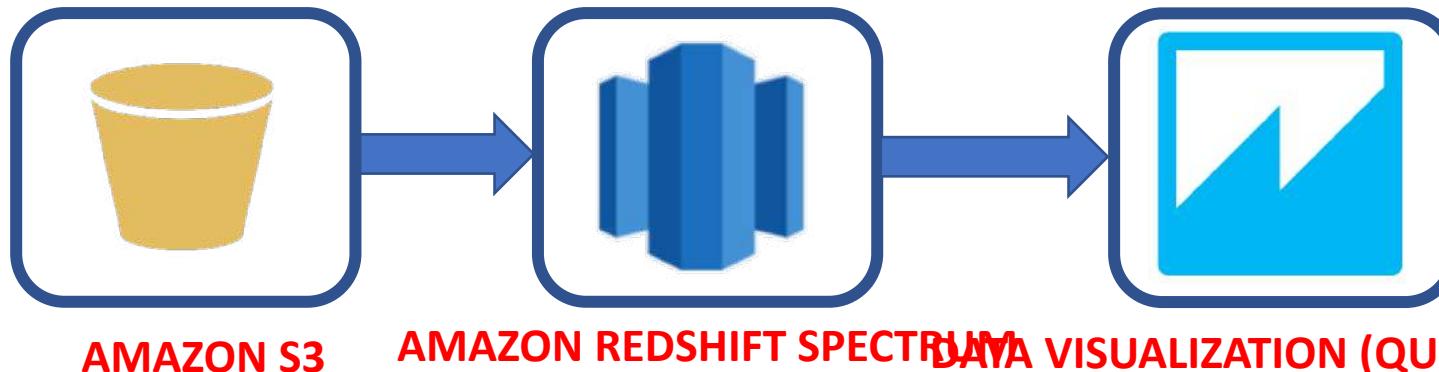
A red arrow points from the text "LAUNCH REDSHIFT CLUSTERS" to the "Quick launch cluster" button on the left side of the dashboard.

LAUNCH REDSHIFT CLUSTERS

3. AMAZON REDSHIFT SPECTRUM



- AWS Amazon Redshift Spectrum allows analysts to run SQL queries on data stored in **Amazon S3 buckets directly**.
- Redshift can dramatically save time because it does not require transferring data from S3 to a database.
- Redshift Spectrum can work well with unstructured S3 data lakes.



AMAZON REDSHIFT SPECTRUM

AMAZON S3

AMAZON REDSHIFT SPECTRUM

DATA VISUALIZATION (QUICKSIGHT)



4. AMAZON DYNAMODB

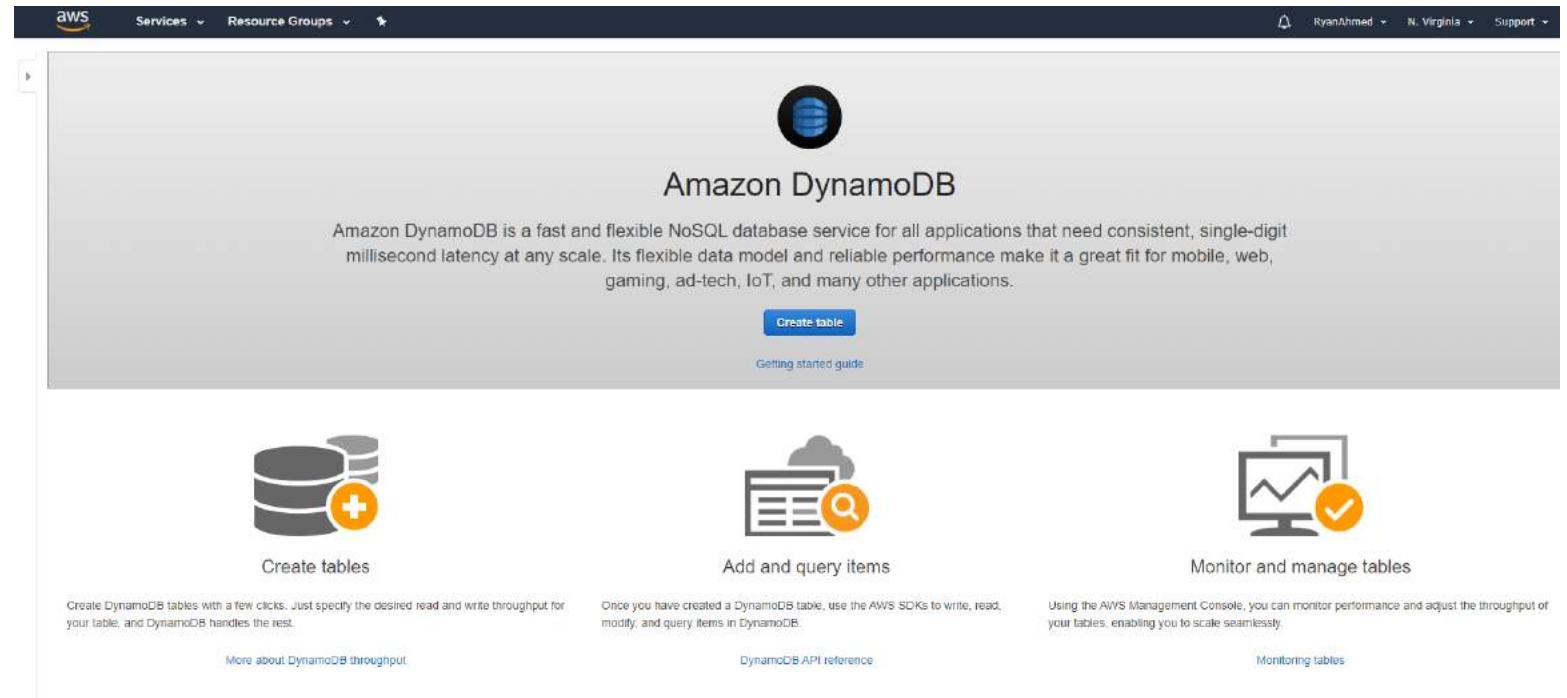
- Amazon DynamoDB is a fully managed NoSQL key-value and document database (not relational database so no schema is required).
- DynamoDB is extremely scalable with minimum latency:
 - 10 trillion requests/day
 - 20 million requests/second
- Create a new table for your application and let DynamoDB handle the rest.
- It works great for storing machine learning models for inference by application
- Watch Video: <https://aws.amazon.com/dynamodb/>



AMAZON DYNAMODB

4. AMAZON DYNAMODB

```
{  
  "Customers": [  
    {  
      "ID": 1,  
      "Name": Ryan,  
      "Age": 28,  
    },  
    {  
      "ID": 2,  
      "Name": Mitch,  
      "Age": 21,  
    }  
  ]  
}  
  
KEY:VALUE
```



AWS MACHINE LEARNING CERTIFICATION



DOMAIN #1: DATA ENGINEERING (20% EXAM)



AWS ML CERTIFICATION EXAM DOMAINS

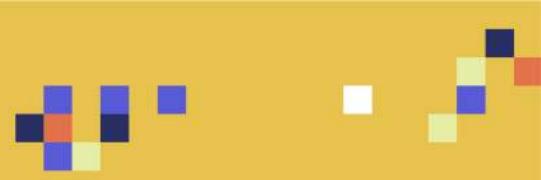


Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #1: WHERE ARE WE NOW!!?

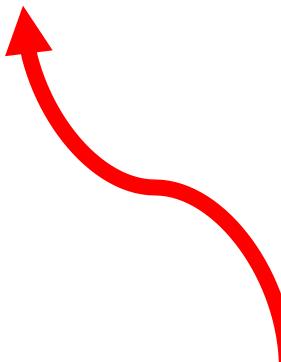


SECTION #1: INTRODUCTION, DATA/ML LINGO, AWS DATA STORAGE

- What is Machine Learning and Artificial Intelligence?
- What is Amazon Web Services (AWS)?
- Artificial Intelligence and Machine learning Lingo (data types, Labeled vs. unlabeled, sagemaker groundtruth)
- structured vs. unstructured and database vs. data lake vs. data storage
- AWS Data Storage (Redshift, RDS, S3, DynamoDB)

SECTION #2: AMAZON S3

- Amazon S3 in Depth (partitions, tags)
- Amazon S3 Storage Tiers and Lifecycles
- Amazon S3 Encryption and Security
- Amazon S3 Encryption and Security – Part #2 (ACL, CloudWatch, CloudTrail, VPC)
- Additional Notes (Elasticsearch, ElastiCache, and Database vs. data warehouse)



WE ARE HERE!



DOMAIN #1 OVERVIEW:

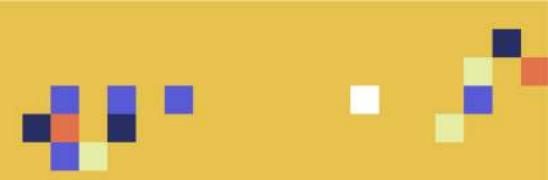
SECTION #3: AWS DATA MIGRATION, GLUE, PIPELINE, STEP AND BATCH

- AWS Glue (crawlers, features, built-in transformations etc)
- AWS Data pipeline
- AWS Data Migration Service (DMS)
- AWS Batch
- Step Function

SECTION #4: DATA STREAMING & KINESIS

- Kinesis Overview
- Kinesis Video Streams
- Kinesis Data Streams
- Kinesis Firehose
- Kinesis Analytics and Random Cut Forest

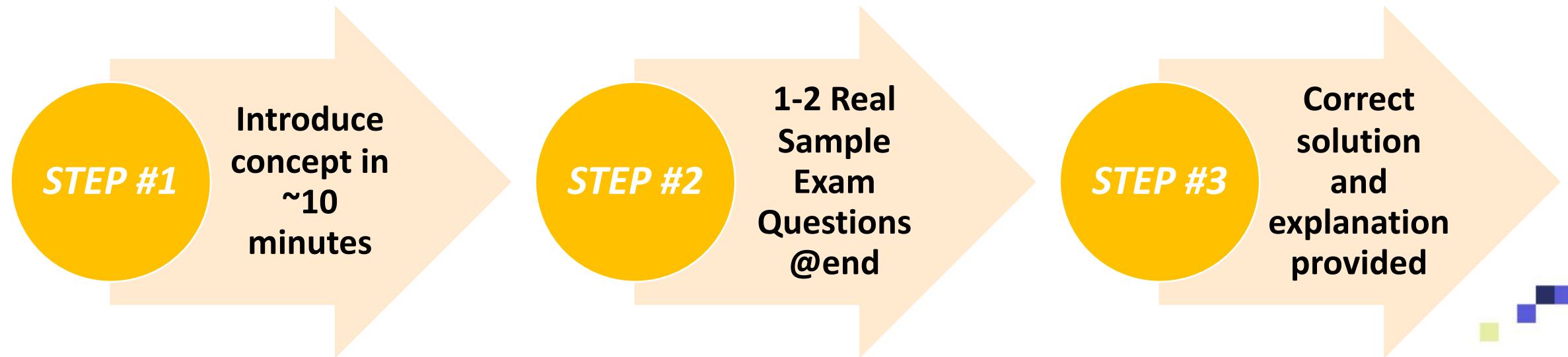
LECTURE DESIGN



- We know how hard it is to study for an exam especially if you have a busy schedule.
- This course is designed to be extremely on point and optimized to pass the exam.

No boring content. Zero unnecessary information.

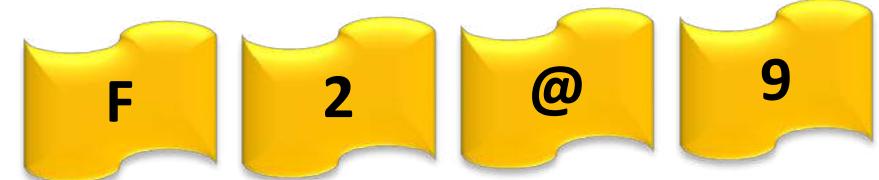
- Here's the lecture structure that we will follow:



RECALL OUR MINI CHALLENGE AND PRIZE!



- For those of you who will successfully complete the entire section and watch the videos till the end, they will receive a valuable prize!



AMAZON S3



WHAT IS AMAZON S3?



- Amazon Simple Storage Service (Amazon S3) is a storage service that allows enterprises/individuals to store and protect any amount of data.
- Amazon S3 offers numerous enhanced features such as:
 - (1) Scalability
 - (2) Data availability
 - (3) Security
 - (4) Performance
- Amazon S3 is extremely easy to use and allows enterprises to organize their data and configure finely-tuned access controls.
- Amazon S3 extremely durable to 99.99999999% (11 9's).

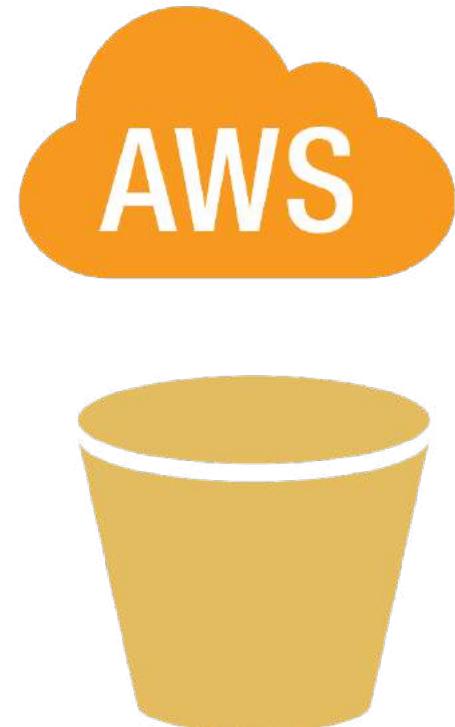


Photo Credit: https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_AWS_Cloud.svg

Photo Credit: https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg



WHAT IS AMAZON S3? CONTINUED

- Amazon Simple Storage Service (Amazon S3) is built to be extremely simple and robust.
- Amazon S3 allows customers to store data in buckets or directories (much like folders).
- A bucket is a container for objects stored in Amazon S3. Every object is contained in a bucket.
- Each of the buckets will have **global unique name**.
- You can store an infinite amount of data in a bucket in which each object can contain up to 5 TB of data.
- For example, if we have an object **images/mycat.jpg** is stored in the **mitchsteve** bucket, use can use the following URL to access it:

<http://mitchsteve.s3.amazonaws.com/images/mycat.jpg>



Photo Credit: https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_AWS_Cloud.svg

Photo Credit: https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg

WHAT IS AMAZON S3?



**CREATE A
BUCKET AND
SIMPLY UPLOAD
DATA TO IT**

The screenshot shows the AWS S3 buckets page. On the left sidebar, there are links for 'Buckets', 'Batch operations', 'Block public access (account settings)', and 'Feature spotlight'. A red arrow points from the text 'CREATE A BUCKET AND SIMPLY UPLOAD DATA TO IT' to the '+ Create bucket' button. The main content area is titled 'S3 buckets' and features a search bar and a 'Create bucket' button. It displays a message: 'You do not have any buckets. Here is how to get started with Amazon S3.' Below this are three sections: 'Create a new bucket' (with an icon of a bucket and clouds), 'Upload your data' (with an icon of a bucket and an upward arrow), and 'Set up your permissions' (with an icon of two people and a plus sign). Each section has a 'Learn more' link and a 'Get started' button.

Amazon S3

Buckets

Batch operations

Block public access (account settings)

Feature spotlight

+ Create bucket

Search for buckets

All access types

0 Buckets 0 Regions

You do not have any buckets. Here is how to get started with Amazon S3.

Create a new bucket

Upload your data

Set up your permissions

Buckets are globally unique containers for everything that you store in Amazon S3.

After you create a bucket, you can upload your objects (for example, your photo or video files).

By default, the permissions on an object are private, but you can set up access control policies to grant permissions to others.

Learn more

Learn more

Learn more

Get started

OBJECT TAGS

- You can use object Tags (key value pairs) to categorize storage.
- Here's an example of an object tag:
 - Project = Machinelearning
 - Classification = confidential
 - PHI = True
- The maximum number of tags per object is 10.
- Objects tags are important for:
 - Granting or denying permission (for example: read only user access).
 - Managing object lifecycle by creating a lifecycle rule based on associated tags.
 - Performing analytics.

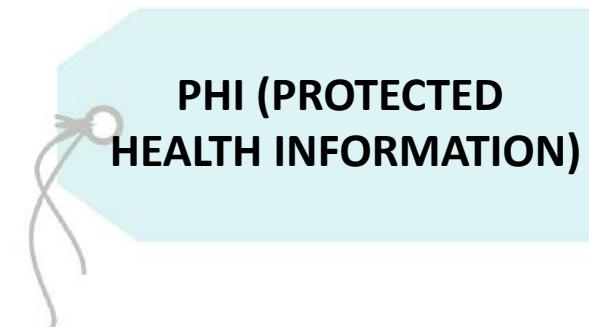
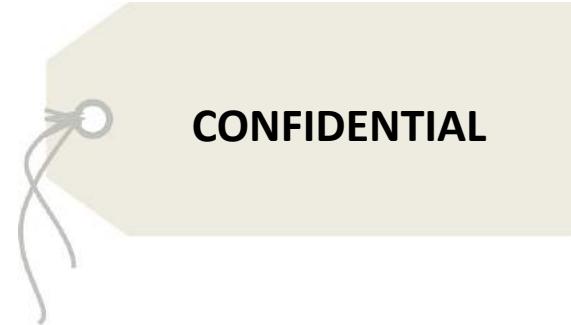


Photo Credit: <https://www.needpix.com/photo/1265094/tag-label-price-hang-tag-ticket-string-retail-business-promotion>

AWS S3 DATA LAKE



- Amazon S3 offers a great, easy to use solution to create a data lake because its highly scalable.
- Amazon S3 allows enterprises and individuals to increase their storage from gigabytes to petabytes in minutes.
- It is cost effective and pay-per-use model is very efficient. You do not need to buy or administer any hardware.
- Amazon S3 is extremely durable with 99.99999999%.
- Offers easy to use access controls so users can grant and deny fine grained access.
- Amazon S3 works with common formats such as: CSV, Parquet, ORC, Avro, Protobuf and JSON.
- AWS S3 hosts the data that machine learning services such as SageMaker will use for model training and testing.



AWS S3 DATA LAKE: ENABLING FEATURES



COMPUTE AND DATA PROCESSING ARE DECOUPLED

- Storage and compute are coupled in most data warehousing solutions (Hadoop) increasing complexity and cost.
- Amazon S3 offers a cost effective solution to store any data of any size in its native format.
- Users can launch Amazon Elastic Compute (EC2) cloud instances to access and process data.

CENTRALIZED DATA ARCHITECTURE

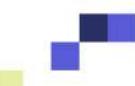
- Traditional solutions forces enterprises to have multiple copies of data distributed across multiple processing platforms.
- S3 overcomes that and offers a “multi-tenant” environment.
- Include one copy of the data and many users can apply their tools on the same data.

INTEGRATION WITH CLUSTERLESS/SERVERLESS AWS SERVICES

- Amazon S3 is seamlessly integrated with other services to query/process data such as Athena, Redshift, Rekognition, AWS Glue.
- Amazon S3 integrates with AWS Lambda serverless computing to run code without provisioning or managing servers.

STANDARDIZED APIs

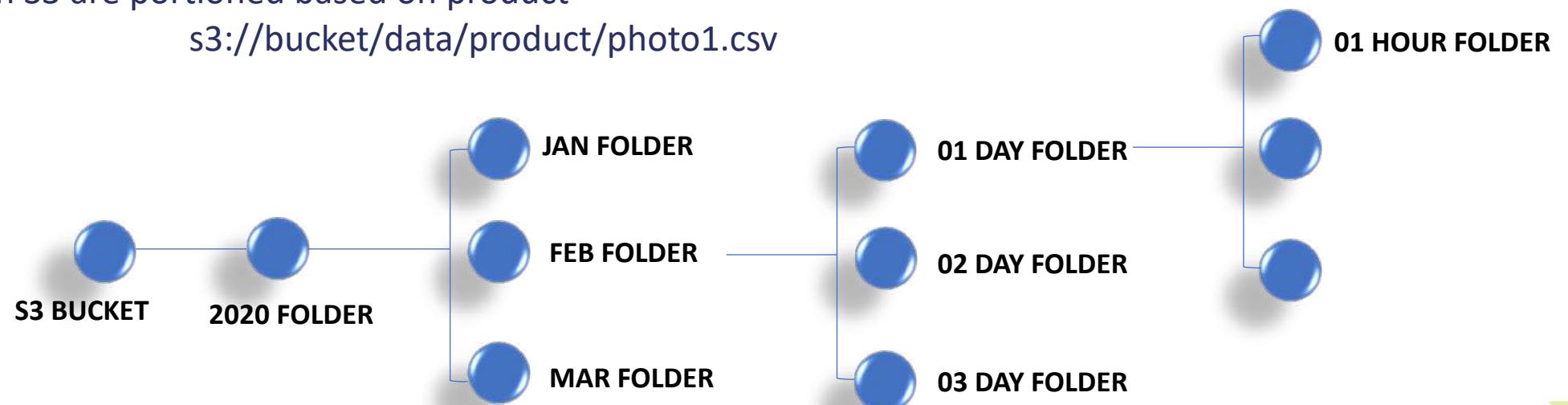
- Amazon S3 RESTful APIs easy to and work well with Apache Hadoop and many analytics tools.
- Users can leverage their skills in certain analytics tool but using data available in Amazon S3.



AWS S3 DATA PARTITIONING



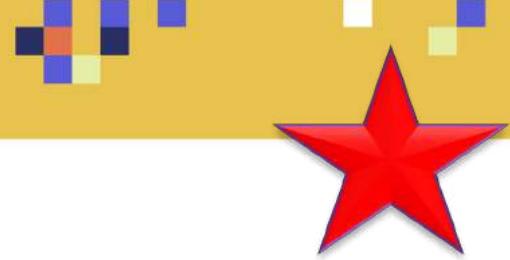
- AWS S3 data partitioning is critical when querying data because it could dramatically reduce the cost required for scanning.
- Partitioning is important when data query in Amazon Athena an Redshift (\$5/TB scanned).
- Data could be partitioned by time, product or customized by user.
- Examples:
 - Folders on S3 are partitioned based year/month/day/hour
`s3://bucket/data/year/month/day/hour/photo1.csv`
 - Folders on S3 are portioned based on product
`s3://bucket/data/product/photo1.csv`



AMAZON S3 STORAGE TIERS AND LIFECYCLES

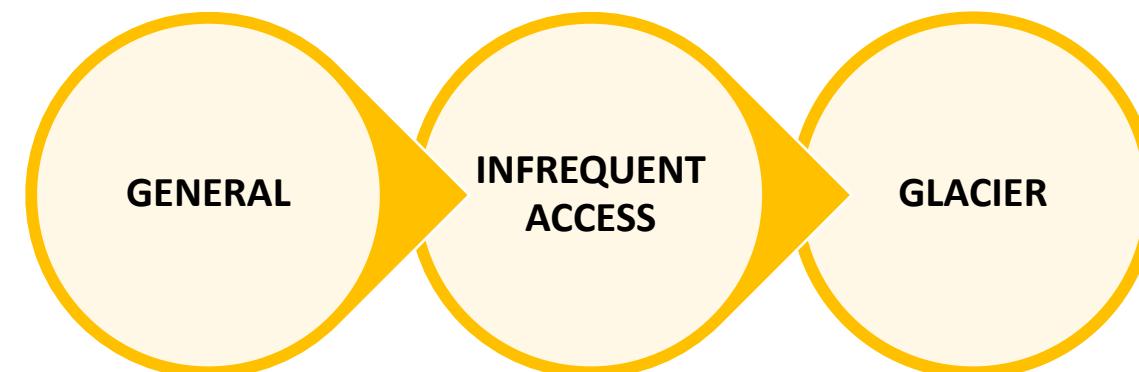


AWS S3 STORAGE TIERS



- Amazon S3 offers a range of storage classes:
 - S3 Standard: works well with storage that is general purpose and frequently accessed
 - S3 Intelligent-Tiering: works well with data that has varying access patterns
 - S3 Standard-Infrequent Access (Standard-IA): for long-lived but less frequently accessed data
 - S3 One Zone-Infrequent Access (One Zone-IA): for long-lived, but less frequently accessed data
 - Amazon S3 Glacier (S3 Glacier) and Amazon S3 Glacier Deep Archive (S3 Glacier Deep Archive): works well for long-term archived data.

Amazon S3 offers the ability to change the storage tiers throughout the data lifecycle by setting a lifecycle policy.



AWS S3 STORAGE TIERS SUMMARY



FREQUENTLY
ACCESSED

CHANGING
PATTERN

INFREQUENTLY ACCESSED

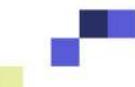
ARCHIVED

	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA†	S3 Glacier	S3 Glacier Deep Archive
Designed for durability	99.999999999% (11 9's)					
Designed for availability	99.99%	99.9%	99.9%	99.5%	99.99%	99.99%
Availability Zones	≥3	≥3	≥3	1	≥3	≥3
Minimum storage duration charge	N/A	30 days	30 days	30 days	90 days	180 days
Retrieval fee	N/A	N/A	per GB retrieved	per GB retrieved	per GB retrieved	per GB retrieved
First byte latency	milliseconds	milliseconds	milliseconds	milliseconds	select minutes or hours	select hours
Storage type	Object	Object	Object	Object	Object	Object
Lifecycle transitions	Yes	Yes	Yes	Yes	Yes	Yes

AWS S3 LIFECYCLES



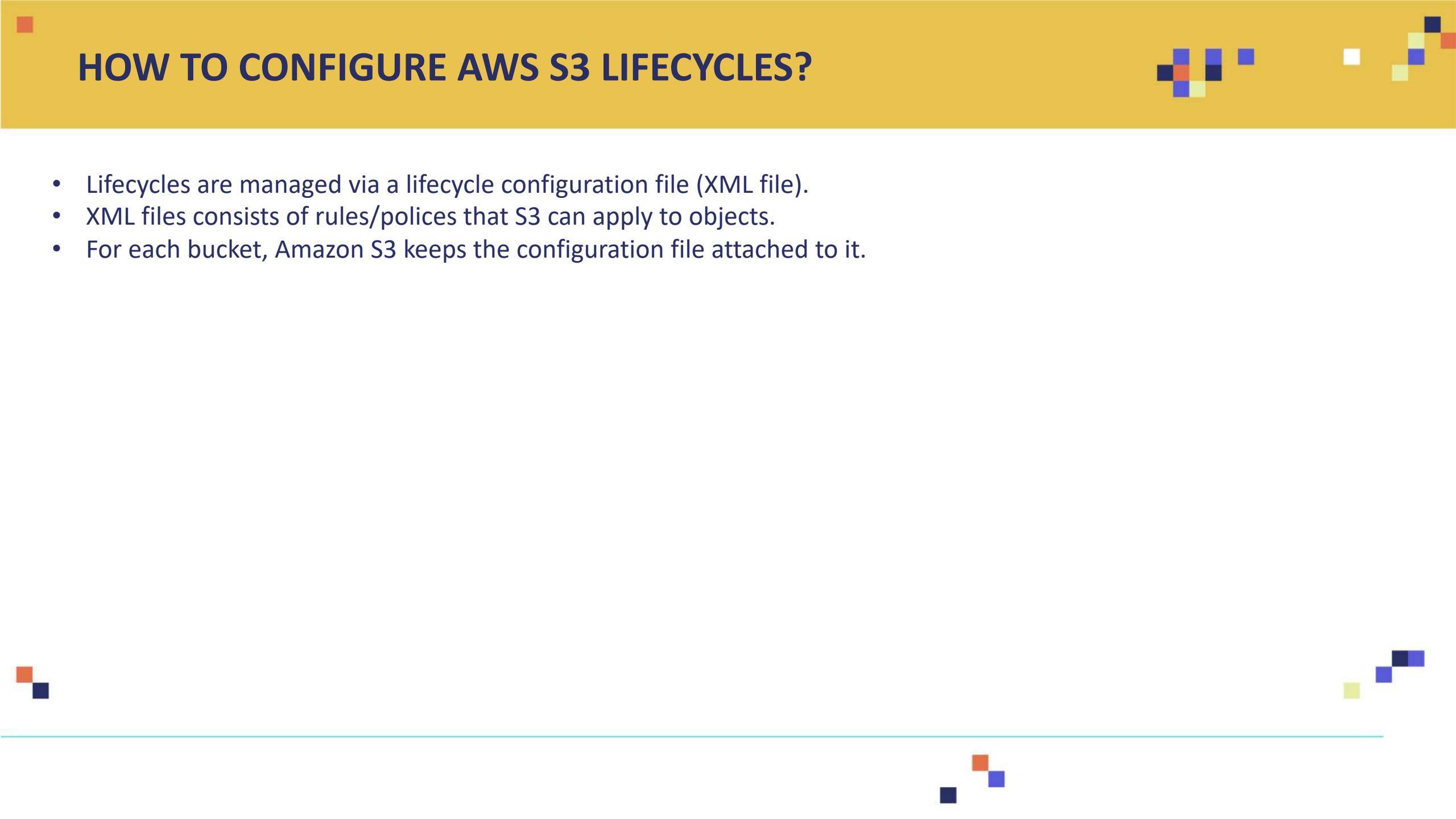
- In order to reduce cost, you can configure a lifecycle policy that S3 could apply to group of objects.
- Two types of actions are present:
 - **Transition actions:** policy that governs when an object transitions from one storage class to another.
 - transition object from STANDARD_IA storage class 30 days creation
 - Transition objects to the GLACIER storage class after 1 year.
 - **Expiration actions:** policy that governs when objects expire (deleted by S3).
- You can set the data to be archived once uploaded (Glacier storage class) for some cases such as:
 - Data that need to be retained for compliance purposes
 - Healthcare records
 - Long-term database backups



HOW TO CONFIGURE AWS S3 LIFECYCLES?



- Lifecycles are managed via a lifecycle configuration file (XML file).
- XML files consists of rules/polices that S3 can apply to objects.
- For each bucket, Amazon S3 keeps the configuration file attached to it.



AMAZON S3 ENCRYPTION



S3 ENCRYPTION

- Amazon S3 focuses primarily on data security so data encryption is an important feature of AWS S3.
- You can set a default encryption which can allow for default encryption settings to your created S3 bucket.
- With server-side encryption, Amazon **S3 encrypts an object before saving it to disk and decrypts it the object is downloaded.**
- Here are the options for encryption:
 - Amazon S3-managed keys (SSE-S3) – *common with ML services*
 - Customer master keys (CMKs) stored in AWS Key Management Service (AWS KMS). This allows for extra security – *common with ML services.*
 - SSE-C: users can manage their own keys to perform data encryption.
 - Client side encryption

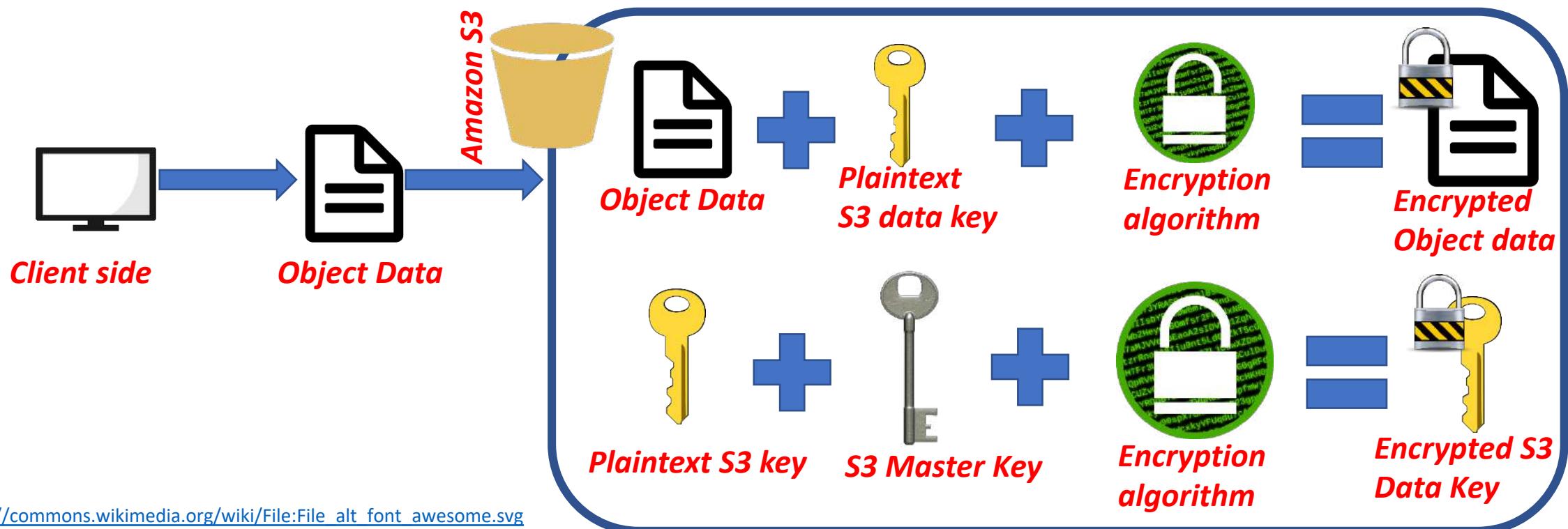


Photo Credit: <https://pixabay.com/vectors/computer-encryption-1294045/>

AMAZON S3 MANAGED KEYS (SSE-S3)



- Client uploads objects to S3 and select encryption method SSE-S3.
- S3 generates a plaintext key and encrypts object.
- Encrypted object is stored in S3.
- Amazon S3 master key then encrypts the plaintext key so now the key is encrypted as well and stored in S3.



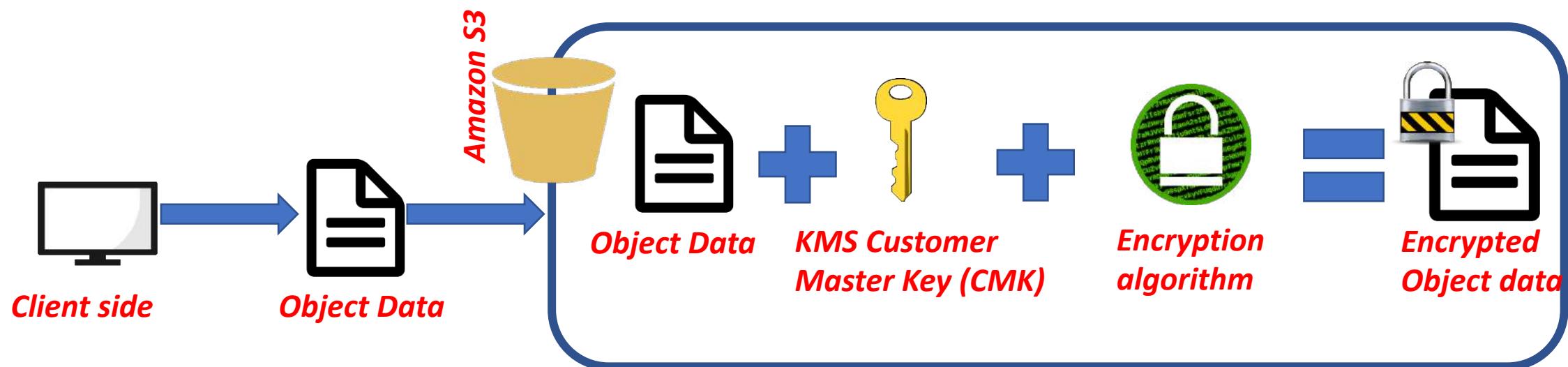
https://commons.wikimedia.org/wiki/File:File_alt_font_awesom.svg

https://en.wikipedia.org/wiki/File:Crypto_key.svg

Photo Credit: <https://pixabay.com/vectors/computer-encrypt-encryption-1294045/>

<https://www.needpix.com/photo/98552/padlock-encrypt-encrypted-lock-locked-protected-resistant-safe-secure>

AWS SSE-KMS: AWS KEY MANAGEMENT SERVICE



https://commons.wikimedia.org/wiki/File:File_alt_font_awesomel.svg

https://en.wikipedia.org/wiki/File:Crypto_key.svg

Photo Credit: <https://pixabay.com/vectors/computer-encrypt-encryption-1294045/>

<https://www.needpix.com/photo/98552/padlock-encrypt-encrypted-lock-locked-protected-resistant-safe-secure>

AMAZON S3 SECURITY



AWS S3 SECURITY

- Amazon S3 ensures the highest level of security to its customers.
- Amazon S3 follows a *shared security model* as follows:
 - **Security of the cloud:**
 - ❖ AWS ensures the protection of the infrastructure.
 - ❖ All services offered by AWS are very secure.
 - ❖ Security is being regularly audited by third party to ensure compliance.
 - **Security in the cloud:**
 - ❖ Users of AWS are responsible for their own data sensitivity and organization requirements.

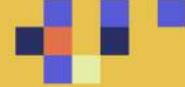
AWS S3 SECURITY: DATA PROTECTION IN S3



- Amazon S3 ensures data protection and durability by:
 1. Adding redundancy storage over multiple devices across many facilities/regions
 2. S3 ensure that any corrupted files are detected and repaired
 3. Using versioning: which allow users to retrieve several versions of the same object (S3 automatically retrieves the most up to date version).
- Amazon S3 has the following unique characteristics:
 - Ensures 99.99999999% durability and 99.99% availability
 - Can sustain data loss in two locations at the same time



AWS S3 SECURITY: BEST PRACTICES



MACIE IS A SECURITY SERVICE TO AUTOMATICALLY FIND SENSITIVE DATA AND ALERT USERS FOR ANOMALIES



AWS S3 SECURITY: RESOURCE Vs. USER BASED POLICES

- S3 buckets are private by default, only the owner can access the bucket.
- Owner can allows access to others by creating an access policy.
- There are generally two types of polices in AWS to grant permission to resources in AWS S3:
 - **Resource based:** these are access polices that you attach to your buckets.
 - ❖ Example: bucket polices and Access Control List (ACL)
 - **User based:** polices assigned to specific users in the account
 - ❖ Example: Identity and Access Management (IAM) polices
- Both polices use JSON-based access policy language.



AWS S3 SECURITY: ACCES CONTROL LIST (ACL)

- Access control lists (ACLs) belongs to the resource-based access policy.
- ACL could be used to allow permission to other AWS accounts to read/write objects
- Scenario: if a bucket owner let another AWS account upload object to the bucket. The AWS account that owns the object can only allow ACLs to this object.

AWS S3 SECURITY: S3 BLOCK PUBLIC ACCESS

- In order to manage public access to your Amazon S3, Amazon has a feature to “**block public access**” settings for buckets.
- Any newly created buckets does not allow public access by default.
- Users can change this and allow for public access if they want to.
- Amazon S3 block public settings overrides any created policies and permissions to block public access to resources/buckets (centralized control).
- If any request is made from anywhere to access a specific bucket, Amazon S3 checks “**block public access**” settings and these settings has the ultimate power (overrides any other policies).
- If “**block public access**” settings prevents access to the request, access request will be denied.

AMAZON S3 SECURITY – PART #2



AWS S3 SECURITY: LOGGING AND MONITORING IN AMAZON S3

- Monitoring is important to keep track of the health of AWS services/resources to ensure availability and performance.
- AWS provides many tools to monitor S3 resources as follows:

CloudWatch
Alarms

CloudTrail
Logs

S3 Access
Logs

AWS trusted
advisor

AWS S3 SECURITY: LOGGING AND MONITORING IN AMAZON S3

Amazon CloudWatch Alarms

- Used to send an alarm once a certain threshold is exceeded for multiple number of cycles.
- The alarm is sent to AWS autoscaling policy.



AWS CloudTrail Logs

- CloudTrail is used to track activity made by users, roles, or on AWS service.
- CloudTrail provides a record of past requests, IP addresses, timing of the request...etc.

Amazon S3 Access Logs

- Access logs are important to ensure security and to conduct access audits.
- Access logs are used to record/track requests made to buckets.

AWS Trusted Advisor

- Amazon offers trusted advisory service that makes recommendations on how to improve the systems performance and close any security gaps.
- Trusted Advisor provides the following:
 - Check that amazon S3 buckets have proper configuration.
 - Check amazon S3 buckets that have permissions set to “open access”.
 - Checks Amazon S3 buckets that did not enable versioning.

AWS S3 SECURITY: ADDITIONAL SECURITY (IMPORTANT)

- **Networking - VPC Endpoint Gateway:**

- Amazon Virtual Private Cloud (VPC) allows users to create AWS resources inside a virtual Network.
- This means that the traffic will never leave or go through the public internet and will stay inside the VPC for maximum security.
- A VPC endpoint will route requests to Amazon S3 and back to the VPC.

- **Tagging**

- You can use tagging in tandem with bucket policies and IAM policies to ensure security.
- Tag example: sensitive = TRUE

ADDITIONAL NOTES



DATA WAREHOUSE Vs. DATABASE



	Data Warehouse	Database
Used for?	Used for data analytics	Used for Transaction processing
Sources of data?	Data gathered from multiple sources	Data collected as-is from one source
Data writing frequency?	Bulk write operations per fixed schedule (every day)	Many write operations as data becomes available
Storage optimized for?	Optimized for high-speed query in column format	Optimized for high throughout write operations to a single row

<https://aws.amazon.com/data-warehouse/>



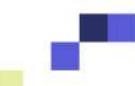
AMAZON ELASTICSEARCH



- Elasticsearch is an open source distributed search and analytics engine.
- Elasticsearch works with various types of data such as numerical, text, structured and unstructured.
- Elasticsearch performs data ingestion, enrichment, storage, analysis, and visualization.
- Tools are known as ELK Stack (Elasticsearch, Logstash, and Kibana).
- Amazon Elasticsearch Service is a fully managed service that allows for Elasticsearch deployment easily and securely.
- Amazon Elasticsearch is cost effective with zero upfront cost.
- Elasticsearch could be used for clickstream analytics, data indexing.
- [Watch Video: https://aws.amazon.com/elasticsearch-service/](https://aws.amazon.com/elasticsearch-service/)



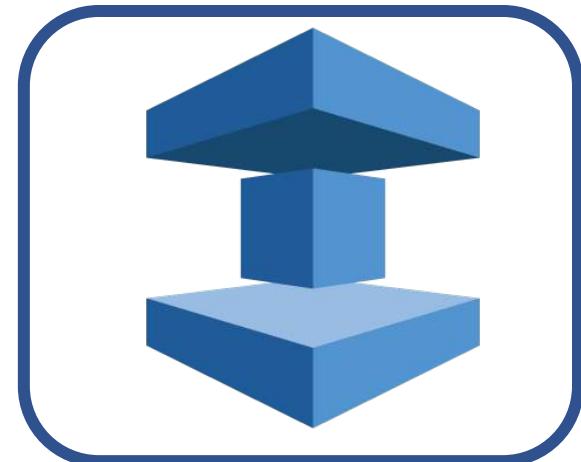
AMAZON
ELASTICSEARCH



AMAZON ELASTICACHE



- Amazon ElastiCache is an in-memory data store and cache.
- Amazon ElastiCache is extremely fast and designed specifically for demanding applications that need sub-millisecond response times.
- Amazon ElastiCache is designed for Gaming, Internet of things, and Healthcare applications.
- ElastiCache is used for data intensive apps by retrieving data from high throughput and low latency in-memory data stores.



AMAZON ELASTICACHE



AWS MACHINE LEARNING CERTIFICATION



DOMAIN #1: DATA ENGINEERING (20% EXAM)

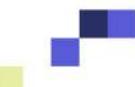


AWS ML CERTIFICATION EXAM DOMAINS



Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #1: WHERE ARE WE NOW!!?



SECTION #1: INTRODUCTION, DATA/ML LINGO, AWS DATA STORAGE

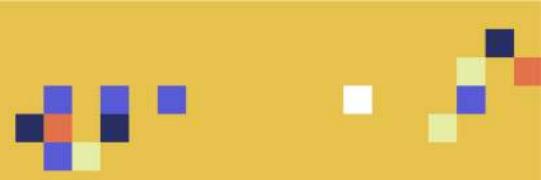
- What is Machine Learning and Artificial Intelligence?
- What is Amazon Web Services (AWS)?
- Artificial Intelligence and Machine learning Lingo (data types, Labeled vs. unlabeled, sagemaker groundtruth)
- structured vs. unstructured and database vs. data lake vs. data storage
- AWS Data Storage (Redshift, RDS, S3, DynamoDB)

SECTION #2: AMAZON S3

- Amazon S3 in Depth (partitions, tags)
- Amazon S3 Storage Tiers and Lifecycles
- Amazon S3 Encryption and Security
- Amazon S3 Encryption and Security – Part #2 (ACL, CloudWatch, CloudTrail, VPC)
- Additional Notes (Elasticsearch, ElastiCache, and Database vs. data warehouse)



DOMAIN #1: WHERE ARE WE NOW!!?



SECTION #3: AWS DATA MIGRATION, GLUE, PIPELINE, STEP AND BATCH

- AWS Glue (crawlers, features, built-in transformations etc)
- AWS Data pipeline
- AWS Data Migration Service (DMS)
- AWS Batch
- Step Function

WE ARE HERE!

SECTION #4: DATA STREAMING & KINESIS

- Kinesis Overview
- Kinesis Video Streams
- Kinesis Data Streams
- Kinesis Firehose
- Kinesis Analytics and Random Cut Forest



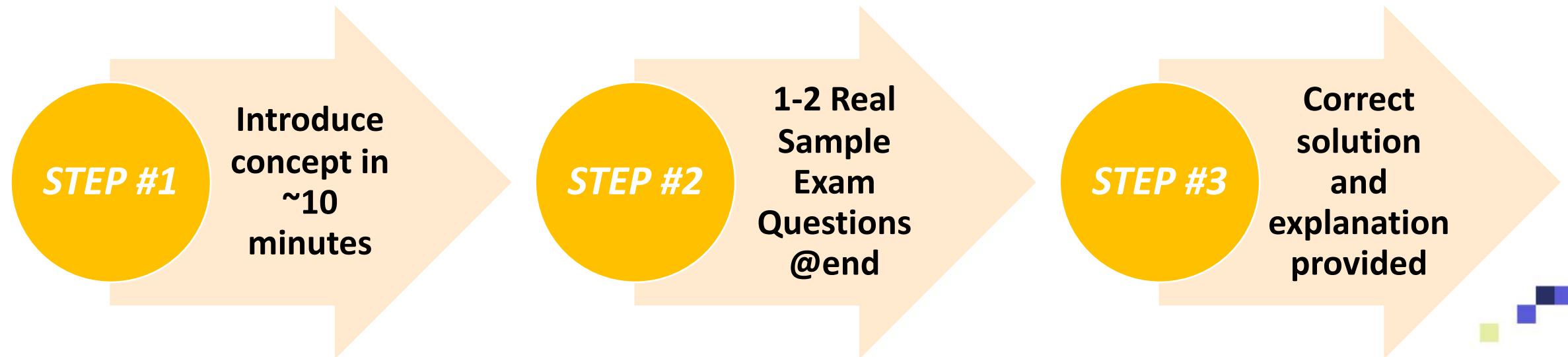
LECTURE DESIGN



- We know how hard it is to study for an exam especially if you have a busy schedule.
- This course is designed to be extremely on point and optimized to pass the exam.

No boring content. Zero unnecessary information.

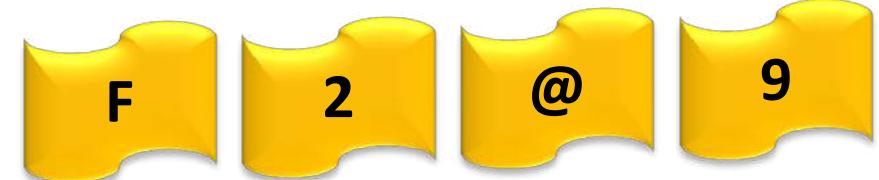
- Here's the lecture structure that we will follow:



RECALL OUR MINI CHALLENGE AND PRIZE!



- For those of you who will successfully complete the entire section and watch the videos till the end, they will receive a valuable prize!



- AWS GLUE – Part #1



AWS GLUE: OVERVIEW

- AWS Glue is a fully managed service that enable users to extract, transform, and load (ETL) data.
- Data generated by AWS Glue could be fed into analytics tools such as Amazon QuickSight.
- AWS Glue steps:
 1. Setup AWS Glue and direct it to your stored data on AWS
 2. AWS Glue extracts metadata such as table definition and schema and put it in the AWS Glue Data Catalog.
 3. Once cataloged, you can query the data and perform ETL.



AWS GLUE: UNIQUE FEATURES



SEAMLESS INTEGRATION WITH AWS SERVICES

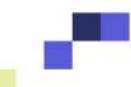
- AWS Glue works seamlessly with other AWS services which allow for easy integration.
- AWS Glue works well with stored data in Redshift, S3, and databases in Virtual Private Cloud (VPC) running on EC2.

MINIMUM CONFIGURATION HASSLE AND OPTIMIZED COST

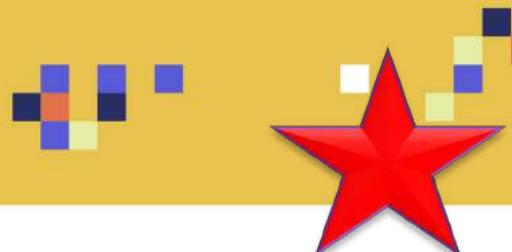
- AWS Glue offers pay per use service so no upfront cost or servers to manage.
- AWS Glue will take care of scaling and configuration in running the ETL jobs.

EASY TO USE

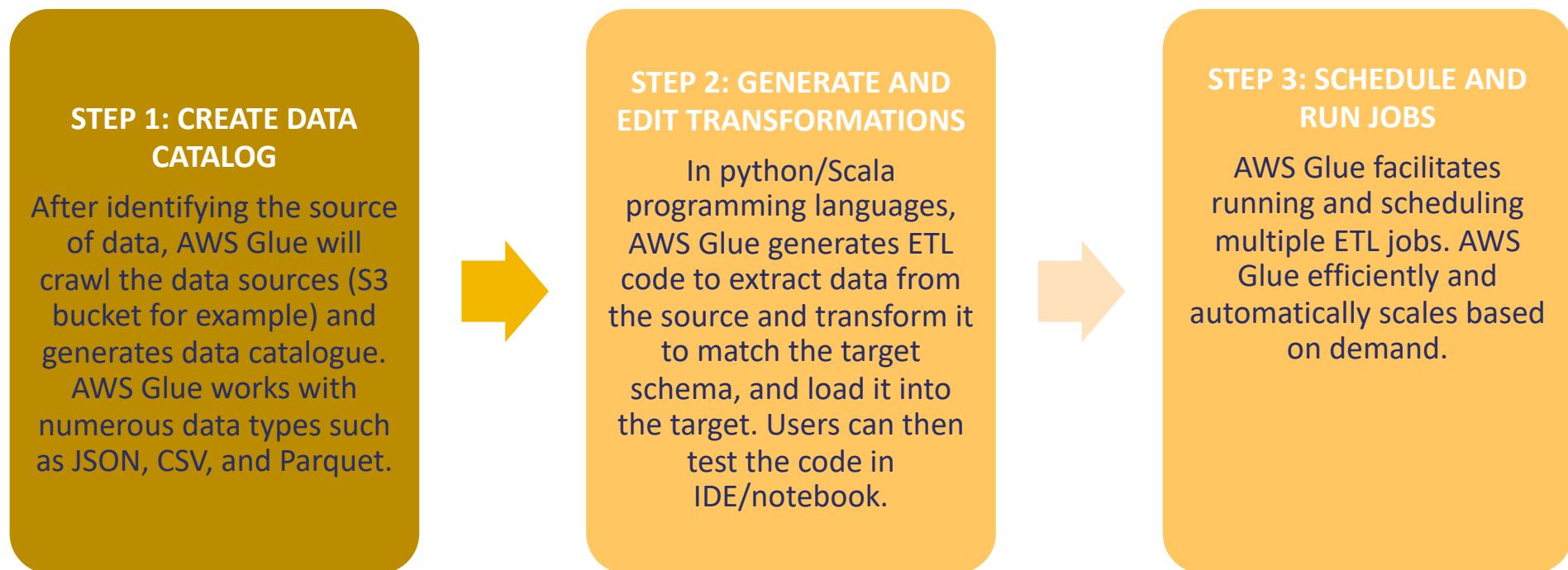
- AWS Glue crawls data storages, extract data schemas and transformations.
- AWS Glue generates data transformation codes on its own.



AWS GLUE: HOW DOES IT WORK?



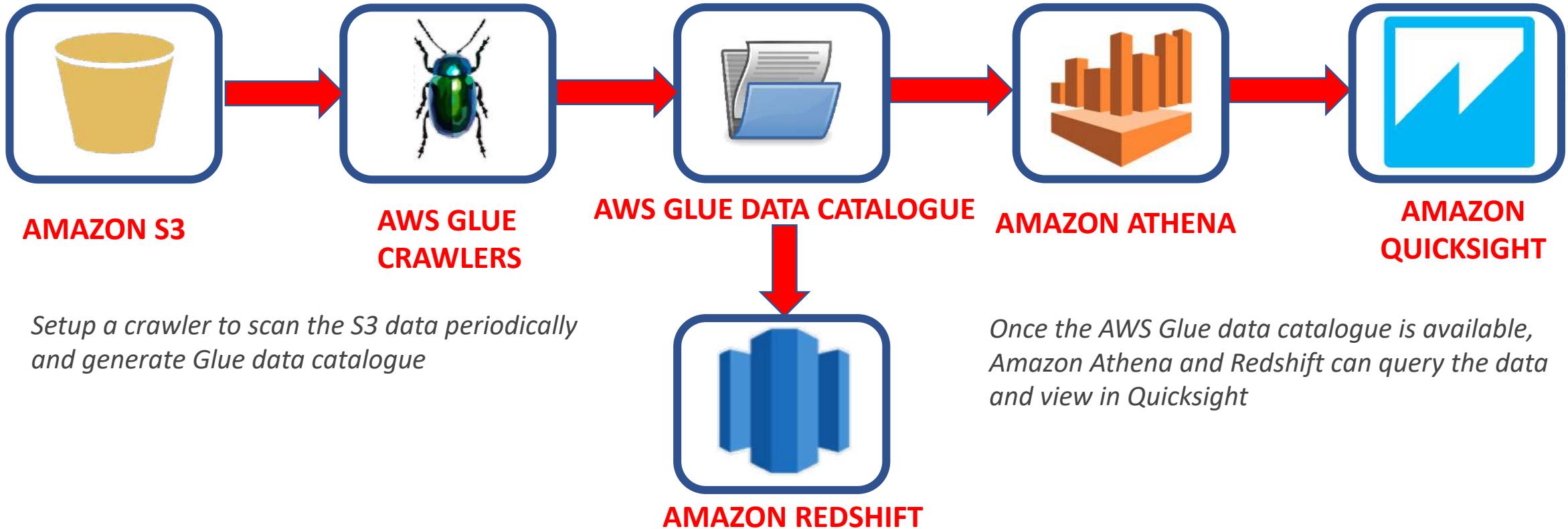
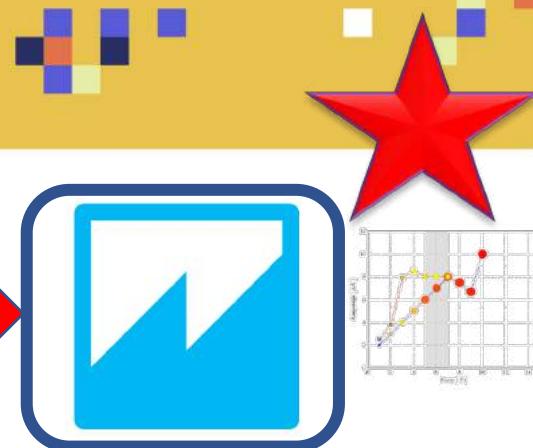
- AWS Glue is extremely easy to work with, users will have to first identify the data source/target.
- In python/Scala programming languages, AWS Glue generates ETL code to extract data from the source and transform it to match the target schema, and load it into the target.
- Users can then test the code in IDE/notebook.



- AWS GLUE – Part #2



AWS GLUE: USE CASE



Setup a crawler to scan the S3 data periodically and generate Glue data catalogue

Once the AWS Glue data catalogue is available, Amazon Athena and Redshift can query the data and view in Quicksight

- Athena and AWS Glue Data Catalog work seamlessly together, AWS Glue can be used to create databases and tables (schema) so that Athena can use it to query the data.

Photo Credit: https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg

Photo Credit: <https://pixabay.com/illustrations/insect-insects-insect-perfection-4470664/>

Photo Credit: <https://commons.wikimedia.org/wiki/File:Document-open.svg>

Photo Credit: https://commons.wikimedia.org/wiki/File:Magnifying_glass_01.svg

Photo Credit: <http://pgfplots.net/tikz/examples/plot-markers/>

Photo Credit: <https://publicdomainvectors.org/en/free-clipart/Image-of-electricity-spark-orange-icon/31132.html>

AWS GLUE: WHAT ARE CRAWLERS?



- Crawlers are used to scan the stored data and infer schemas and partitions.
- Once the crawler scans the data, it generates/updates the table in the Data Catalog.
- AWS Glue will rely on this data catalogue in performing Extract, transform, and load (ETL) jobs.
- You will require setting up an IAM role so that the crawler is able to access the data storage and infer schemas.
- AWS Glue works with several data formats such as JSON, Parquet, CSV.
- You can run the Crawler on demand or based on a schedule



**AWS GLUE
CRAWLERS**

Photo Credit: <https://pixabay.com/illustrations/insect-insects-insect-perfection-4470664/>



AWS GLUE: CRAWLERS AND CLASSIFIERS



Servi

AWS Glue

Data catalog

Databases

Tables

Connections

Crawlers

Classifiers

Settings

ETL

Workflows

Jobs

ML Transforms

Triggers

Dev endpoints

Notebooks

Security

Security configurations

Tutorials

Add crawler

Explore table

Add job

Resources

What's new 10+

- A classifier reads the storage data and generates a schema if it finds one.
- AWS Glue provides built-in classifiers for several formats as shown below.



**AWS GLUE
CRAWLERS**

Built-In Classifiers in AWS Glue

AWS Glue provides built-in classifiers for various formats, including JSON, CSV, web logs, and many database systems.

If AWS Glue doesn't find a custom classifier that fits the input data format with 100 percent certainty, it invokes the built-in classifiers in the order shown in the following table. The built-in classifiers return a result to indicate whether the format matches (`certainty=1.0`) or does not match (`certainty=0.0`). The first classifier that has `certainty=1.0` provides the classification string and schema for a metadata table in your Data Catalog.

Classifier type	Classification string	Notes
Apache Avro	avro	Reads the schema at the beginning of the file to determine format.
Apache ORC	orc	Reads the file metadata to determine format.
Apache Parquet	parquet	Reads the schema at the end of the file to determine format.
JSON	json	Reads the beginning of the file to determine format.
Binary JSON	bson	Reads the beginning of the file to determine format.
XML	xml	Reads the beginning of the file to determine format. AWS Glue determines the table schema based on XML tags in the document. For information about creating a custom XML classifier to specify rows in the document, see Writing XML Custom Classifiers .
Amazon Ion	ion	Reads the beginning of the file to determine format.
Combined Apache log	combined_apache	Determines log formats through a grok pattern.
Apache log	apache	Determines log formats through a grok pattern.
Linux kernel log	linux_kernel	Determines log formats through a grok pattern.
Microsoft log	microsoft_log	Determines log formats through a grok pattern.
Ruby log	ruby_logger	Reads the beginning of the file to determine format.
Squid 3.x log	squid	Reads the beginning of the file to determine format.

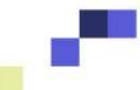
MANAGING PARTITIONS IN AWS GLUE



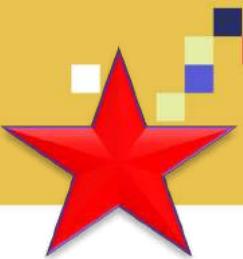
- Partitioning is critical for data organization.
- Proper portioning allow for an efficient query process.
- Once partitioned, AWS services such as Athena, Redshift, and Glue can utilize these partitions to filter through the data instead of scanning the entire bucket on S3.
- Partitioning allows for organizing data in a hierarchy directory structure as follows:

`s3://my_bucket/logs/year=2018/month=01/day=23/`

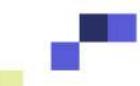
- AWS Glue Crawlers can rely on the partition structure to generate AWS Glue Data Catalog.
- The resulting partition columns can be queried using (1) AWS Glue ETL jobs and (2) Amazon Athena.



AWS GLUE BUILT-IN TRANSFORMATIONS



- AWS Glue offer several built-in transformations that could be called from an ETL script.
- Generally there are two types of transformations:
- **Bundled Transformations:**
 - DropFields Class
 - DropNullFields Class
 - Filter Class
 - Join Class
 - Map Class
 - MapToCollection Class
 - RenameField Class
 - ResolveChoice Class
- **Machine Learning Transformations:**
 - FindMatches ML: could be used to find matching data which enables you to find related products, customers, and places.
 - FindMatches ML could be used to find duplicate customers who have signed up more than once.

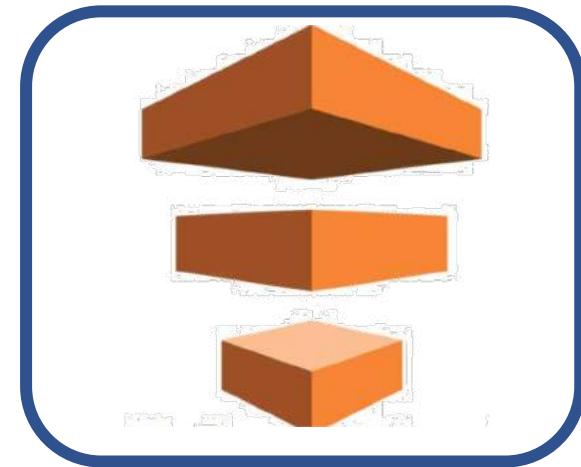


AWS DATA PIPELINE



AWS DATA PIPELINE

- AWS Data Pipeline is an **orchestration service** that allows AWS users to transfer data reliably and securely between various AWS compute and storage services.
- You can transfer data to Amazon S3, Amazon RDS, Amazon DynamoDB, and Amazon EMR.
- AWS manages the workflow so minimum maintenance is required.
- AWS guarantees resource availability and handles failures.



**AMAZON DATA
PIPELINE**



AWS DATA PIPELINE: EXAMPLE #1

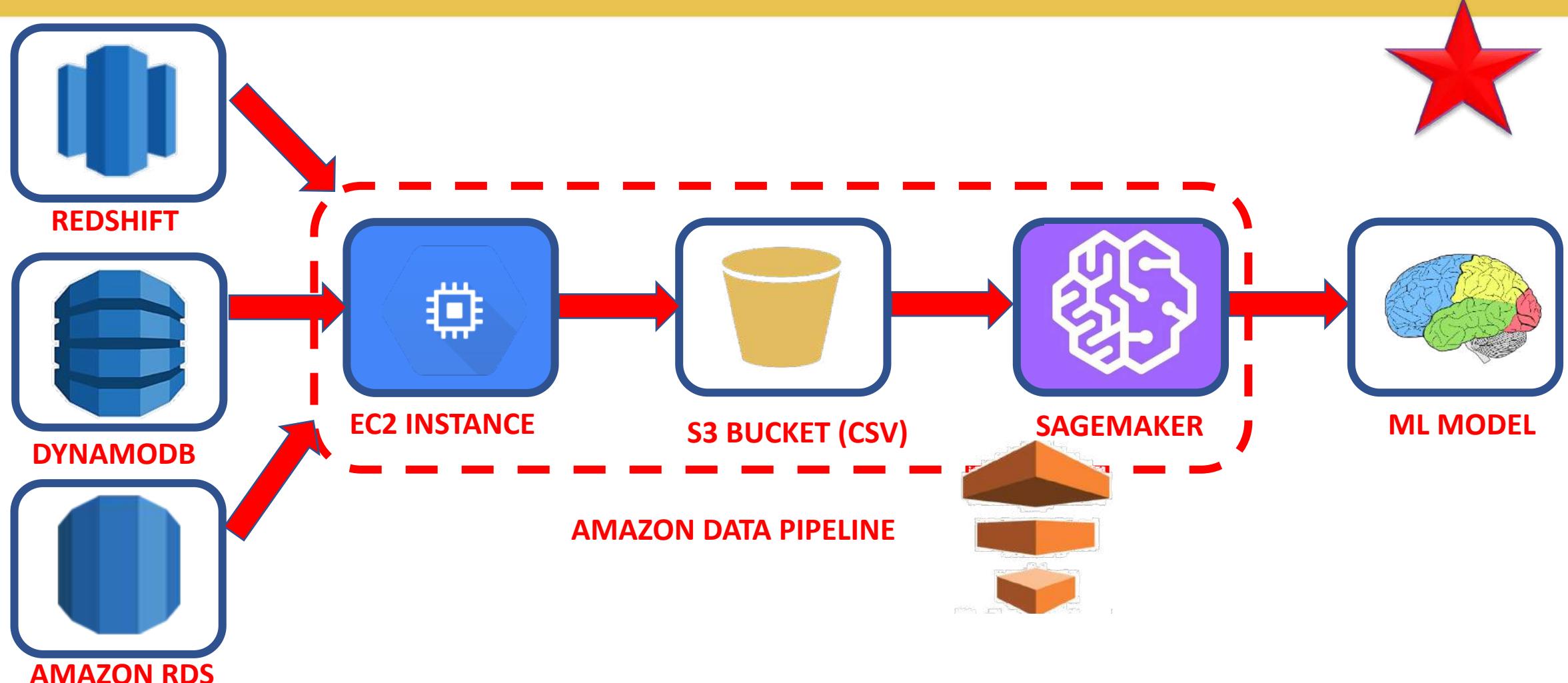


Photo Credit: https://en.wikipedia.org/wiki/File:User_icon_2.svg

<https://pixabay.com/illustrations/brain-lobes-neurology-human-body-1007686/>

<https://en.wikipedia.org/wiki/File:Google-Compute-Engine-Logo.svg>

AWS DATA PIPELINE: EXAMPLE#2

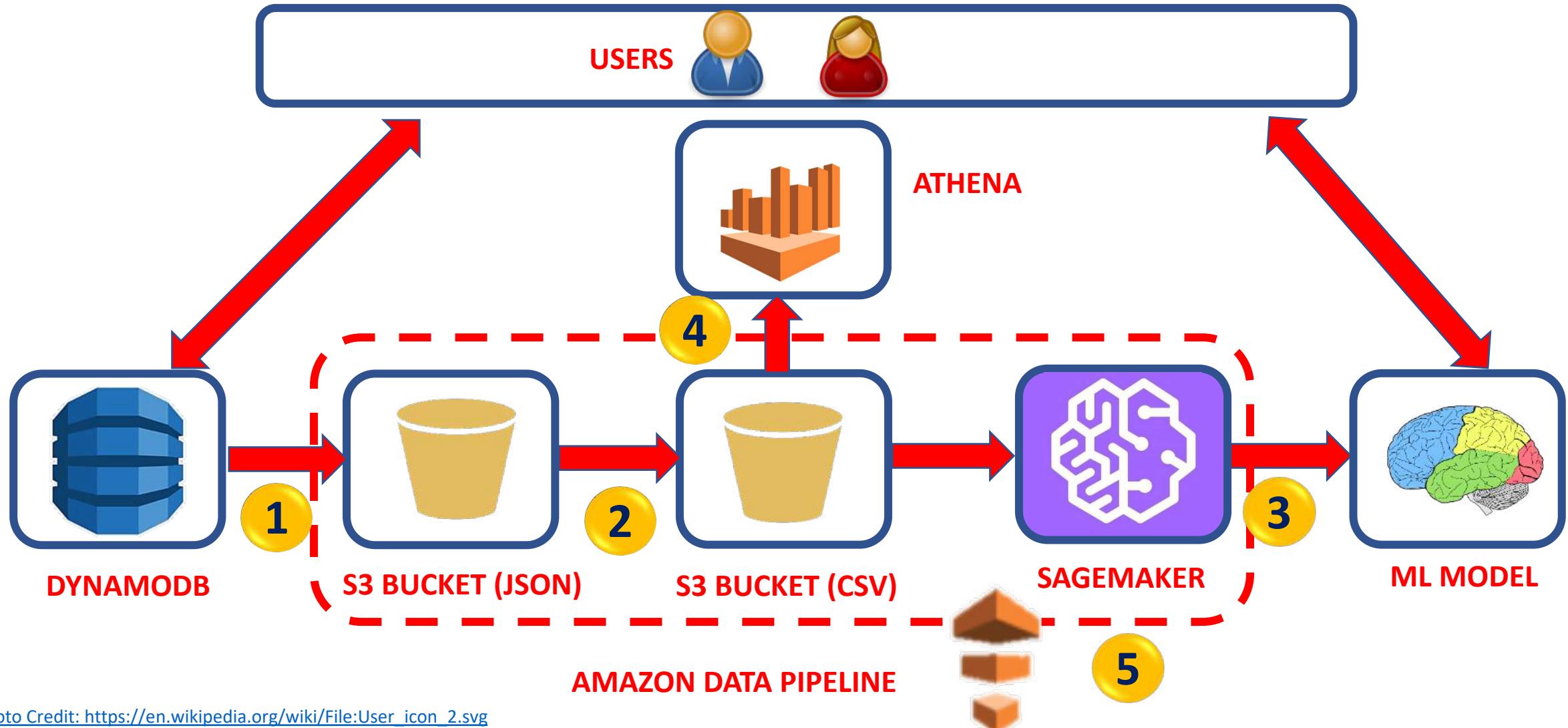
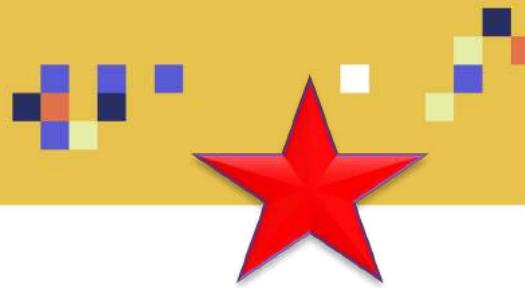


Photo Credit: https://en.wikipedia.org/wiki/File:User_icon_2.svg

<https://pixabay.com/illustrations/brain-lobes-neurology-human-body-1007686/>

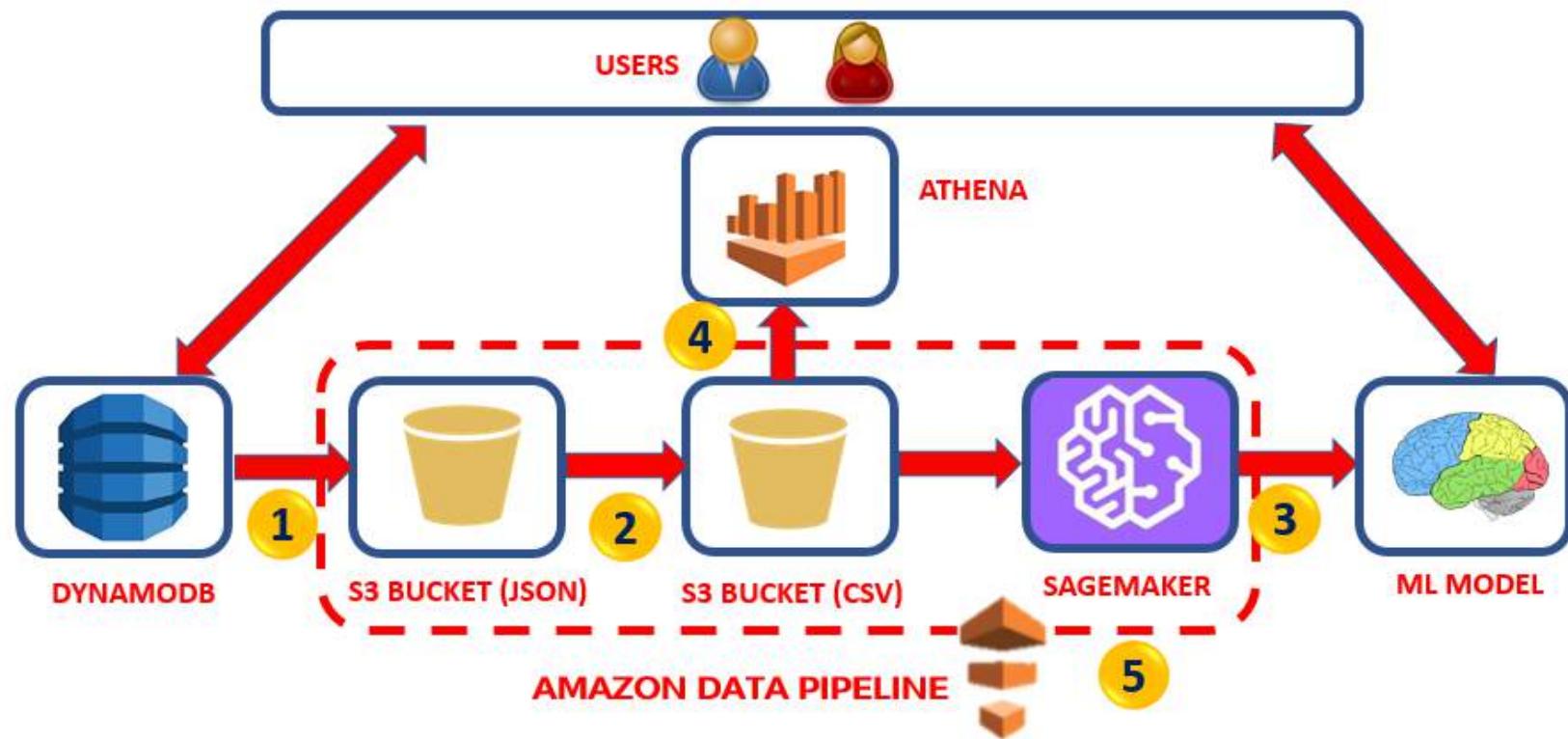
<https://aws.amazon.com/blogs/big-data/analyze-data-in-amazon-dynamodb-using-amazon-sagemaker-for-real-time-prediction/>

AWS DATA PIPELINE: EXAMPLE



The steps are as follows:

1. Data in DynamoDB is copied to Amazon S3 bucket (JSON) continuously as part of the Data Pipeline.
2. JSON files are then converted to CSV format to be used by Amazon SageMaker.
3. Amazon SageMaker consumes the data, trains the model, update endpoint for inference.
4. The data in CSV format is being queried (ad hoc) by Amazon Athena.
5. Data Pipeline manages the workflow as per the customer requirements.



WHEN SHOULD I USE AWS GLUE VS. DATA PIPELINE?



AWS GLUE:

- Glue is an ETL service that runs on a serverless Apache Spark environment.
- As a user, you do not have to configure or manage resources.
- Glue contains a data catalogue for ETL that could be used with Athena and Redshift Spectrum.
- AWS Glue ETL jobs uses Scala or Python.

AWS DATA PIPELINE:

- AWS Data Pipeline is a managed orchestration service.
- AWS Data pipeline offers more flexibility over EC2 instances that runs your code.
- If you are not using Apache Spark, its recommended to use AWS data pipeline.



AWS PIPELINE TEMPLATES



READYLY AVAILABLE TEMPLATES FOR
REDSHIFT, RDS, AND DYNAMODB

The screenshot shows the AWS Data Pipeline landing page. At the top, there's a navigation bar with a bell icon and the text "Ry". Below it is a large circular icon with a stylized orange and black design. The main heading is "AWS Data Pipeline". A subtext explains: "AWS Data Pipeline helps you move, integrate, and process data across AWS compute and storage resources, as well as your on-premises resources. AWS Data Pipeline supports integration of data and activities across multiple AWS regions." A blue "Get started now" button is visible. Below this, there are three sections with icons: "Define Data Nodes" (database icon), "Schedule Compute Activities" (monitor icon), and "Activate & Monitor" (person icon). Each section has a brief description and a "Learn more" link.

Create Pipeline

You can create pipeline using a template or build one using the Architect page.

Name:

Description (optional):

Source: Build using a template

Choose...
Choose...
 Getting Started
Getting Started using ShellCommandActivity
AWS Command Line Interface (CLI) Templates
Run AWS CLI command
 DynamoDB Templates
Export DynamoDB table to S3
Import DynamoDB backup data from S3
 Elastic MapReduce (EMR) Templates
Run job on an Elastic MapReduce cluster
 RDS Templates
Full copy of RDS MySQL table to S3
Incremental copy of RDS MySQL table to S3
Load S3 data into RDS MySQL table
 Redshift Templates
Full copy of RDS MySQL table to Redshift
Incremental copy of RDS MySQL table to Redshift
Load data from S3 into Redshift

Schedule

Run: Run every Starting Ending

Run every: after occurrence(s)

Starting: 2019-12-01 UTC (Current time is 00:50 UTC)
YYYY-MM-DD HH:MM

Ending: after occurrence(s)

Pipeline Configuration

Logging: Enabled Disabled

Copy execution log to clipboard

The screenshot shows the AWS Data Pipeline landing page. At the top, there's a navigation bar with a bell icon and the text "Ry". Below it is a large circular icon with a stylized orange and black design. The main heading is "AWS Data Pipeline". A subtext explains: "AWS Data Pipeline helps you move, integrate, and process data across AWS compute and storage resources, as well as your on-premises resources. AWS Data Pipeline supports integration of data and activities across multiple AWS regions." A blue "Get started now" button is visible. Below this, there are three sections with icons: "Define Data Nodes" (database icon), "Schedule Compute Activities" (monitor icon), and "Activate & Monitor" (person icon). Each section has a brief description and a "Learn more" link.

Pipeline Configuration

Logging: Enabled Disabled

Copy execution log to clipboard

AWS DATA MIGRATION SERVICE (DMS)



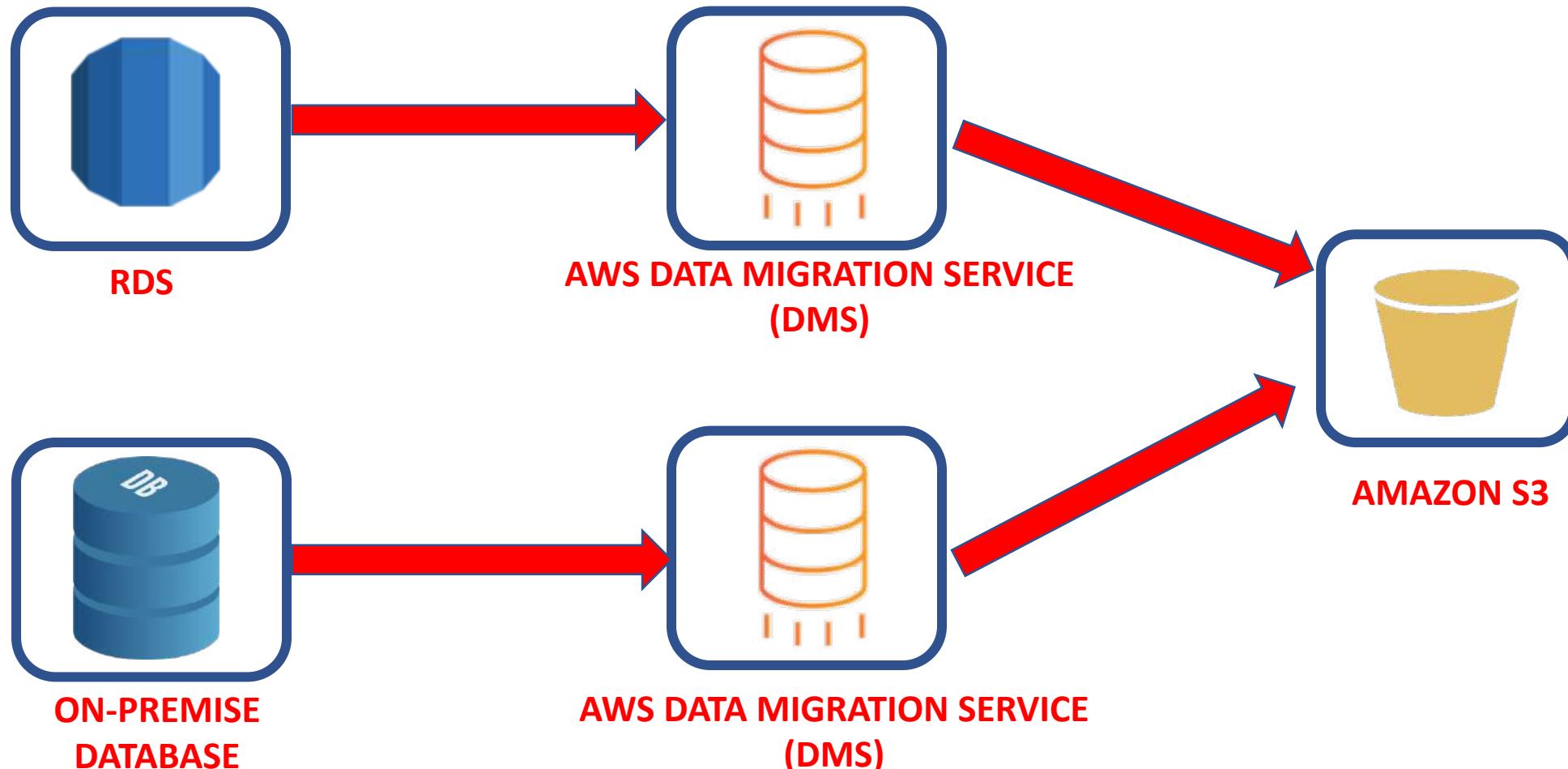
AWS DATA MIGRATION SERVICE (DMS)

- As the name stated, AWS Database Migration Service allows for database migration to AWS quickly.
- Source database remains functional during migration.
- Oracle to oracle migration or oracle to Aurora (RDS).
- Allows for database consolidation.
- Allows for data replication in a Data warehouse such as Amazon Redshift and S3.
- Watch Video: <https://aws.amazon.com/dms/>



**AWS DATA MIGRATION
SERVICE (DMS)**

AWS DATA MIGRATION SERVICE (DMS)



AWS DATA MIGRATION SERVICE (DMS)



SELECT THE SOURCE AND ENDPOINT

AWS Database Migration Service
Migrate your databases to AWS with minimal downtime

The quickest and easiest way to migrate databases to AWS with low cost and minimal downtime.

How it works

AWS Database Migration Service helps you migrate databases to AWS quickly and securely. The source database remains fully operational during the migration, minimizing downtime to applications that rely on the database. The AWS Database Migration Service can migrate your data to and from most widely used commercial and open-source databases.

Use cases [Learn more](#)

Endpoint type [Info](#)

Source endpoint
A source endpoint allows AWS DMS to read data from a database (on-premises or in the cloud), or from other data source such as Amazon S3.

Target endpoint
A target endpoint allows AWS DMS to write data to a database, or to other data source.

Select RDS DB instance

Endpoint configuration

Endpoint identifier [Info](#)
A label for the endpoint to help you identify it.
ProdEndpoint

Source engine
The type of database engine this endpoint is connected to.
Choose an engine

Q |
aurora
aurora-postgresql
s3
db2
mariadb
azuredb
sqlserver
mongodb
mysql
oracle

WHEN SHOULD I USE AWS GLUE VS. AWS MIGRATION SERVICE?

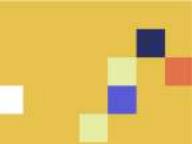


AWS GLUE:

- Glue is an ETL service that runs on a serverless Apache Spark environment.
- As a user, you do not have to configure or manage resources.
- Glue contains a data catalogue for ETL that could be used with Athena and Redshift Spectrum.
- AWS Glue ETL jobs uses Scala or Python.

AWS MIGRATION (DMS):

- AWS DMS allows for data migration to AWS safely and cheaply.
- AWS DMS is recommended when database migration is needed to AWS.
- Once data is available in AWS, AWS Glue could be used to transform/move data source to data warehouse such as Amazon Redshift.



AWS BATCH



AWS BATCH

- AWS Batch allows for running of batch computing jobs on AWS.
- AWS batch optimizes the type and number of compute resources based on volume.
- No management or hassle (serverless), AWS takes care of the batch computing software and resources.
- AWS Batch performs all the scheduling and execution of the batch using Amazon EC2 and Spot Instances.
- Users pay for the EC2 resources that the batch run on.
- AWS Batch jobs could be scheduled using CloudWatch Events
- AWS batch jobs could be orchestrated using AWS Step Functions



AWS BATCH

WHEN SHOULD I USE AWS GLUE VS. BATCH?



AWS GLUE:

- Glue is an ETL service that runs on a serverless Apache Spark environment.
- As a user, you do not have to configure or manage resources.
- Glue contains a data catalogue for ETL that could be used with Athena and Redshift Spectrum.
- AWS Glue ETL jobs uses Scala or Python.

AWS BATCH:

- Regardless of the type of the job, AWS Batch enables running of batch computing jobs on AWS.
- AWS Batch creates and manages the compute resources in the AWS account.
- AWS batch gives users full visibility and control over resources (EC2 instances).

For ETL use cases, use AWS Glue

For any other batch services (that might include ETL), use AWS Batch

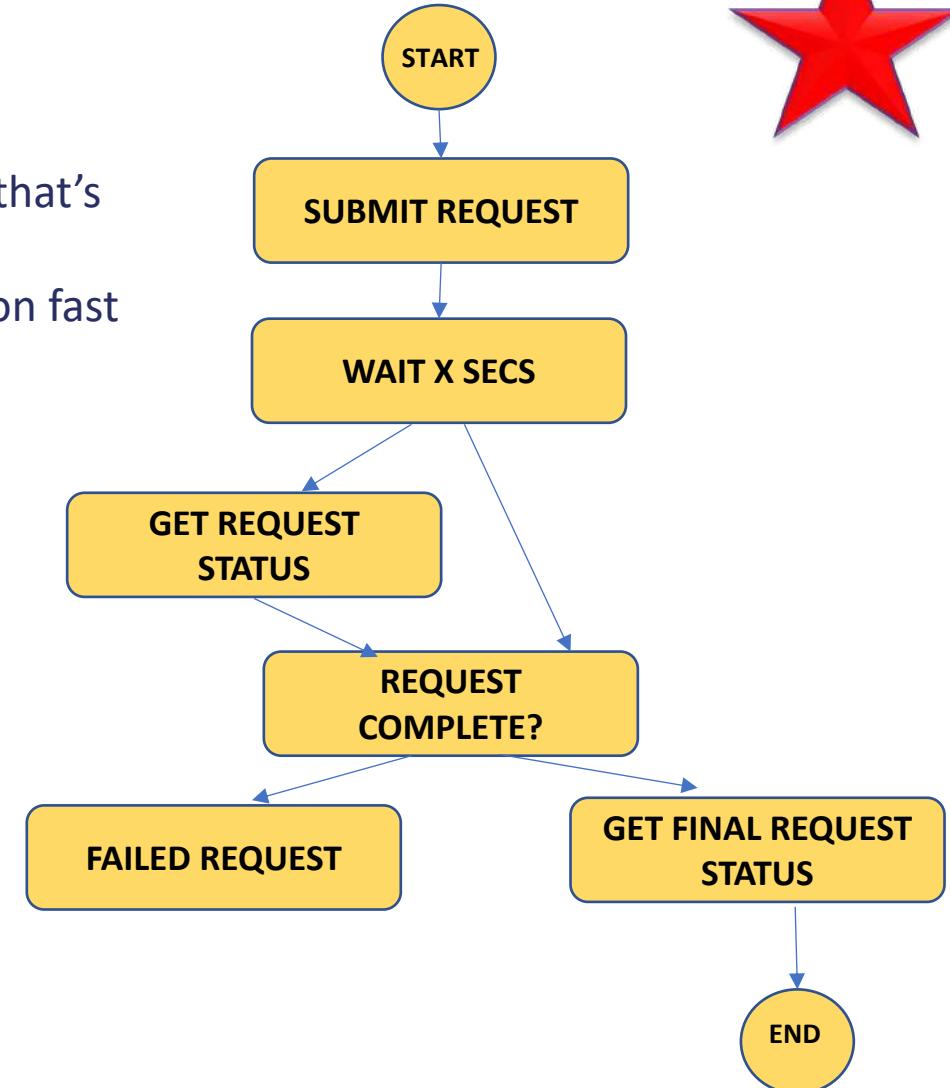


AWS STEP FUNCTION

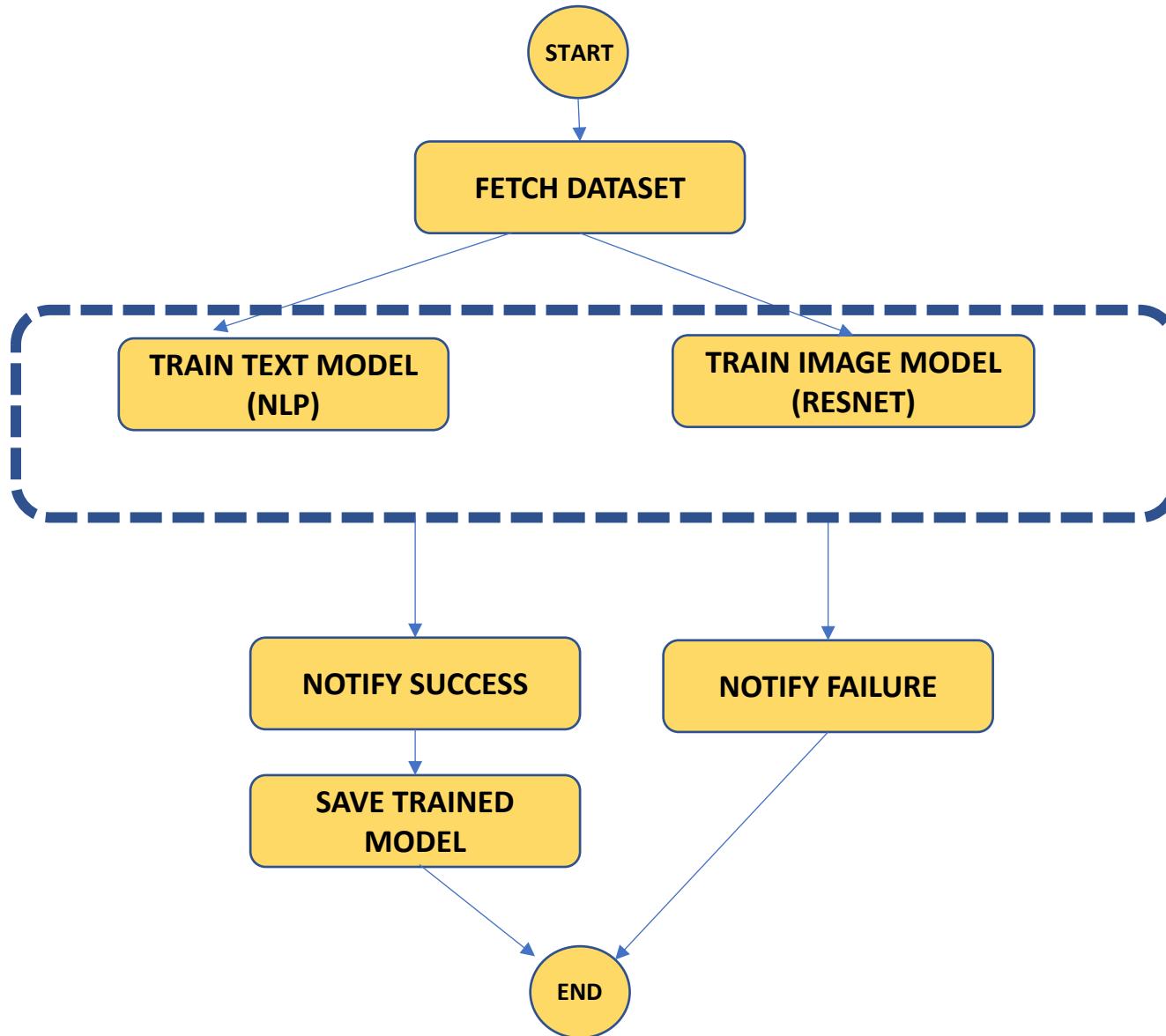


AMAZON AWS STEP FUNCTIONS

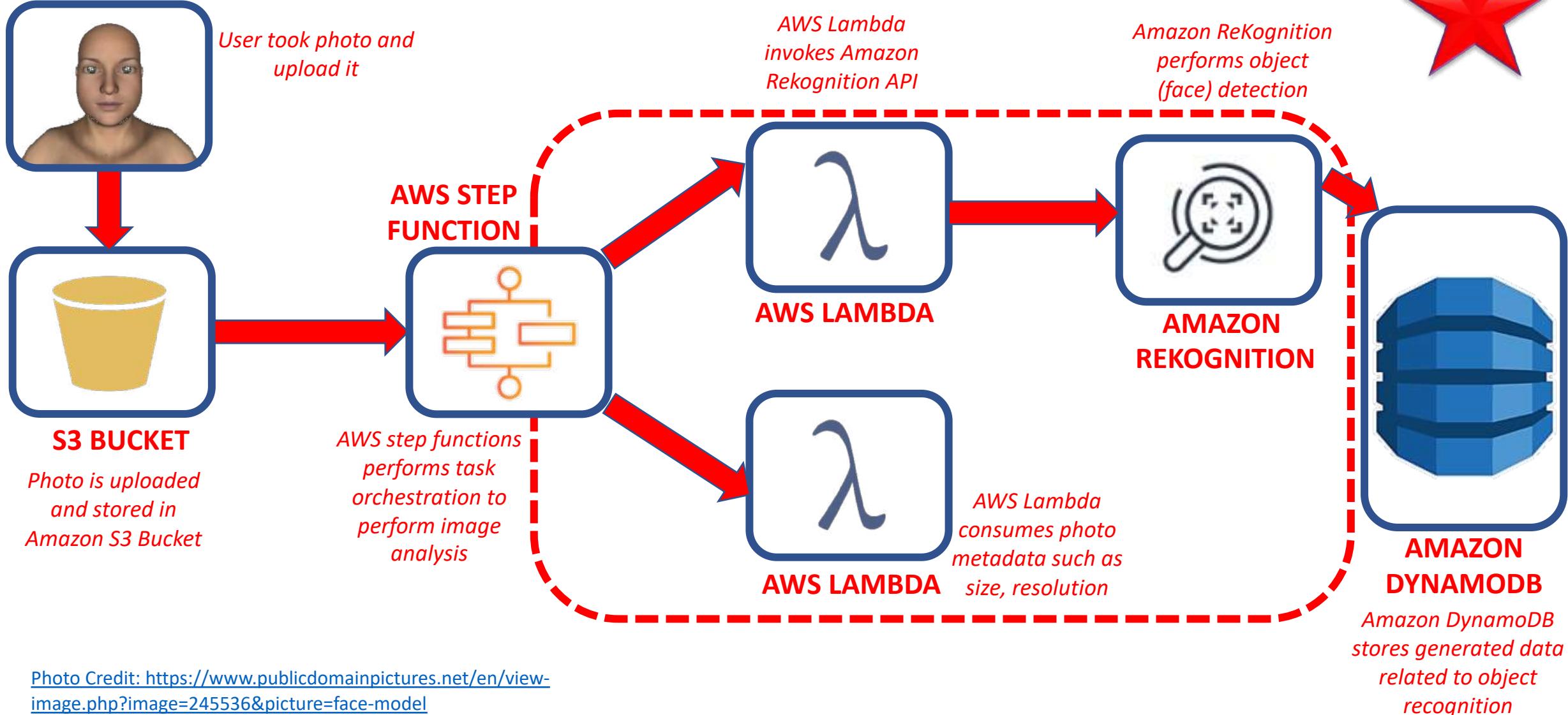
- AWS Step Functions allows for creating serverless workflows.
- Output from a step is fed as an input to the next step.
- AWS Step functions converts a workflow into a state machine diagram that's easy to debug and understand.
- AWS Step Functions allows for performing resilient workflow automation fast without writing code.
- It allows for advanced error handling and retrying mechanisms.
- Watch Video: <https://aws.amazon.com/step-functions/getting-started/>



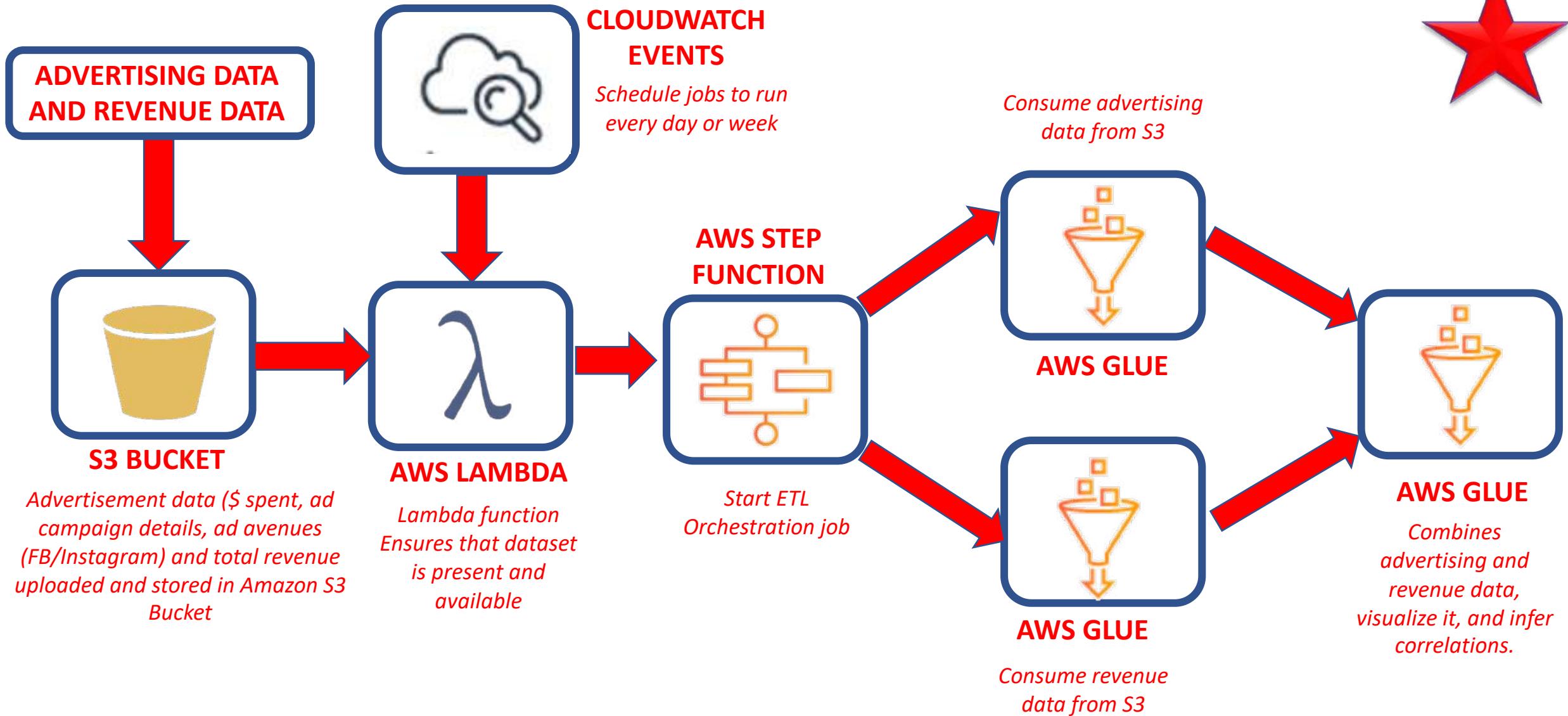
AMAZON AWS STEP FUNCTIONS: MACHINE LEARNING EXAMPLE



AWS STEP FUNCTION: EXAMPLE #1



AWS STEP FUNCTION: EXAMPLE #2



AWS MACHINE LEARNING CERTIFICATION



DOMAIN #1: DATA ENGINEERING (20% EXAM)



DATA STREAMING



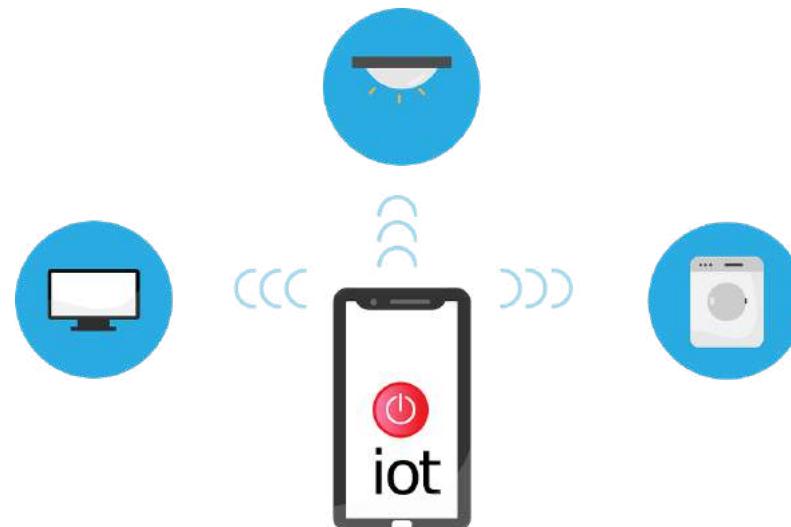
HOW TO INGEST AND ANALYZE STREAMING DATA?



- Streaming Data can come from so many sources such as clickstreams, IOT devices, stock data..etc.
- Collecting, structuring and analysing this data is critical for companies to gain customers insights and set their marketing and product strategies.
- Data could arrive in real-time and gaining valuable insights from it in real-time as well is crucial.



STOCK DATA



INTERNET OF THINGS (IOT) DEVICES

Photo Credit: <https://pixabay.com/illustrations/internet-of-things-iot-network-3671222/>

Photo Credit: <https://www.pexels.com/photo/blue-and-yellow-graph-on-stock-market-monitor-159888/>



AWS ML CERTIFICATION EXAM DOMAINS



Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #1: WHERE ARE WE NOW!!?



SECTION #1: INTRODUCTION, DATA/ML LINGO, AWS DATA STORAGE

- What is Machine Learning and Artificial Intelligence?
- What is Amazon Web Services (AWS)?
- Artificial Intelligence and Machine learning Lingo (data types, Labeled vs. unlabeled, sagemaker groundtruth)
- structured vs. unstructured and database vs. data lake vs. data storage
- AWS Data Storage (Redshift, RDS, S3, DynamoDB)

SECTION #2: AMAZON S3

- Amazon S3 in Depth (partitions, tags)
- Amazon S3 Storage Tiers and Lifecycles
- Amazon S3 Encryption and Security
- Amazon S3 Encryption and Security – Part #2 (ACL, CloudWatch, CloudTrail, VPC)
- Additional Notes (Elasticsearch, ElastiCache, and Database vs. data warehouse)



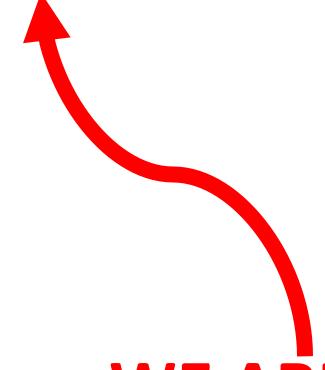
DOMAIN #1 OVERVIEW:

SECTION #3: AWS DATA MIGRATION, GLUE, PIPELINE, STEP AND BATCH

- AWS Glue (crawlers, features, built-in transformations etc)
- AWS Data pipeline
- AWS Data Migration Service (DMS)
- AWS Batch
- Step Function

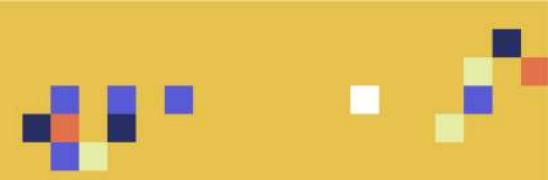
SECTION #4: DATA STREAMING & KINESIS

- Kinesis Overview
- Kinesis Video Streams
- Kinesis Data Streams
- Kinesis Firehose
- Kinesis Analytics and Random Cut Forest



WE ARE HERE!

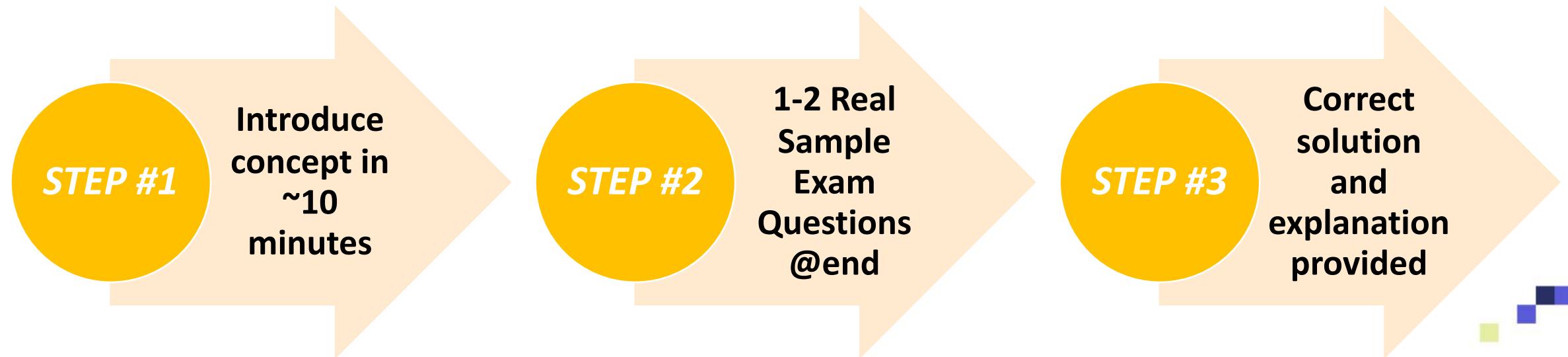
LECTURE DESIGN



- We know how hard it is to study for an exam especially if you have a busy schedule.
- This course is designed to be extremely on point and optimized to pass the exam.

No boring content. Zero unnecessary information.

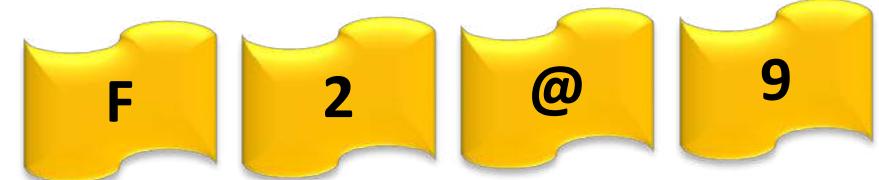
- Here's the lecture structure that we will follow:



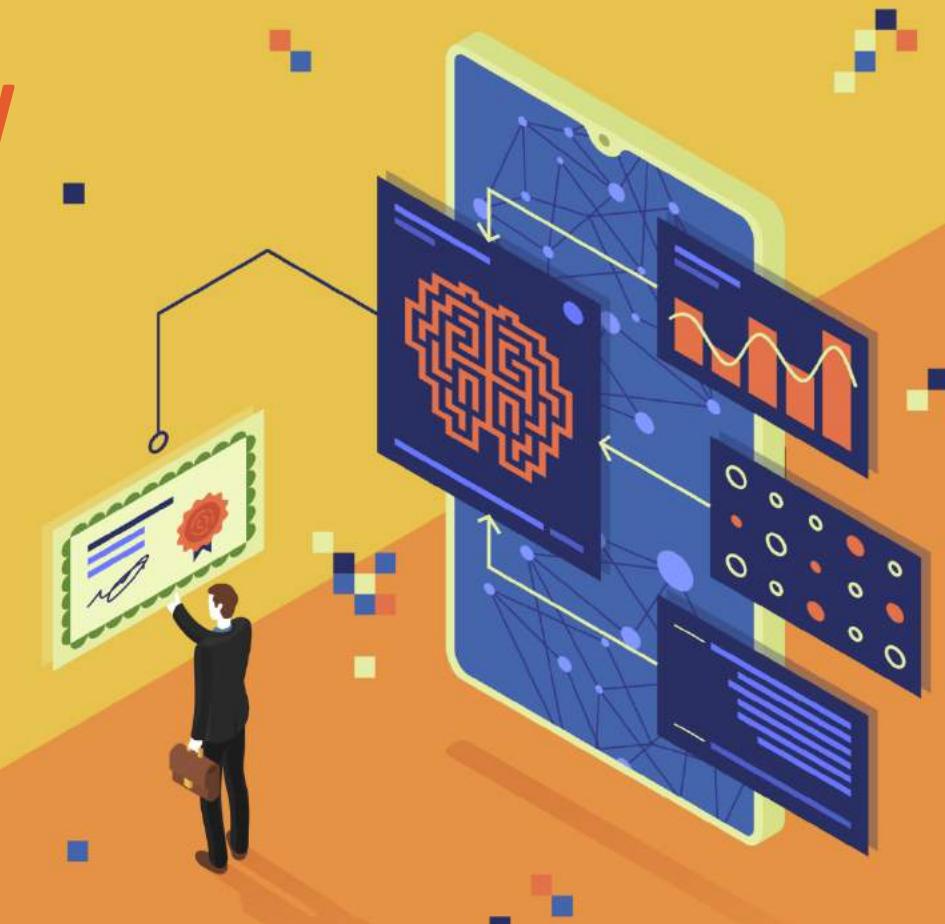
RECALL OUR MINI CHALLENGE AND PRIZE!



- For those of you who will successfully complete the entire section and watch the videos till the end, they will receive a valuable prize!



AWS KINESIS OVERVIEW



AWS KINESIS



- Amazon Kinesis enable enterprises/individuals to consume and analyze streaming data in real-time.
- Kinesis is cost optimized.
- This feature is critical for some enterprises who want to get information quickly such as stock traders.
- Real-time data is consumed and processed by Kinesis in Realtime such as video, audio, application logs, website clickstreams.
- Data analytics and machine learning techniques could be applied to this real-time data to get valuable quick insights.
- Kinesis is a managed alternative to Apache Kafka.

Real-time

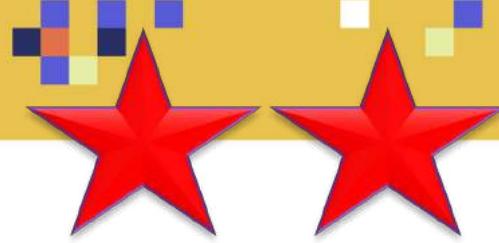
Consume and process data in real-time to get valuable insights in minutes.

Scalable

Process extremely large amount of data at great speed.

Fully managed

Fully managed service without the need to configure servers or compute clusters.



1. Amazon Kinesis Video Streams

- Video streaming service from connected devices to AWS.
- Data is processed using machine learning (ML)/data analytics.

2. Amazon Kinesis Data Streams

- Real-time data streaming service.
- Capture gigabytes of data per second from hundreds of thousands of sources.

3. Amazon Kinesis Data Firehose

- Near real-time service for capturing and loading data streams into AWS.
- Near real-time analytics with existing business intelligence tools.

4. Amazon Kinesis Data Analytics

- Data Analytics and processing service with SQL or Java.
- No need to know programming or setup any frameworks.

AWS KINESIS



<https://aws.amazon.com/kinesis>

aws Services Resource Groups

Get started with Amazon Kinesis

To get started, choose an Amazon Kinesis resource to create.

Ingest and process streaming data with Kinesis streams

Process data with your own applications, or using AWS managed services like Amazon Kinesis Data Firehose, Amazon Kinesis Data Analytics, or AWS Lambda.

[Create data stream](#)

Deliver streaming data with Kinesis Firehose delivery streams

Continuously collect, transform, and load streaming data into destinations such as Amazon S3 and Amazon Redshift.

[Create delivery stream](#)

Analyze streaming data with Kinesis analytics applications

Run continuous analysis on streaming data from Kinesis data streams and Kinesis Firehose delivery streams.

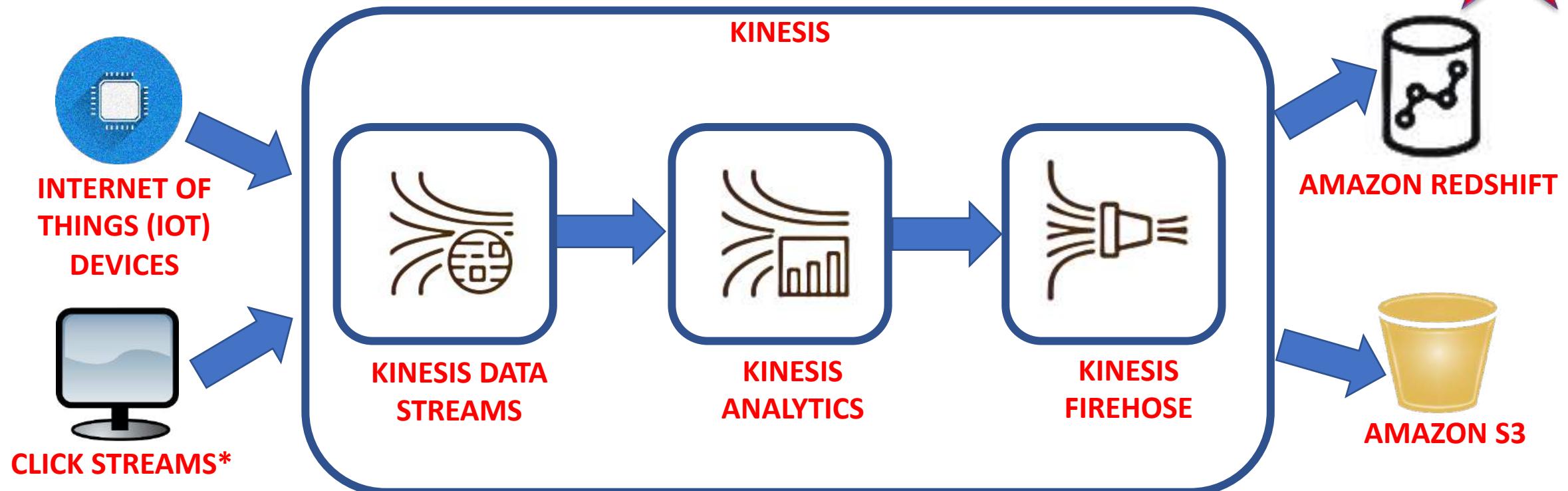
[Create analytics application](#)

Ingest and process media streams with Kinesis video streams

Build applications to process or analyze streaming media.

[Create video stream](#)

AWS KINESIS



* *clickstream* is a summary of customers activity including websites, clicks, time spent on website..

Photo Credit: <https://pixabay.com/vectors/cpu-processor-computer-electronics-2103856/>

Photo Credit: https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg

Photo Credit: <https://freesvg.org/computer-flat-monitor-symbol-vector-illustration>

AWS KINESIS VIDEO STREAMS – PART #1



1. KINESIS VIDEO STREAMS



- Amazon Kinesis Video Streams allows for seamless video streaming from millions of devices.
- Feed this video stream to computer vision/Deep Learning algorithms (ex: face detection).
- Kinesis video streams is extremely elastic and automatically scales for any number of devices.
- As always, pay per use model and data is kept for 1 hour and up to 10 years.
- Kinesis video streams is secure since it durably encrypts the video streams and store it.
- It is super easy to use with powerful APIs
- Connect Kinesis video streams to OpenCV, Amazon Rekognition, Apache MXNet, TensorFlow.
- You can setup Kinesis video streams quickly and easily from the AWS main console. Then, on your device (mobile phone), download Kinesis video streams SDK, and voila! You have access to secure data to store and run analytics on!
- You can integrate with Kinesis video streams with AWS DeepLens and Realtime Streaming Protocol (RTSP) camera



Photo Credit: <https://pixabay.com/vectors/surveillance-camera-security-video-147831/>

Photo Credit: <https://commons.wikimedia.org/wiki/File:Mobile-Smartphone-icon.png>



1. KINESIS VIDEO STREAMS: UNIQUE FEATURES



VIDEO STREAMING FROM NUMEROUS EDGE DEVICES

- Kinesis video streams offer easy to use SDKs to allow easy integration with edge device.
- Data could be ingested from cameras, mobile phones, satellites, LiDARs.

DEVELOP COMPUTER VISION-BASED APPS

- Kinesis video streams allow for seamless integration with Amazon Rekognition.
- It can also be integrated with other frameworks such as TensorFlow, MXNet, OpenCV.

EASILY PLAYBACK VIDEOS

- Kinesis video streams offer HTTP live streaming (HLS) service. You can play recorded videos or play live stream on any browser.



1. KINESIS VIDEO STREAMS: UNIQUE FEATURES



ELEVATED SECURITY

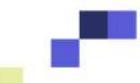
- Amazon Kinesis Video Streams automatically encrypt data both in transit and at rest using:
 - **AT REST:** using AWS Key Management Service (KMS)
 - **IN TRANSIT:** Using industry-standard Transport Layer Security (TLS) protocol.
- Access management using AWS Identity and Access Management (IAM).

HIGH DURABILITY

- Amazon Kinesis Video Streams relies on Amazon S3 for storage so you're in good hands!

ZERO HASSLE/NO SERVERS TO MANAGE

- No maintenance, software update management and hassle! Amazon Kinesis Video Streams manages the entire infrastructure.



AWS KINESIS VIDEO STREAMS – PART #2



1. KINESIS VIDEO STREAMS: HOW DOES IT WORK?



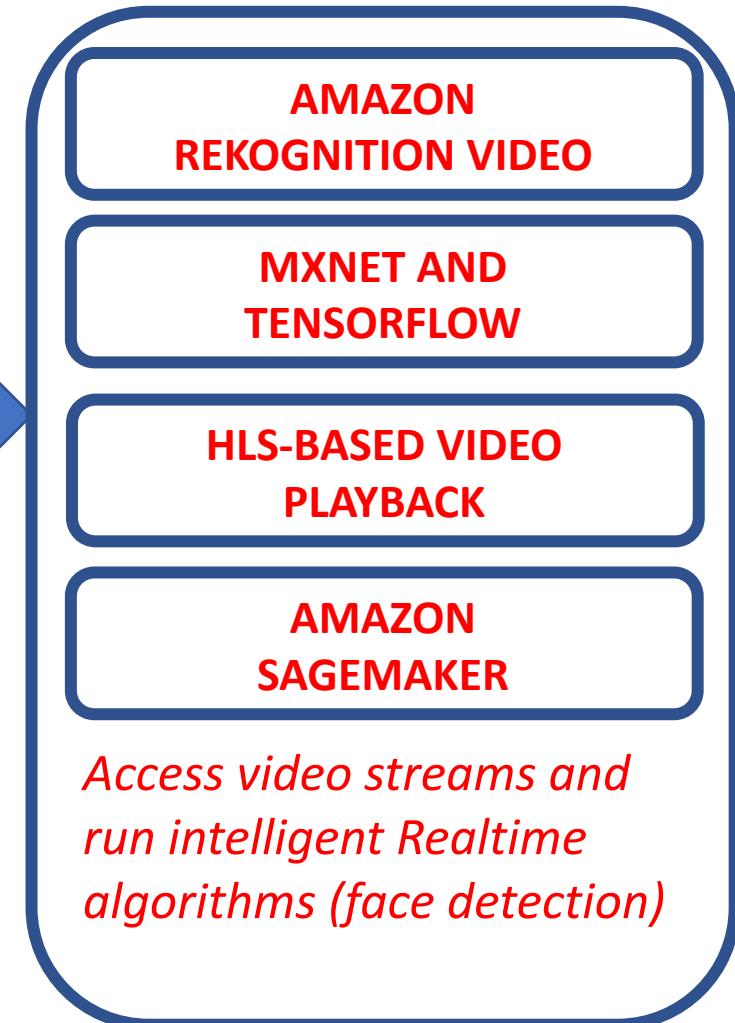
VIDEO SOURCES

Using Kinesis video streams SDK, multiple devices can stream videos in real-time

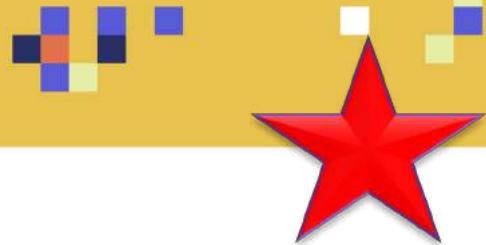


KINESIS VIDEO STREAMS

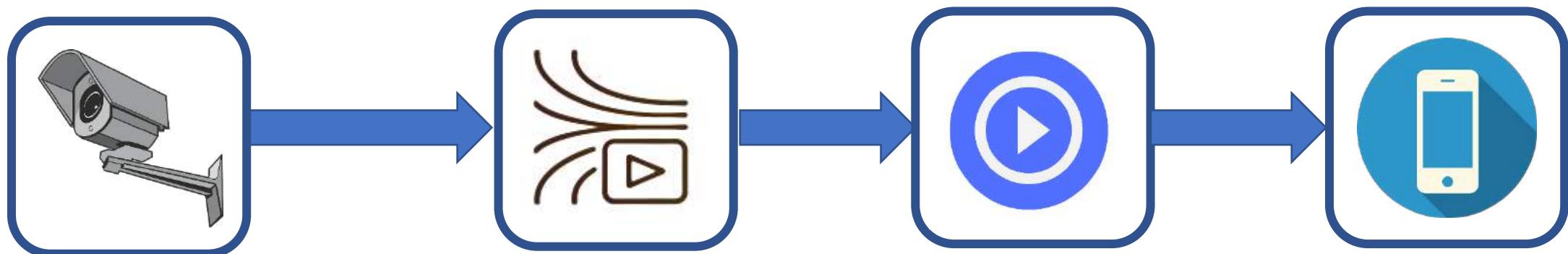
Durably and securely ingests and stores video streams in Realtime



1. KINESIS VIDEO STREAMS: USE CASE #1



- You can use Amazon Kinesis video streams to develop a smart home
- By easily installing video cameras such as baby video cameras, outside home surveillance cameras, and pet cameras.
- You can then ingest these video streams and record it, play it live on your device, or even run deep learning algorithms to detect certain people.
- Happy House Problem cannot be made easier! Only smiling people are allowed in the house!
<https://www.kaggle.com/iarunava/happy-house-dataset>



**PET MONITORING
CAMERA**

*Using Kinesis video streams SDK,
stream pet video*

**KINESIS VIDEO
STREAMS**

*Durably and securely ingests and stores
video streams in Realtime*

**VIDEO
PROCESSING APP**

*Play back video using HTP
Live Streaming (HLS)*

SMART PHONE

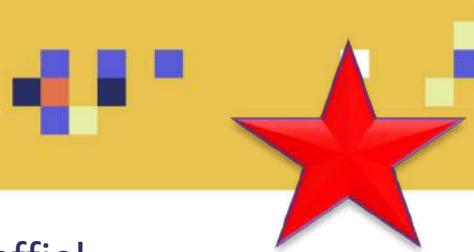
*Watch your favourite pet
and send Deep Learning-
based alarms*

Photo Credit: <https://pixabay.com/vectors/surveillance-camera-security-video-147831/>

Photo Credit: <https://commons.wikimedia.org/wiki/File:Mobile-Smartphone-icon.png>

<https://pixabay.com/illustrations/icon-play-video-movie-youtube-1968245/>

1. KINESIS VIDEO STREAMS: USE CASE #2



- Governments can use Amazon Kinesis to develop smart cities by reducing crime rates and solving traffic!
- By installing cameras in traffic lights and shopping malls.

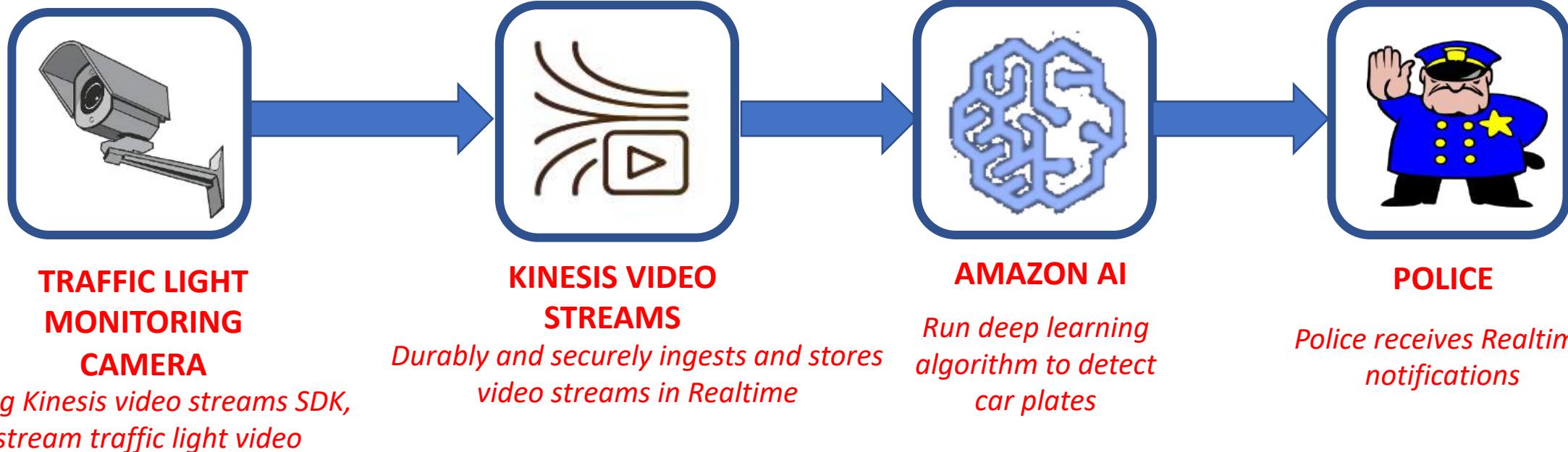
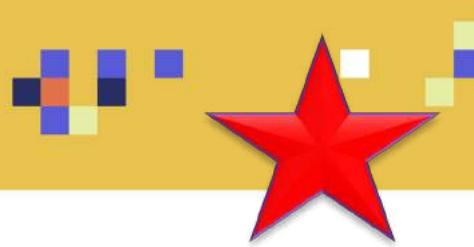
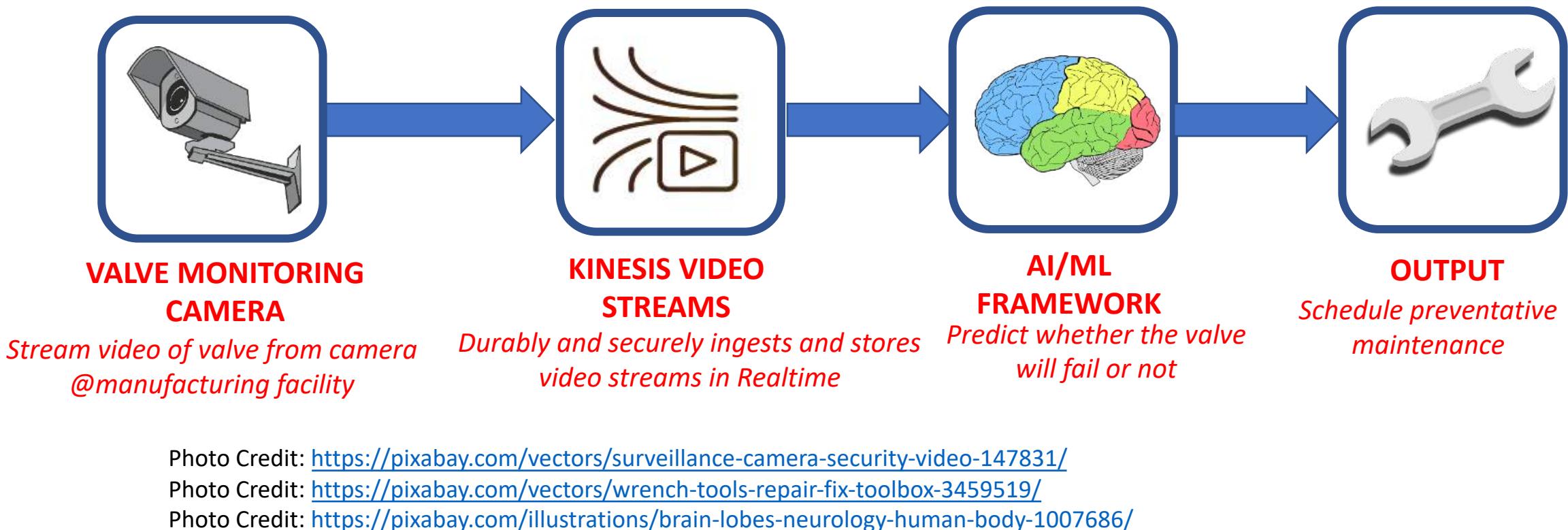


Photo Credit: <https://pixabay.com/vectors/surveillance-camera-security-video-147831/>
Photo Credit: <https://commons.wikimedia.org/wiki/File:Mobile-Smartphone-icon.png>
<https://pixabay.com/illustrations/icon-play-video-movie-youtube-1968245/>
<https://publicdomainvectors.org/en/free-clipart/Police-man-vector-drawing/7211.html>

1. KINESIS VIDEO STREAMS: USE CASE #3

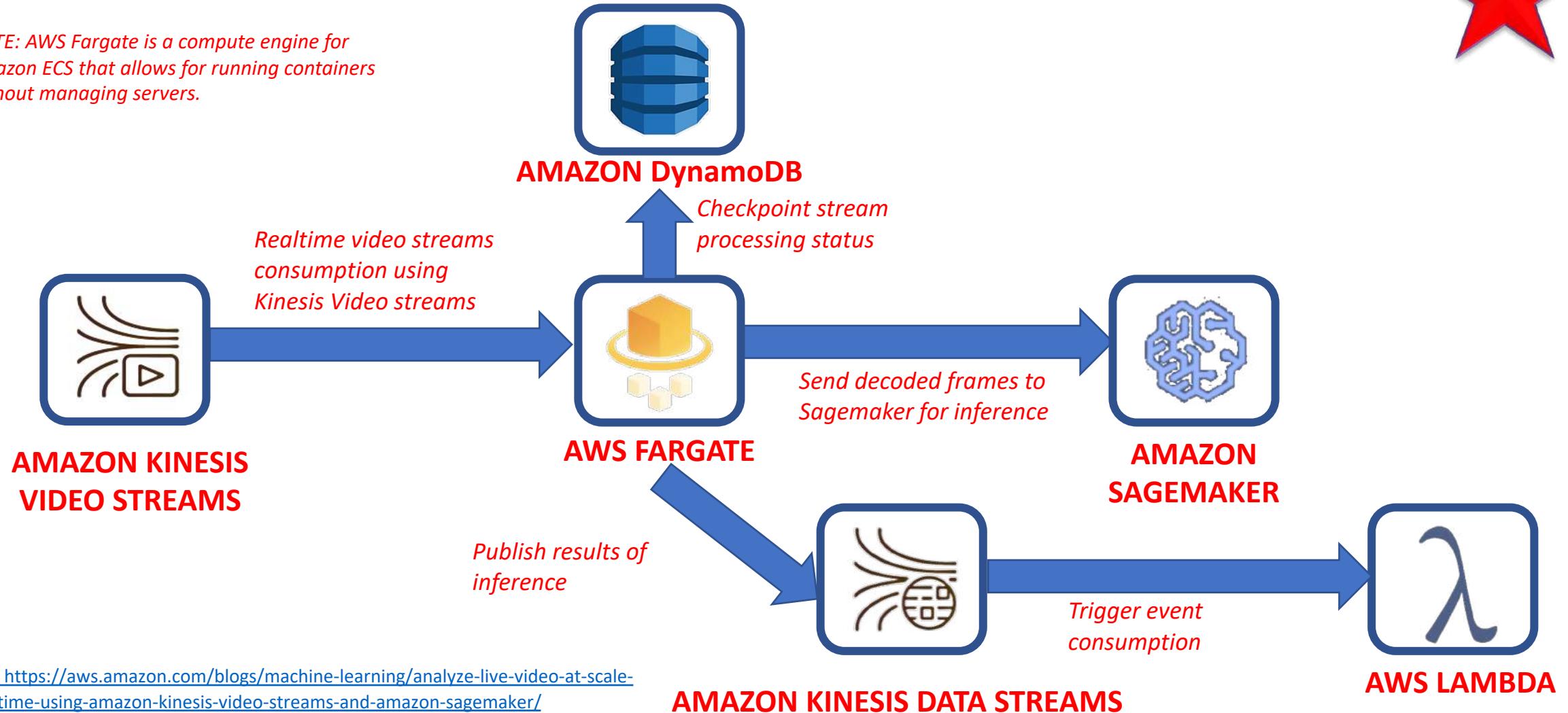


- You can use Amazon Kinesis video streams to develop a smart preventative maintenance system in factories.
- For example: you can use Amazon Kinesis Video Streams to ingest data from radars, Lidars and cameras and then run AI/ML algorithms using TensorFlow/MXNet to predict when a valve would fail. This is crucial to reduce downtime.



1. KINESIS VIDEO STREAMS: SAGEMAKER, DATASTREAMS, AWS LAMBDA, DYNAMODB INTEGRATION

NOTE: AWS Fargate is a compute engine for Amazon ECS that allows for running containers without managing servers.



Source: <https://aws.amazon.com/blogs/machine-learning/analyze-live-video-at-scale-in-real-time-using-amazon-kinesis-video-streams-and-amazon-sagemaker/>

Photo Credit: <https://en.wikipedia.org/wiki/File:Lambda-letter-lowercase-symbol-Garamond.svg>

AWS KINESIS DATA STREAMS – PART #1



2. KINESIS DATA STREAMS



- Amazon Kinesis Data Streams (KDS) is streaming service that works seamlessly in Realtime.
- You can ingest millions of data coming from various sources such as click streams, IOT, and several devices.
- Realtime analytics can be conducted on the data and displayed on a dashboard.



**KINESIS DATA
STREAMS**



[Photo Credit: https://www.needpix.com/photo/1539231/nyse-newyorkstockexchange-floor-business-commerce-trading-economy-people-newyork](https://www.needpix.com/photo/1539231/nyse-newyorkstockexchange-floor-business-commerce-trading-economy-people-newyork)



2. KINESIS DATA STREAMS



WORKS IN REAL-TIME

- Within 70 ms, let your streaming data ready for analysis (Amazon S3, AWS lambda) from multiple sources.

HIGH DATA DURABILITY

- Kinesis data streams ensures data durability by (1) data replication over 2 data centers, (2) 7 days data storage.

ELEVATED SECURITY

- Kinesis data streams offer elevated security by: (1) allowing KDS-enabled data encryption, (2) Amazon Virtual Private Cloud (VPC) privately accessed network, (3) server-side encryption and AWS KMS master keys.

SIMPLE

- Super easy and simple to build powerful and reliable streaming service by (1) integrating KDS with Kinesis Firehose, Kinesis data analytics, AWS lambda, (2) leveraging AWS SDK, Kinesis Client Library (KCL), connectors, and agents.

HIGH ELASTICITY

- Kinesis data streams is very elastic and can be scaled easily (scale from thousands to millions of PUT records per second)

OPTIMIZED (REDUCED) COST

- Zero infrastructure/upfront cost. Pay per use model.



2. KINESIS DATA STREAMS: HOW IT WORKS?

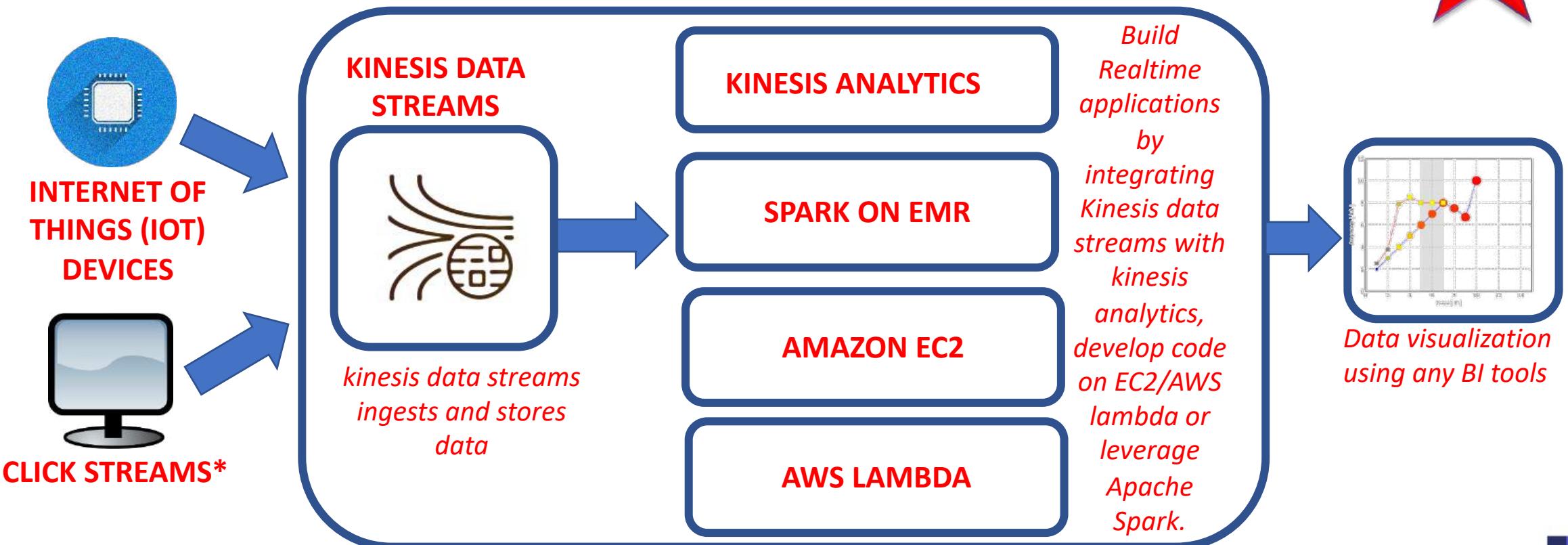


Photo Credit: <https://pixabay.com/vectors/cpu-processor-computer-electronics-2103856/>

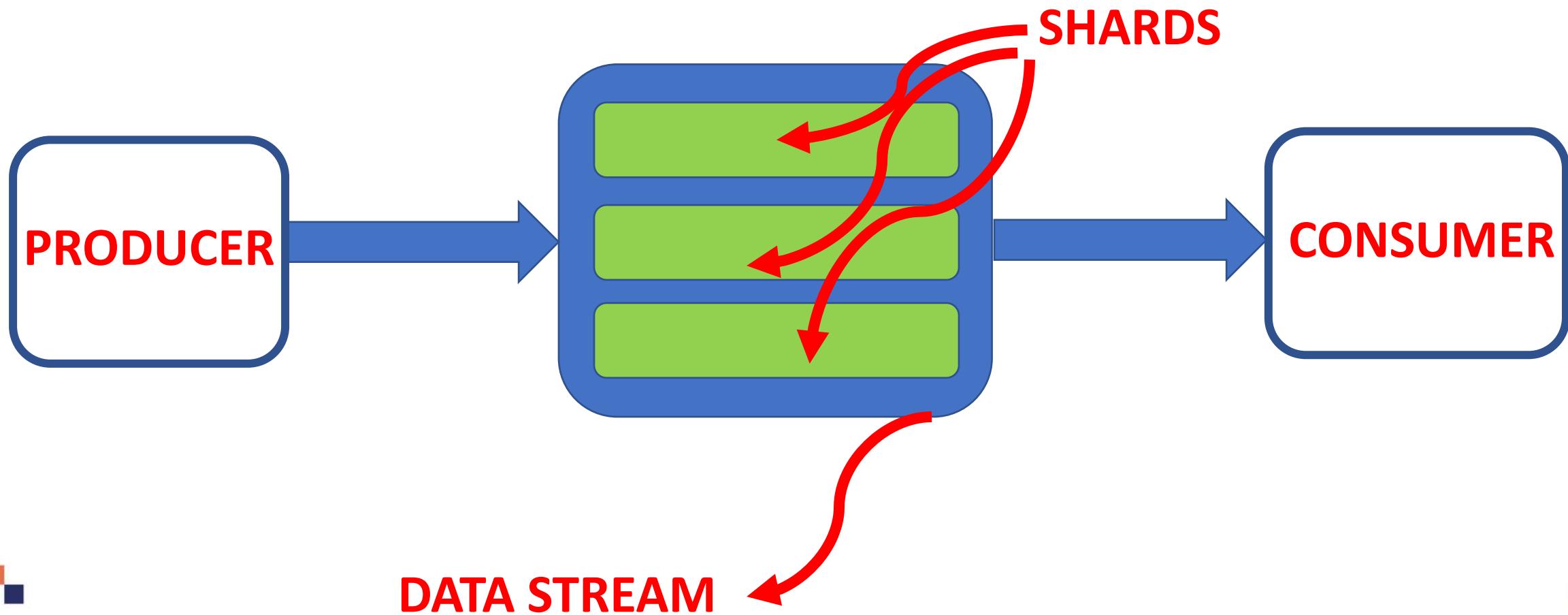
Photo Credit: https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg

Photo Credit: <https://freesvg.org/computer-flat-monitor-symbol-vector-illustration>

AWS KINESIS DATA STREAMS – PART #2



2. KINESIS DATA STREAMS: DEFINITIONS



2. KINESIS DATA STREAMS: DEFINITIONS



Data Producer:

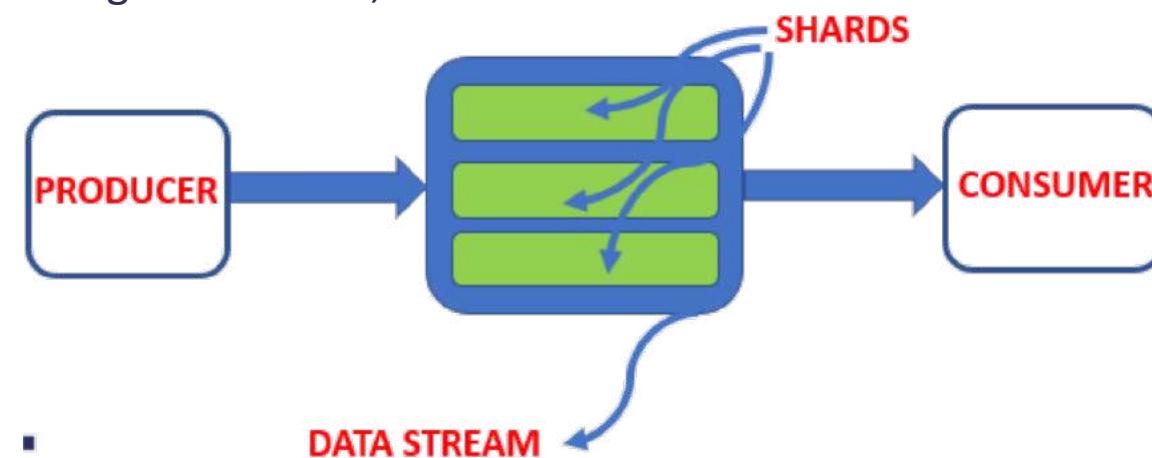
- Applications that are generating data to be ingested by Kinesis data stream.
- For each record, producers associate partition keys.
- Partition keys are important to map records to shards.

Data Consumer:

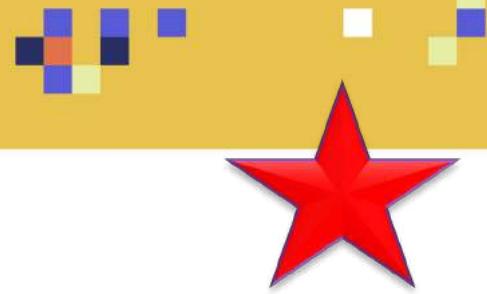
- Consists of Kinesis applications receiving data from shards in stream.
- Real-time performance allow for spot analytics.

Data Stream:

- A group of shards form a data stream.
- Data is retained for 24 hours by default.
- Data could be retained up to 7 days (upon request).
- Data is immutable; once ingested in KDS, cannot be deleted

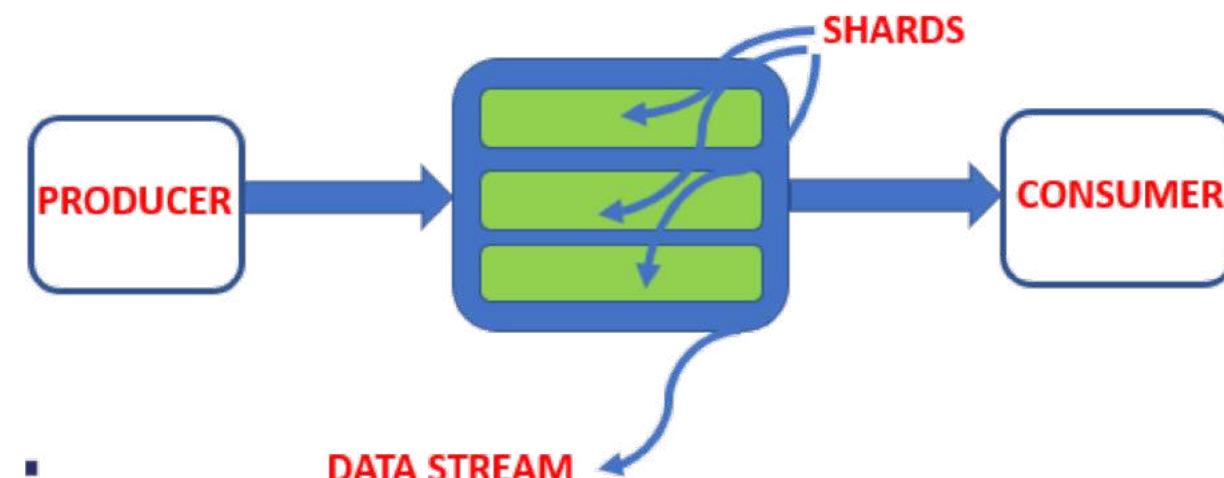
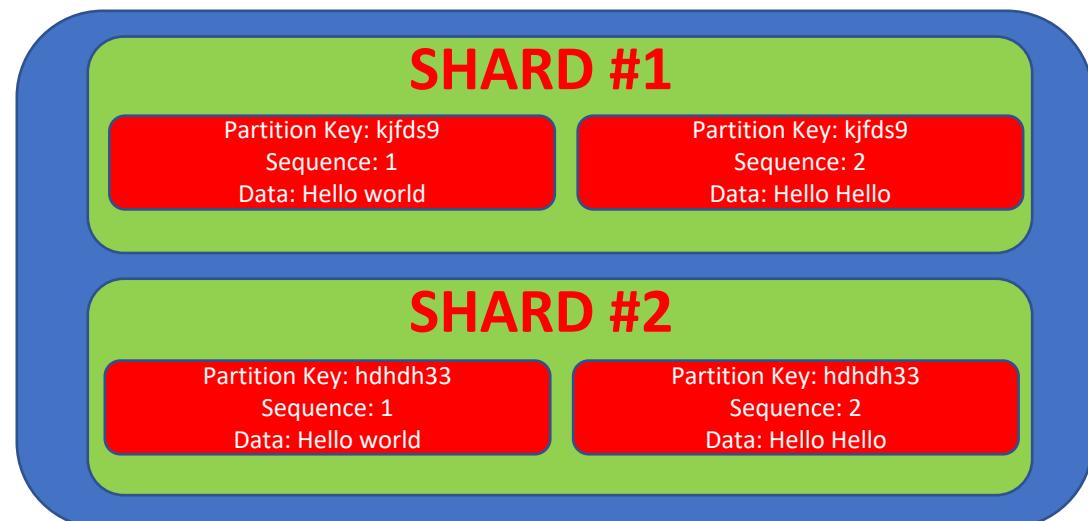


2. KINESIS DATA STREAMS: DEFINITIONS



Shard:

- A shard is the base throughput unit
- Shards are portioned in advance which require capacity planning.
- Each shard = 1000 data records per second (or 1MB/sec).
- Shards could be added via AWS console, auto scaling, UpdateShardCount API, trigger automatic scaling via AWS Lambda.
- Default limit is 500 shards but unlimited number could be requested.
- Data record consists of a unit of captured data including: (1) Sequence number, (2) partition key, and (3) payload (limited to 1 MB).



2. KINESIS DATA STREAMS: INTERACTION



KINESIS PRODUCER LIBRARY (KPL)

- The KCL allows for data writing to a Kinesis Data Streams.
- KPL simplifies producer application development

KINESIS CLIENT LIBRARY (KCL)

- The KCL allows for data consumption from Kinesis Data Streams.
- It acts as an intermediary between processing record code and Kinesis Data Streams.
- KCL handles the complex tasks of managing instances.

KINESIS SDK API

- *KPL can incur an additional processing delay compared to AWS SDK API.*
- *For time sensitive applications, it is recommended to use AWS SDK directly.*

2. KINESIS DATA STREAMS: SHARDS



Amazon Kinesis

Dashboard

Data Streams

Data Firehose

Data Analytics

Video Streams

External resources

What's new

Create Kinesis stream

Kinesis stream name* Trial_Stream

Acceptable characters are uppercase and lowercase letters, numbers, underscores, hyphens, and periods.

Shards

A shard is a unit of throughput capacity. Each shard ingests up to 1MB/sec and 1000 records/sec, and emits up to 2MB/sec. To accommodate for higher or lower throughput, the number of shards can be modified after the Kinesis stream is created using the API. [Learn more](#)

▼ Estimate the number of shards you'll need

Shard calculator

Average record size KB
Record size is an integer between 1 and 1024

Max records written per second
(Number of records per second) x (Number of producers)

Number of consumer applications

Estimated shards

Number of shards*

You can provision up to 500 more shards before hitting your account limit of 500.

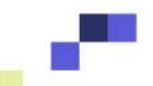
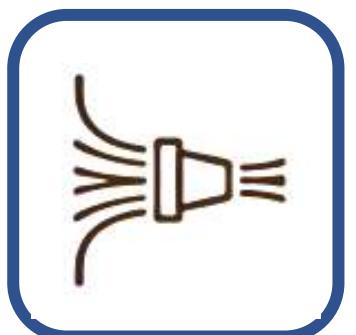
AWS KINESIS FIREHOSE



3. KINESIS FIREHOSE: OVERVIEW



- Amazon Kinesis Data Firehose allows for easy and cost effectively way to load streaming data into data lakes.
- Amazon kinesis firehose allows for **near real-time** analytics.
- Using Kinesis data firehose, data could be ingested, transformed, encrypted and loaded into:
 - Amazon S3
 - Amazon Redshift
 - Amazon Elasticsearch Service
 - Splunk
- Kinesis data firehose does not require upfront cost or infrastructure management.
- It follows pay per use model.
- It automatically scales to match any throughput of data.
- Data Transformation through AWS Lambda (CSV => JSON) and supports compression with Amazon S3 as destination
- To optimize cost, Amazon Kinesis firehose allows for converting the incoming data into columnar formats such as Apache Parquet and Apache ORC, before feeding it into Amazon S3.
- Great Video: <https://aws.amazon.com/kinesis/data-firehose/>



3. KINESIS FIREHOSE: UNIQUE FEATURES



EXTREMELY EASY TO SETUP AND USE

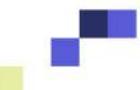
- In minutes, you can capture, transform, and load streaming data from multiple sources.
Integrated with AWS data lakes and data stores
- All scaling, sharding, servers maintenance end management are taken care by AWS.

SEAMLESS INTEGRATION WITH AWS STORAGE

- Amazon Kinesis Data Firehose is easily integrated with Amazon S3, Amazon Redshift, and Amazon Elasticsearch Service.

SERVERLESS DATA TRANSFORMATION

- Without the need to create data processing pipeline (no servers), you can use Kinesis Data Firehose to convert raw data into any format before storing it into S3.



3. KINESIS FIREHOSE: UNIQUE FEATURES



NEAR REAL-TIME PERFORMANCE

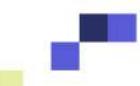
- Amazon Kinesis Data Firehose captures and loads data into S3, Redshift and ElasticSearch within 60 secs data (near real-time).

NO MAINTENANCE AND SERVERS ADMINISTRATION

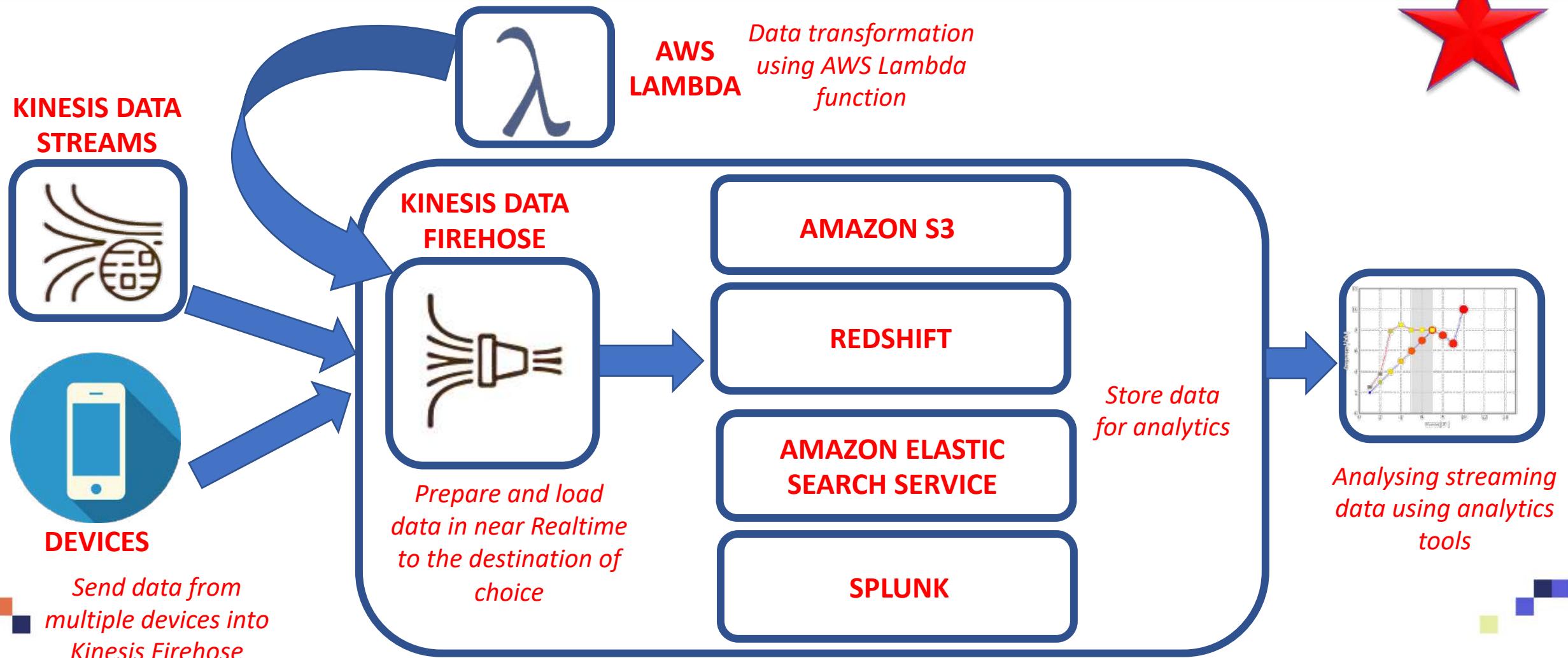
- Fully managed service with automatic scaling.

PAY PER USE

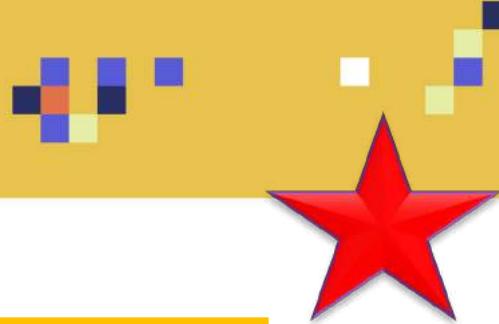
- No upfront cost and pay per use model.



3. KINESIS FIREHOSE: HOW DOES IT WORK?



3. KINESIS FIREHOSE VS. STREAMS



	KINESIS DATA STREAMS	KINESIS FIREHOSE
USE/SERVICE	SCALABLE REAL-TIME STREAMING SERVICE	DATA TRANSFER SERVICE TO LOAD STREAMING DATA INTO S3, REDSHIFT AND ELASTICSEARCH
LATENCY	REAL TIME (~200 MS FOR CLASSIC AND 70 MS FOR ENHANCED FAN OUT)	NEAR REAL TIME (60 SECS)
STORAGE	FROM 1 TO 7 DAYS	NO DATA STORAGE
SCALING	REQUIRE SHARDS ADMINISTRATION	AUTO SCALING
SERVICE MANAGEMENT	MANAGED SERVICE EXCEPT FOR SHARDS CONFIGURATION	FULLY MANAGED

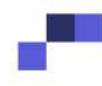
AWS KINESIS DATA ANALYTICS – PART #1



4. KINESIS DATA ANALYTICS: OVERVIEW



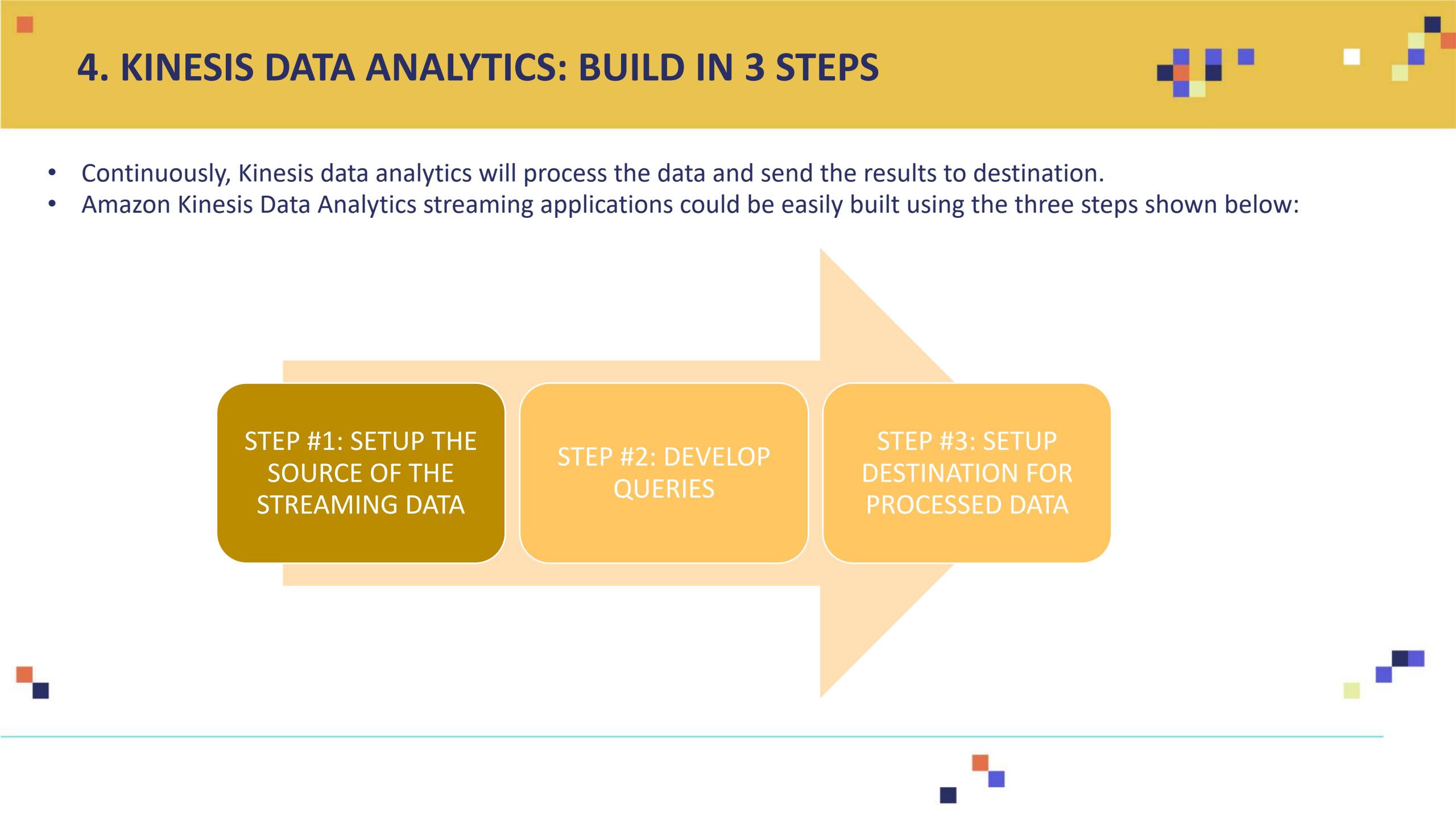
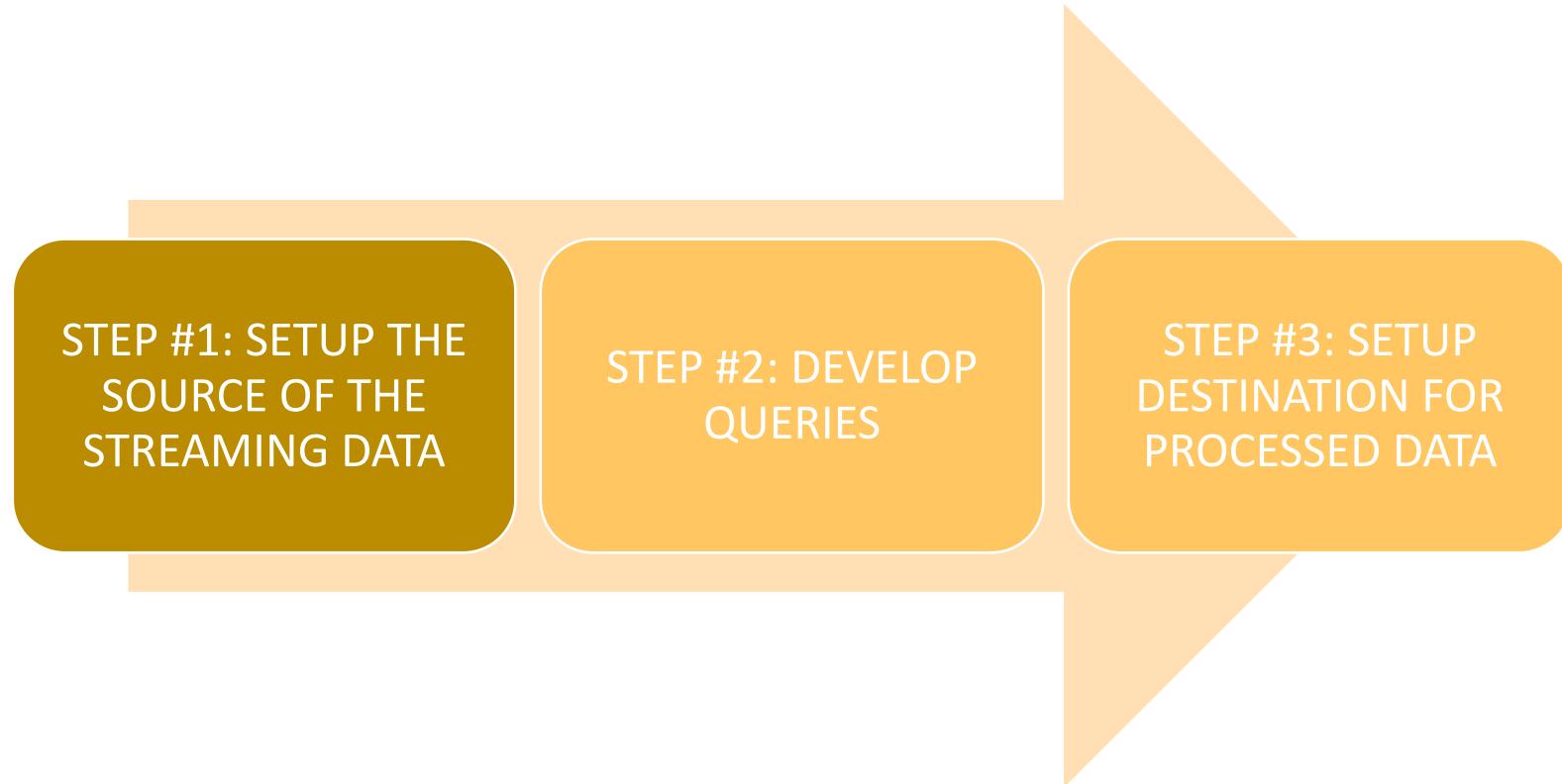
- Amazon Kinesis Data Analytics is a powerful tool to perform analysis on streaming data and get important insights.
- Amazon Kinesis data analytics run in Realtime and is fully automatically scalable based on incoming data throughput.
- Zero setup and upfront cost. Pay per use model.
- Available for SQL and JAVA developers:
 - **For SQL users/developers:**
 - Developers can query streaming data by leveraging readily available templates.
 - Developers can select a template for specific analytics task, edit code using SQL editor.
 - **For JAVA developers:**
 - Developers can leverage open source Java libraries to perform data analysis in Real-time
 - Java Library includes over 25 pre-built operators to perform streaming data aggregation, filtering and transformation.



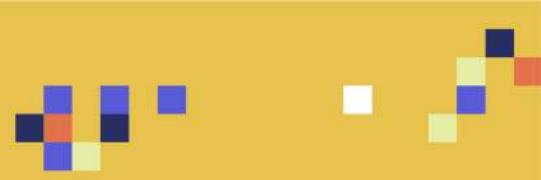
4. KINESIS DATA ANALYTICS: BUILD IN 3 STEPS



- Continuously, Kinesis data analytics will process the data and send the results to destination.
- Amazon Kinesis Data Analytics streaming applications could be easily built using the three steps shown below:



4. KINESIS DATA ANALYTICS: BENEFITS



POWERFUL PERFORMANCE

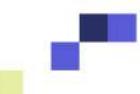
- Realtime performance with minimum latency.

SERVERLESS

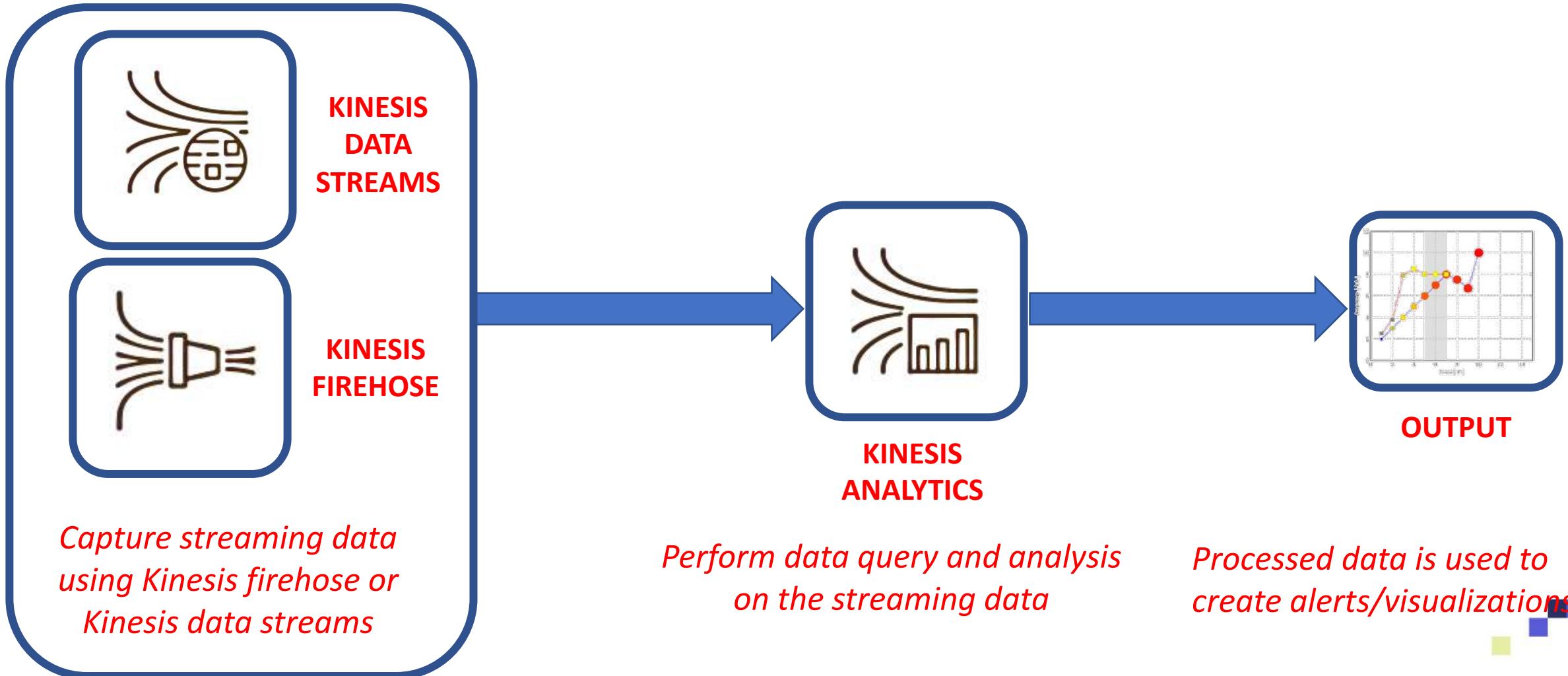
- Zero upfront cost or servers/infrastructure to manage.
- Automatic scaling to ensure best performance.

PAY PER USE

- No commitment and pay-per-use model offers a cost effective solution.



4. KINESIS DATA ANALYTICS: HOW IT WORKS?

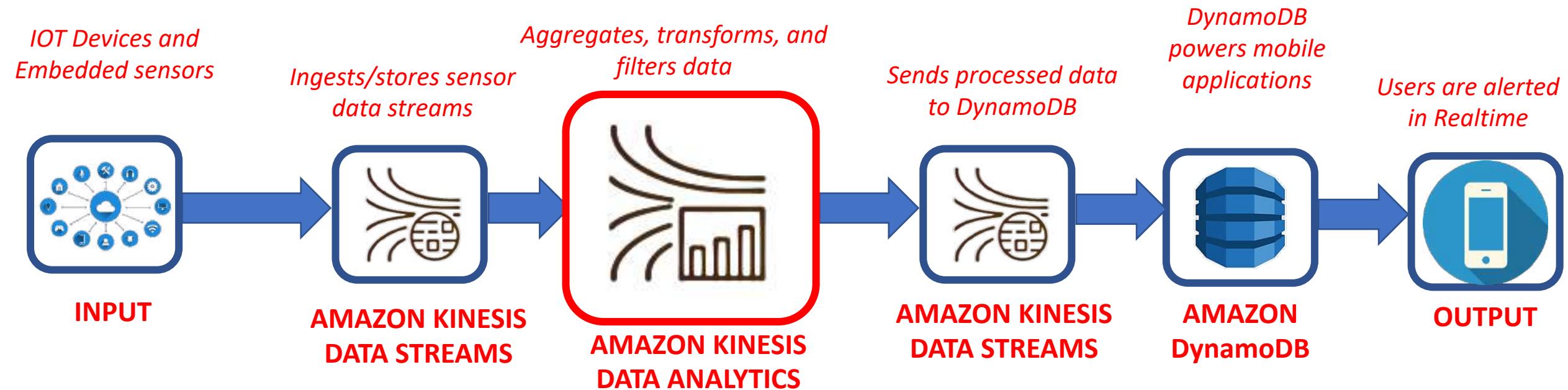


AWS KINESIS DATA ANALYTICS – PART #2



4. KINESIS DATA ANALYTICS: USE CASE #1: STREAMING ETL FOR IOT

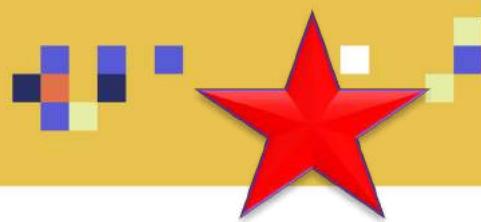
- Amazon Kinesis Data Analytics could be used to transform and filter streaming data from IOT devices such as sensors and then send real-time alerts when a variable exceeds a certain limit.



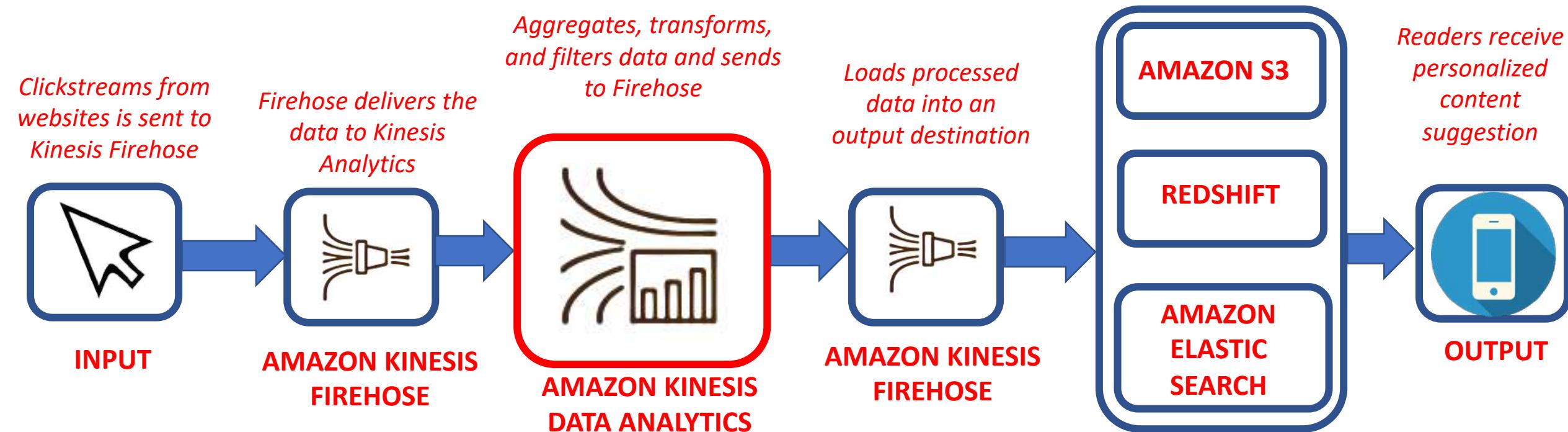
<https://pixabay.com/illustrations/iot-internet-of-things-network-3337536/>

Photo Credit: <https://commons.wikimedia.org/wiki/File:Mobile-Smartphone-icon.png>

4. KINESIS DATA ANALYTICS: USE CASE #2: REALTIME LOG ANALYTICS WITH SQL



- Amazon Kinesis Data Analytics could be used to calculate metrics sent from users clickstreams (what did users click? How long did they stay on website?) and then present users with personalized content suggestions and targeted ads.

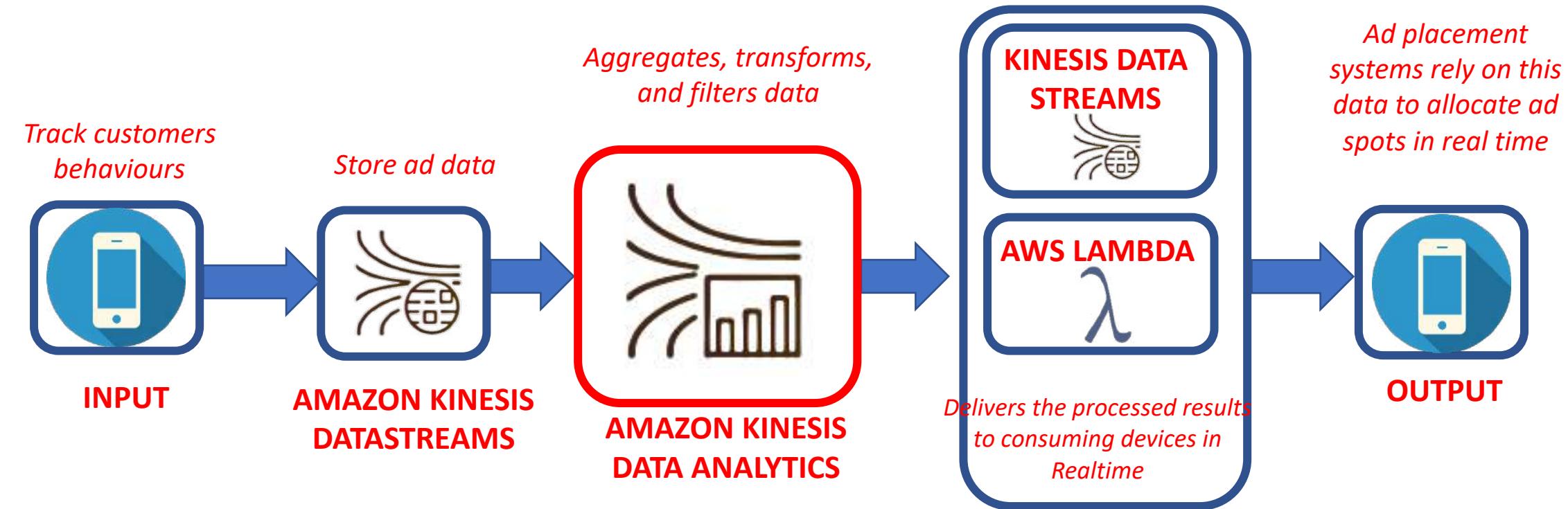


<https://pixabay.com/illustrations/iot-internet-of-things-network-3337536/>

Photo Credit: <https://commons.wikimedia.org/wiki/File:Mobile-Smartphone-icon.png>

4. KINESIS DATA ANALYTICS: USE CASE #3: DIGITAL MARKETING WITH SQL

- Amazon Kinesis Data Analytics could be used to perform data transformation in Realtime to offer an optimized digital marketing solutions.

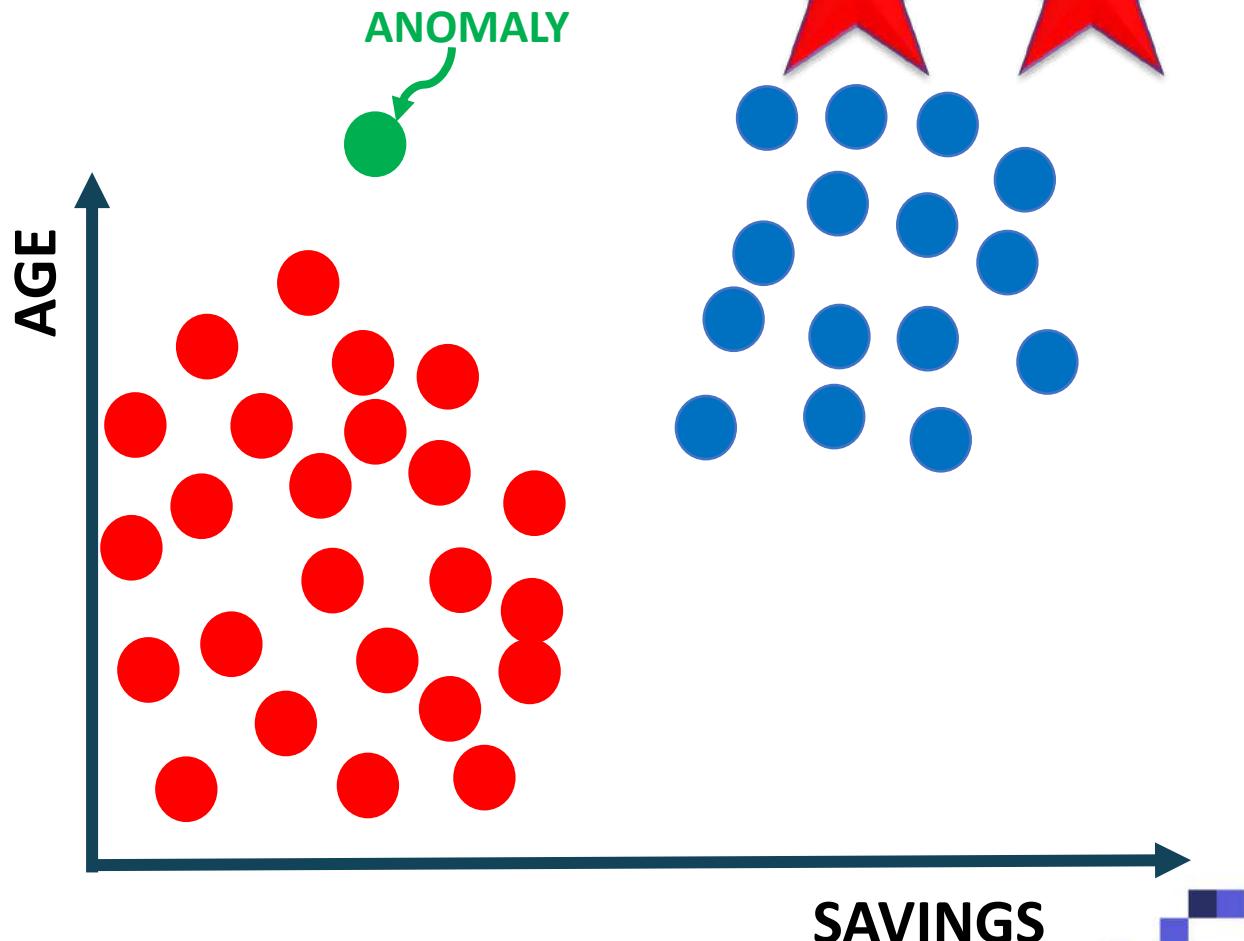


<https://pixabay.com/illustrations/iot-internet-of-things-network-3337536/>

Photo Credit: <https://commons.wikimedia.org/wiki/File:Mobile-Smartphone-icon.png>

4. KINESIS DATA ANALYTICS: MACHINE LEARNING RANDOM CUT FOREST

- Amazon Kinesis Data Analytics includes a function named RANDOM_CUT_FOREST that is used for anomaly detection.
- The function works by assigning an anomaly score to each data record.
- Anomalies are data points that diverge from the rest of the properly structured data.
- Anomalies could be spikes or breaks in the dataset and could be detected from the regular structured dataset.
- Anomaly detection and removal is crucial in machine learning because adding anomalies will unnecessarily increase the complexity of the model.



4. KINESIS DATA ANALYTICS: MACHINE LEARNING HOTSPOTS

- Amazon Kinesis Data Analytics includes a function named HOTSPOTS
- HOTSPOTS can be utilized to locate to identify dense areas in the data (activity in some regions that might be higher than the norm).
- By identifying “hot” regions, you can then focus the attention on these areas and take actions accordingly.

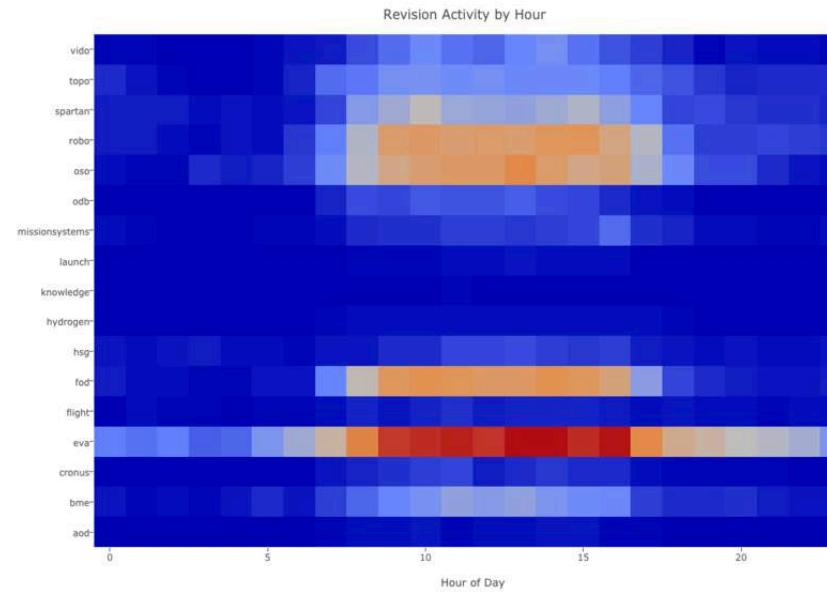


Photo Credit: https://commons.wikimedia.org/wiki/File:Heatmap_of_revision_activity_by_hour.png

KINESIS SUMMARY

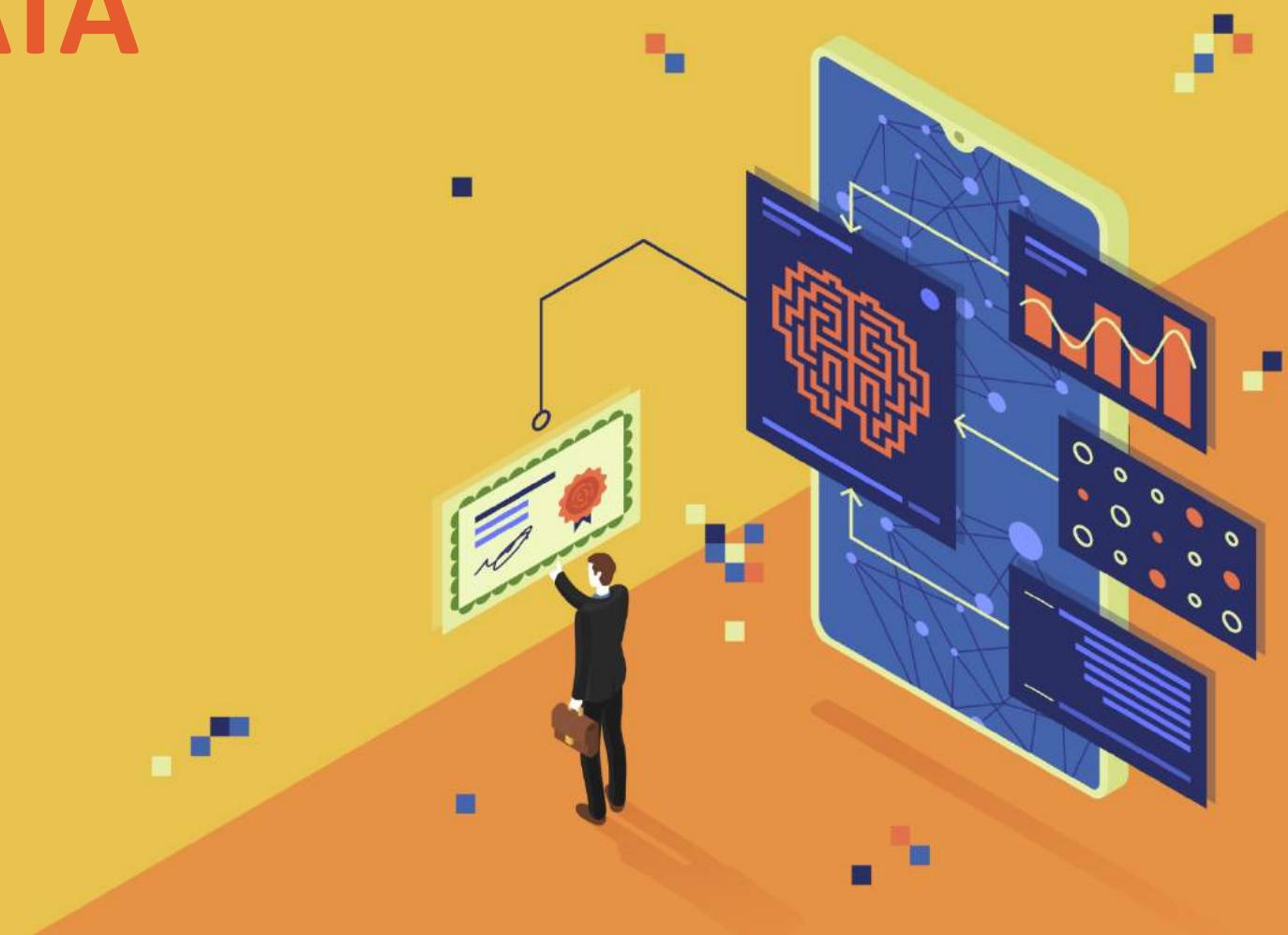


KINESIS SERVICE	USE CASE
Kinesis Data Streams	For real-time streaming of data and to gain real-time insights using ML applications
Kinesis Firehose	For streaming data in near real time and storing them into S3 or redshift
Kinesis Video Streams	For streaming live videos in real time
Kinesis Analytics	To run SQL queries on streaming data and generate graphs and gain valuable metrics.

AWS MACHINE LEARNING CERTIFICATION



DOMAIN #2: EXPLORATORY DATA ANALYSIS (24% EXAM)



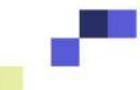
INTRODUCTION



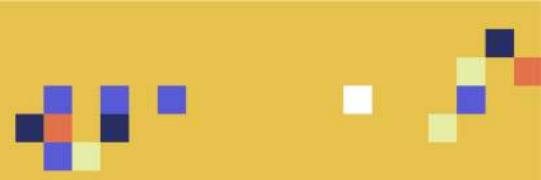
INTRODUCTION



- This part of the exam accounts for 24% and is not specific to AWS or SageMaker.
- This section of the test will assess your skills in feature engineering and data cleaning.
- This section will cover fundamentals such as normalization, one hot encoding, and filling in missing data values.
- This section relies on Python and famous packages such as pandas, numpy, and seaborn.
- Please note that the exam will not test you on python or ask you to write any code.



AWS ML CERTIFICATION EXAM DOMAINS



Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #2 OVERVIEW:

SECTION #5: JUPYTER NOTEBOOKS, SCIKIT LEARN, PYTHON PACKAGES, AND DISTRIBUTIONS

- Introduction
- Jupyter Notebooks and Scikit Learn
- Python Packages (Pandas, Numpy, Matplotlib and Seaborn)
- Distributions (Normal, Standard, Poisson, Bernoulli)
- Time Series

SECTION #6: AMAZON ATHENA, QUICKSIGHT AND ELASTIC MAP REDUCE

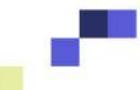
- Amazon Athena Features
- Amazon Athena Deep Dive (Security, Cost, and glue integration)
- Amazon QuickSight Features
- Amazon QuickSight (integration with AWS services)
- Amazon QuickSight ML insights and Use Cases
- Elastic Map Reduce (EMR)
- Apache Hadoop with EMR
- Apache Spark with EMR

DOMAIN #2 OVERVIEW:



SECTION #7: FEATURE ENGINEERING

- Introduction to Feature Engineering
- Amazon SageMaker GroundTruth
- Feature Selection
- Scaling
- Imputation
- Outliers
- One Hot Encoding
- Binning
- Log Transformation
- Shuffling, Feature Splitting, Unbalanced Datasets
- Text Feature Engineering overview
- Bag of words, punctuation, and dates (easy ones!)
- Term Frequency Inverse Document Frequency (TF-IDF)
- N-Grams (Unigram vs. Bigram vs. Trigram)
- Orthogonal Sparse Bigram (OSB)
- Cartesian Product Transformation



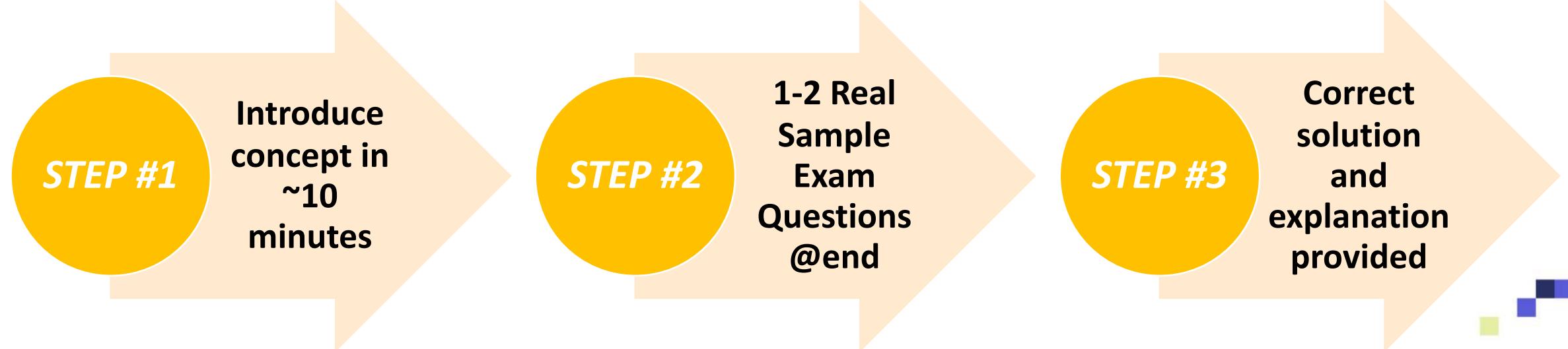
LECTURE DESIGN



- We know how hard it is to study for an exam especially if you have a busy schedule.
- This course is designed to be extremely on point and optimized to pass the exam.

No boring content. Zero unnecessary information.

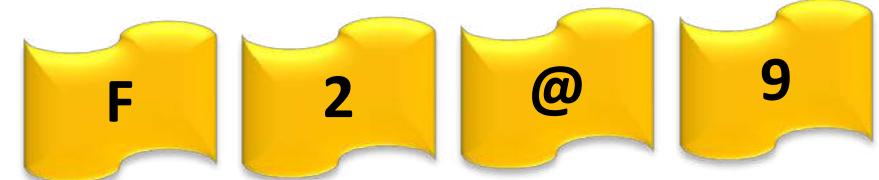
- Here's the lecture structure that we will follow:



RECALL OUR MINI CHALLENGE AND PRIZE!



- For those of you who will successfully complete the entire section and watch the videos till the end, they will receive a valuable prize!



INTRODUCTION



- In order to perform data visualization, there are generally two approaches: (1) Use Developer tools or (2) use Business intelligence tools.

DEVELOPER TOOLS

pandas
 $y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$

Jupyter

INSTALL PROJECT DOCUMENTATION BLOG DONATE

The Jupyter Notebook is a web-based interactive computing platform that allows users to author data- and code-driven narratives that combine live code, equations, narrative text, visualizations, interactive dashboards and other media.

AMAZON SAGEMAKER

BUSINESS INTELLIGENCE TOOLS

AMAZON QUICKSIGHT

TABLEAU

Microsoft Power BI

POWER BI

JUPYTER NOTEBOOKS AND SCIKIT LEARN



JUPYTER NOTEBOOKS



- Jupyter Notebooks are open-source web application that enable developers to develop and distribute codes, text, equations, and figures in one place.
- It's one of the top tools used for machine learning developers.
- In Jupyter notebooks, you can write in 40 programming languages such as Python, R, and Scala.
- You can Share notebooks including code results with other.
- Jupyter enable developers to leverage big data tools, such as Apache Spark, from Python, R and Scala.
- <https://jupyter.org/>

The screenshot shows a Jupyter Notebook titled "Python Programming Fundamentals - Part B". The notebook has a "Trusted" status and is using Python 3. The main content area displays the following text:

This Notebook will cover the following topics:

- Numpy basics
- Built-in methods and functions
- Obtain shape, length and type of Numpy arrays
- Reshape
- Minimum and maximum and their indices
- Mathematical Operations
- Indexing and slicing
- Selection

NUMPY BASICS

- NumPy is a Linear Algebra Library used for multidimensional arrays
- Installation: Use the command window, type: `conda install numpy`

In [2]:

```
1 import numpy as np
```

In [3]:

```
1 # One-dimensional array
2 my_list = [5, 3, 10]
3 my_list
```

Out[3]:

```
[5, 3, 10]
```

In [4]:

```
1 y = np.array(my_list)
```

In [5]:

```
1 y
```

Out[5]:

```
array([5, 3, 10])
```

In [6]:

```
1 type(y)
```

Out[6]:

```
numpy.ndarray
```

In [7]:

```
1 # multi-dimensional (Matrix definition)
2 matrix = np.array([[1, 2], [3, 4]])
```

SCIKIT LEARN



- Scikit-learn is a free machine learning library developed for python.
- Scikit-learn offers several algorithms for classification, regression, clustering
- Several famous models are included such as support vector machines, random forests, gradient boosting, and k-means.
- Scikit learn can be efficiently used in data preprocessing.

```
In [24]: 1 from sklearn.model_selection import train_test_split  
2 |  
3 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20, random_state=5)  
  
In [25]: 1 X_train.shape  
  
Out[25]: (455, 30)  
  
In [26]: 1 X_test.shape  
  
Out[26]: (114, 30)  
  
In [27]: 1 y_train.shape  
  
Out[27]: (455,)  
  
In [28]: 1 y_test.shape  
  
Out[28]: (114,)  
  
In [29]: 1 from sklearn.svm import SVC  
2 from sklearn.metrics import classification_report, confusion_matrix  
3 |  
4 svc_model = SVC()  
5 svc_model.fit(X_train, y_train)
```

SPLITTING THE DATA INTO TRAINING AND TESTING

TRAINING A SUPPORT VECTOR MACHINES

SAGEMAKER JUPYTER NOTEBOOKS



CREATE A NOTEBOOK INSTANCE IN SAGEMAKER



AWS Services Resource Groups

Amazon SageMaker

Dashboard Search

Ground Truth Labeling jobs Labeling datasets Labeling workforces

Notebook **Notebook instances**

Training Algorithms Training jobs Hyperparameter tuning jobs

Inference Compilation jobs Model packages Models

Endpoint configurations Endpoints Batch transform jobs

AWS Marketplace

MACHINE LEARNING

Amazon SageMaker

Build, train, and deploy machine learning models at scale

The quickest and easiest way to get ML models from idea to production.

How it works

Label	Build	Train
		
Set up and manage labeling jobs for highly accurate training datasets within Amazon SageMaker, using active learning and human	Connect to other AWS services and transform data in Amazon SageMaker notebooks	Use Amazon SageMaker's algorithms and frameworks, or bring your own, for distributed training

Get started

Explore AWS data in your notebooks, and use algorithms to create models via training jobs. Leverage Notebook instances in the cloud to begin.

Create notebook instance

Start with an overview

Pricing (US)

With Amazon SageMaker, you pay only for what you use. Authoring, training and hosting is billed by the second, with no minimum fees and no upfront commitments.

Learn more

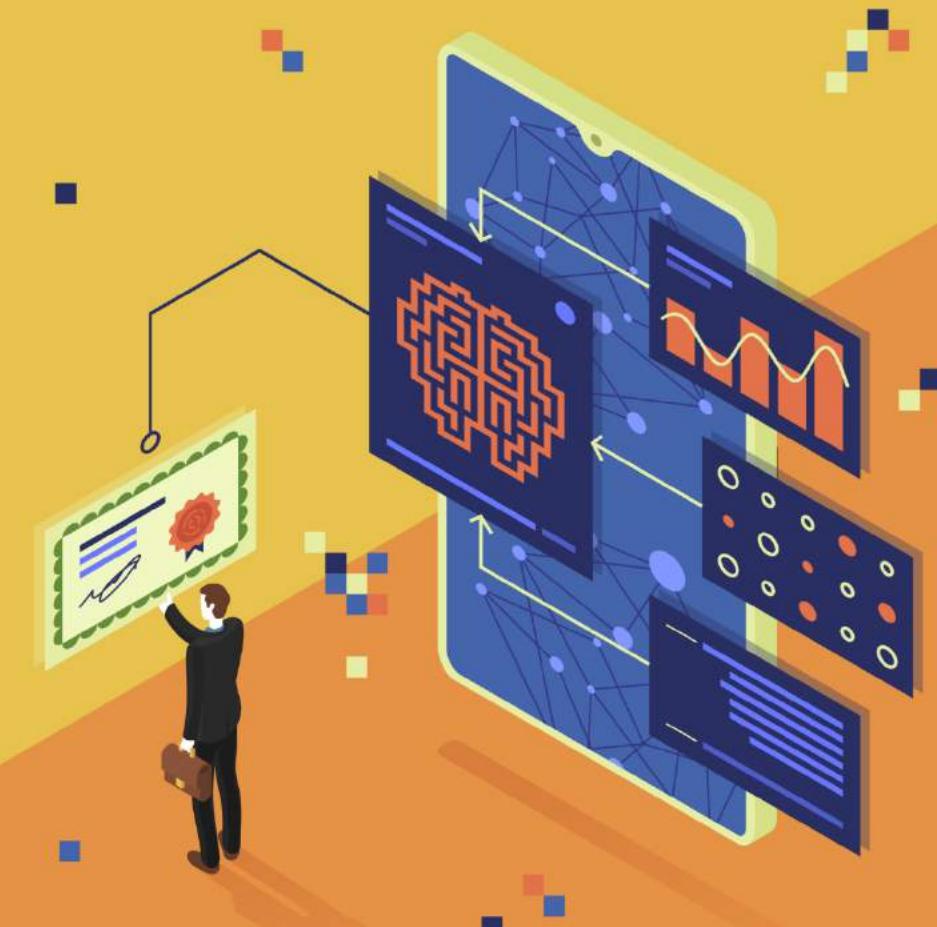
Related services

AWS Glue

Amazon EC2

Amazon Elastic Block Store (EBS)

PYTHON PACKAGES



NUMPY: MATHEMATICAL OPERATIONS



MATHEMATICAL OPERATIONS

```
In [30]: 1 x = np.arange(1, 5)
          2 x
```

```
Out[30]: array([1, 2, 3, 4])
```

```
In [50]: 1 y = np.arange(1, 5)
          2 y
```

```
Out[50]: array([1, 2, 3, 4])
```

```
In [32]: 1 z = x+y
          2 z
```

```
Out[32]: array([2, 4, 6, 8])
```

```
In [33]: 1 z = x**2
          2 z
```

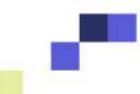
```
Out[33]: array([ 1,  4,  9, 16], dtype=int32)
```

```
In [34]: 1 k = np.sqrt(z)
          2 k
```

```
Out[34]: array([1., 2., 3., 4.])
```

```
In [35]: 1 z = np.exp(y)
          2 z
```

```
Out[35]: array([ 2.71828183,  7.3890561 , 20.08553692, 54.59815003])
```



NUMPY: MATRIX DEFINITIONS AND ELEMENTS SELECTION



ELEMENTS SELECTION

```
In [47]: 1 matrix = np.random.randint(1,10, (5, 5))  
2 matrix
```

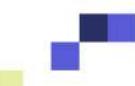
```
Out[47]: array([[5, 9, 8, 8, 8],  
                 [3, 9, 2, 8, 4],  
                 [1, 9, 5, 8, 4],  
                 [3, 3, 3, 8, 9],  
                 [4, 4, 7, 8, 5]])
```

```
In [48]: 1 new_matrix = matrix[matrix>3]  
2 new_matrix
```

```
Out[48]: array([5, 9, 8, 8, 8, 9, 8, 4, 9, 5, 8, 4, 8, 9, 4, 4, 4, 7, 8, 5])
```

```
In [49]: 1 new_matrix = matrix[matrix%2==0]  
2 new_matrix
```

```
Out[49]: array([8, 8, 8, 2, 8, 4, 8, 4, 8, 4, 4, 4, 8])
```





- Pandas is an open source library that offers high-performance data structures and data analysis tools in python.
- Data can also be stored using pandas DataFrame.
- Think of it as using Microsoft excel in python/jupyter environment.

GETTING CSV DATA

```
In [14]: 1 import pandas as pd  
2  
3 df = pd.read_csv('sample_file.csv')
```

```
In [15]: 1 df
```

Out[15]:

	first	last	email	postal	gender	dollar
0	Joseph	Patton	daafeja@boh.jm	M6U 5U7	Male	\$2,629.13
1	Noah	Moran	guutodi@bigwoc.kw	K2D 4M9	Male	\$8,626.96
2	Nina	Keller	azikez@gahew.mr	S1T 4E6	Male	\$9,072.02

```
In [16]: 1 # write to a csv file  
2 df.to_csv('sample_output.csv',index=False)
```

MATPLOTLIB



BASIC PLOT

```
In [3]: 1 import numpy as np  
2 x = np.arange(0, 10, 0.2) # evenly sampled time at 0.2 s intervals  
3 x|
```

```
Out[3]: array([0. , 0.2, 0.4, 0.6, 0.8, 1. , 1.2, 1.4, 1.6, 1.8, 2. , 2.2, 2.4,  
2.6, 2.8, 3. , 3.2, 3.4, 3.6, 3.8, 4. , 4.2, 4.4, 4.6, 4.8, 5. ,  
5.2, 5.4, 5.6, 5.8, 6. , 6.2, 6.4, 6.6, 6.8, 7. , 7.2, 7.4, 7.6,  
7.8, 8. , 8.2, 8.4, 8.6, 8.8, 9. , 9.2, 9.4, 9.6, 9.8])
```

```
In [4]: 1 y = np.sin(x)
```

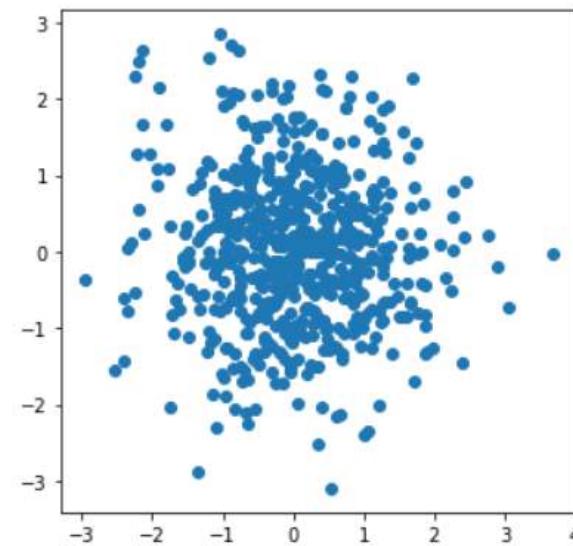
```
In [6]: 1 plt.plot(x, y)  
2 plt.xlabel('Time')  
3 plt.ylabel('Sine Wave')  
4 plt.title('My first plotting exercise!')
```

```
Out[6]: Text(0.5,1,'My first plotting exercise!')
```



```
In [6]: 1 import random  
2  
3 fig = plt.figure(figsize=(5,5))  
4  
5 X = np.random.randn(600)  
6 Y = np.random.randn(600)  
7  
8 plt.scatter(X,Y)  
9
```

```
Out[6]: <matplotlib.collections.PathCollection at 0x1a4214bb400>
```

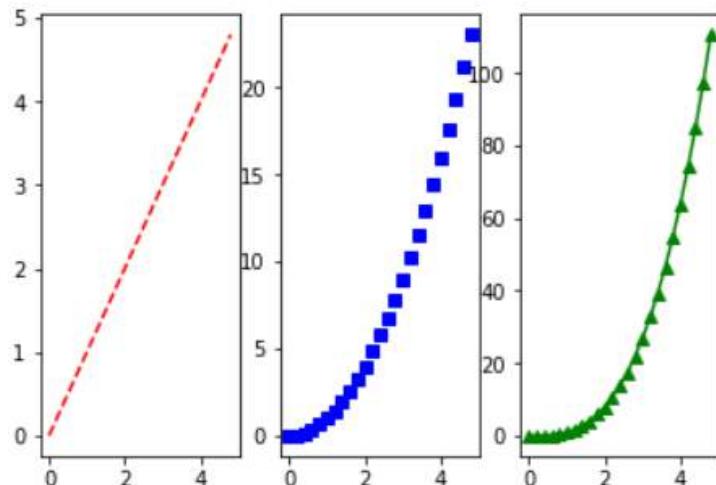




SUBPLOTS

In [8]:

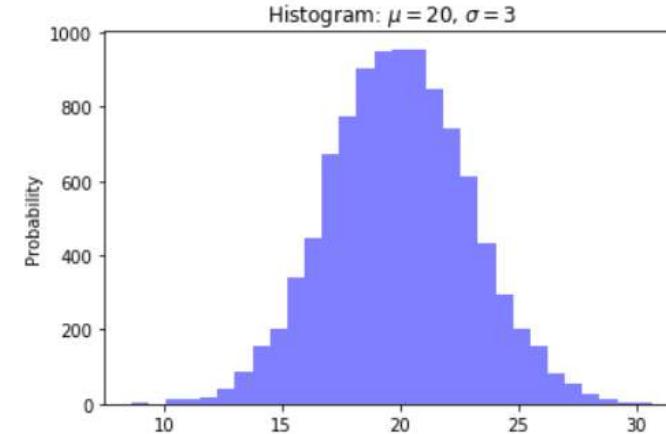
```
1 plt.subplot(1, 3, 1)
2 plt.plot(t, t, 'r--');
3
4 plt.subplot(1, 3, 2)
5 plt.plot(t, t**2, 'bs')
6
7 plt.subplot(1, 3, 3)
8 plt.plot(t, t**3, 'g^-');
```



In [18]:

```
1 mu = 20 # mean of distribution
2 sigma = 3 # standard deviation of distribution
3 x = mu + sigma * np.random.randn(10000)
4
5 num_bins = 30
6
7 n, bins, patches = plt.hist(x, num_bins, facecolor='blue', alpha=0.5)
8
9 plt.ylabel('Probability')
10 plt.title(r'Histogram: $\mu=20$, $\sigma=3$')
11
```

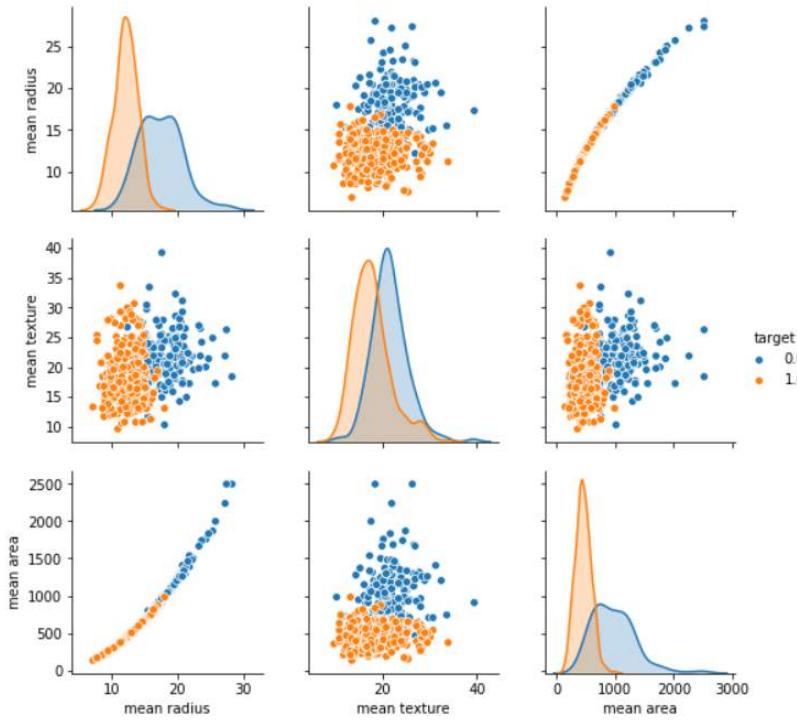
Out[18]: Text(0.5,1,'Histogram: \$\mu=20\$, \$\sigma=3\$')



SEABORN

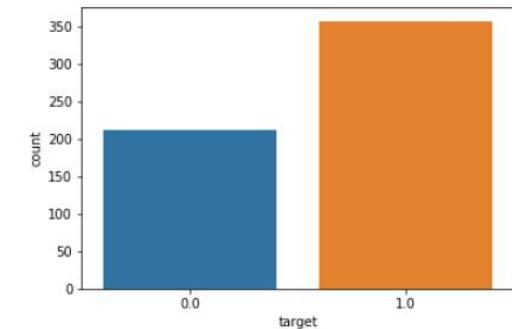
```
In [4]: 1 sns.pairplot(df_cancer, hue = 'target', vars = ['mean radius', 'mean texture', 'mean area'] )
```

```
Out[4]: <seaborn.axisgrid.PairGrid at 0x2cc12018cf8>
```



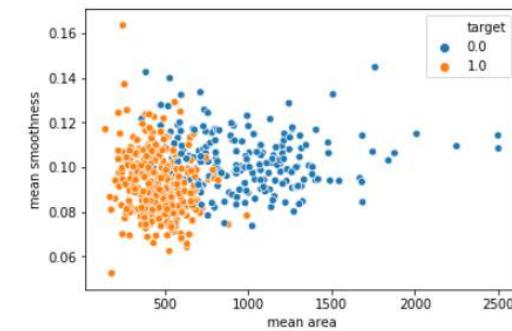
```
In [5]: 1 sns.countplot(df_cancer['target'], label = "Count")
```

```
Out[5]: <matplotlib.axes._subplots.AxesSubplot at 0x2cc13756b38>
```



```
In [6]: 1 sns.scatterplot(x = 'mean area', y = 'mean smoothness', hue = 'target', data = df_cancer)
```

```
Out[6]: <matplotlib.axes._subplots.AxesSubplot at 0x2cc13b139b0>
```

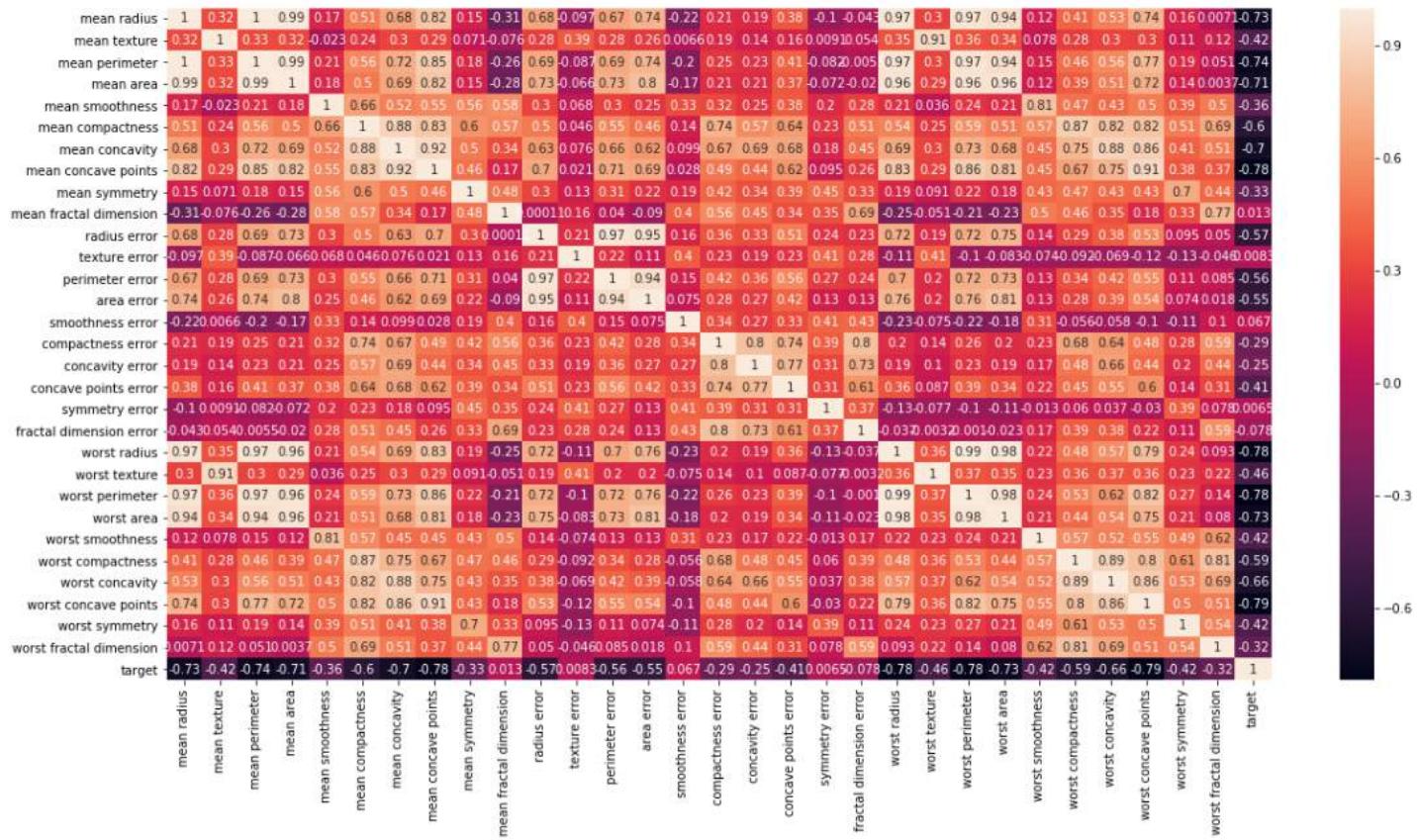


SEABORN: HEATMAPS

- Heat maps are used to represents values as colours.

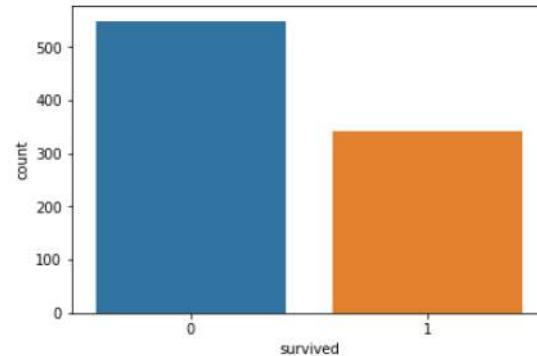
```
In [7]: 1 # Strong correlation between the mean radius and mean perimeter, mean area and mean perimeter
2 plt.figure(figsize=(20,10))
3 sns.heatmap(df_cancer.corr(), annot=True)
```

```
Out[7]: <matplotlib.axes._subplots.AxesSubplot at 0x2cc13b872b0>
```

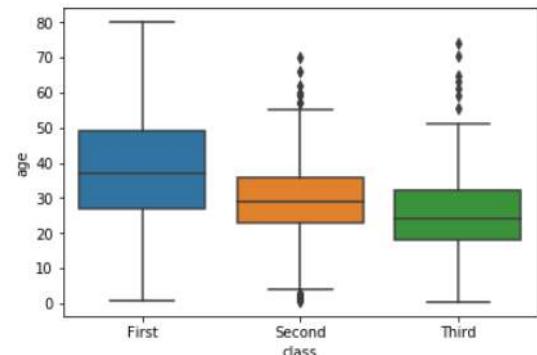


SEABORN

```
In [10]: 1 sns.countplot(x = 'survived', data = titanic_data)  
Out[10]: <matplotlib.axes._subplots.AxesSubplot at 0x2cc146dd6d8>
```

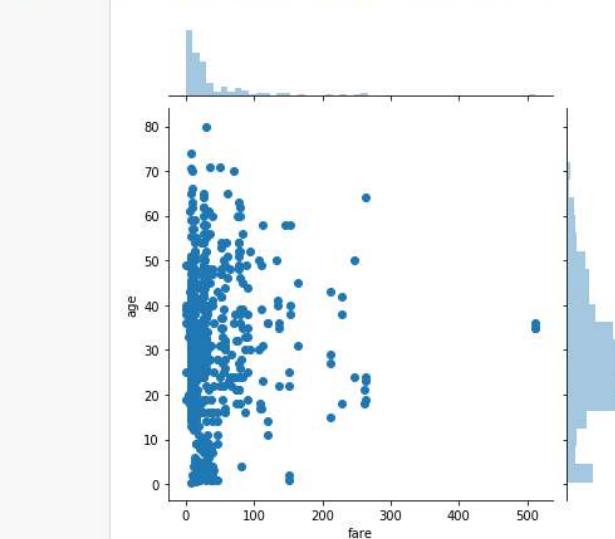


```
In [11]: 1 sns.boxplot(x='class', y='age', data=titanic_data)  
Out[11]: <matplotlib.axes._subplots.AxesSubplot at 0x2cc13cffc50>
```



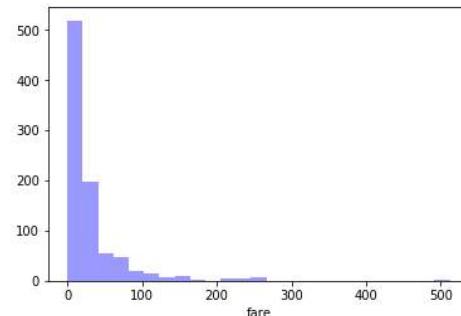
```
In [13]: 1 sns.jointplot(x='fare', y='age', data = titanic_data)
```

```
Out[13]: <seaborn.axisgrid.JointGrid at 0x2cc13ddaf98>
```

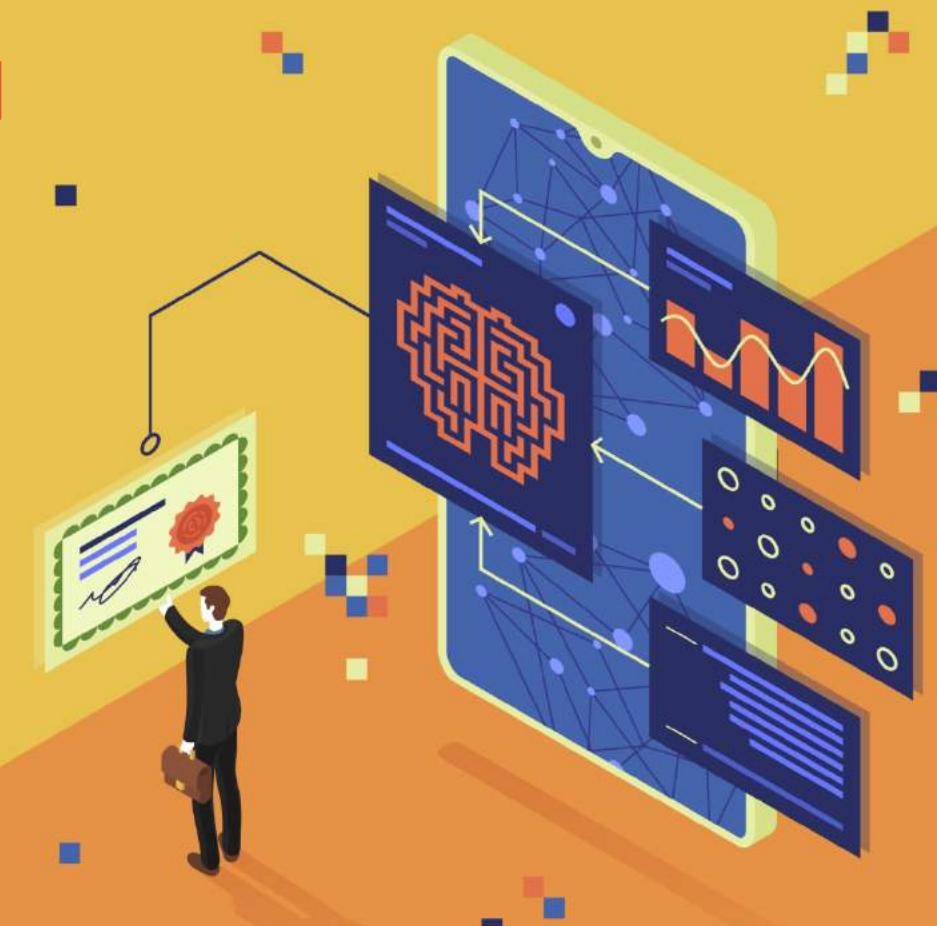


```
In [14]: 1 sns.distplot(titanic_data['fare'], bins = 25, kde= False,color = 'blue')
```

```
Out[14]: <matplotlib.axes._subplots.AxesSubplot at 0x2cc13e950b8>
```



DATA VISUALIZATION

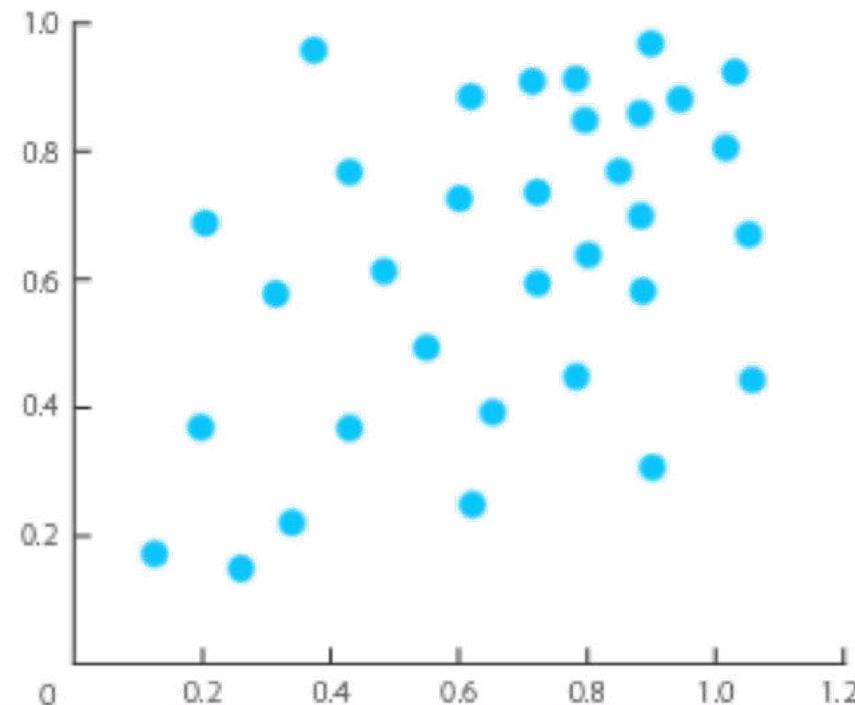


RELATIONSHIPS



SCATTERPLOT

“Scatterplot demonstrates the relationship between two variables (X, Y)”



BUBBLE CHART

“Bubble chart demonstrates the relationship between three variables (X, Y, Bubble Size)”

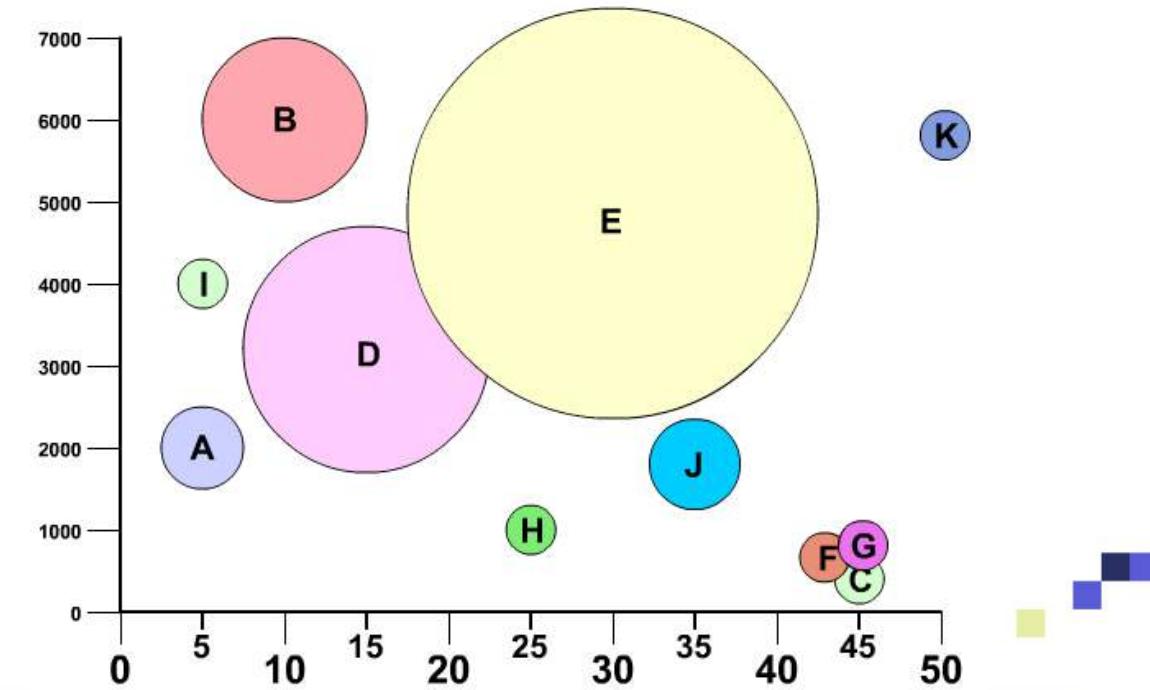


Photo Credit: https://commons.wikimedia.org/wiki/File:Example_of_Scatter_Plot.jpg

Photo Credit: https://commons.wikimedia.org/wiki/File:Bubble_chart.jpg

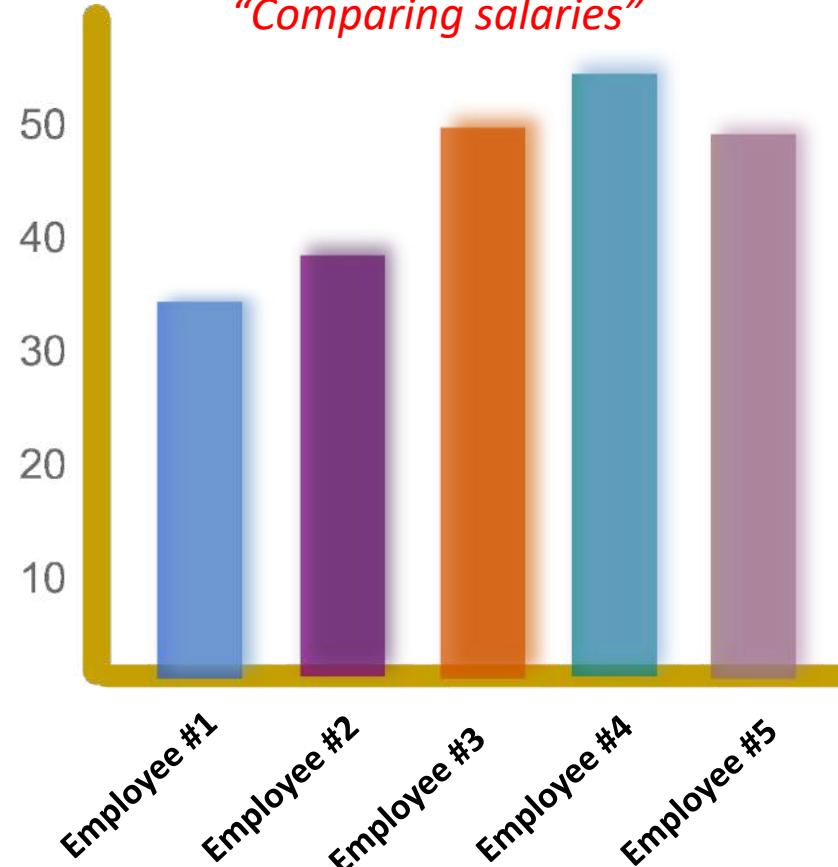


COMPARISONS



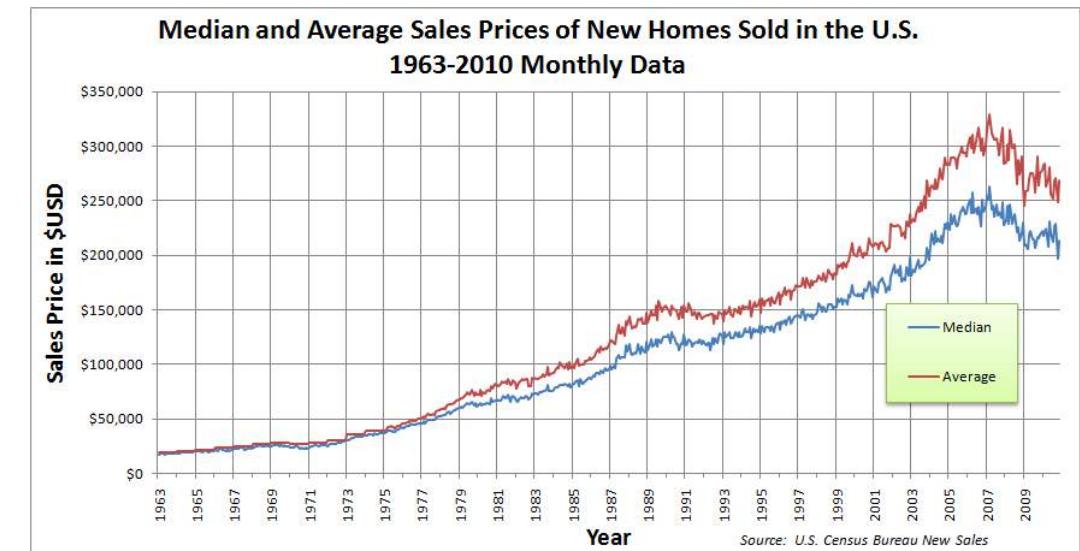
BAR CHART

"Comparing salaries"



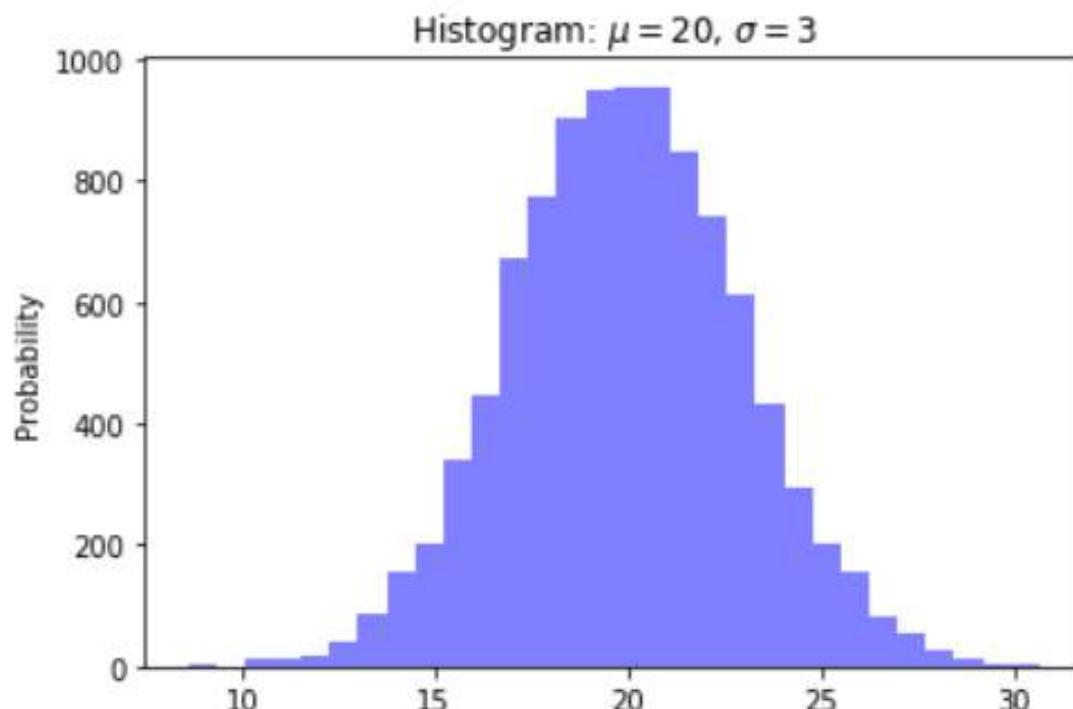
LINE CHART

"Comparing median and average House prices over the years"

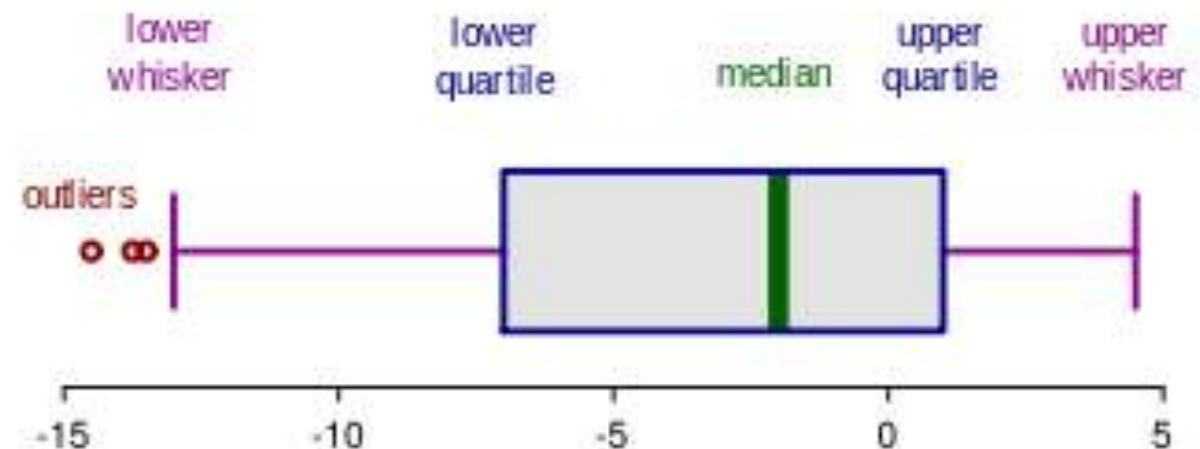




HISTOGRAMS



BOX PLOT



BOX PLOT

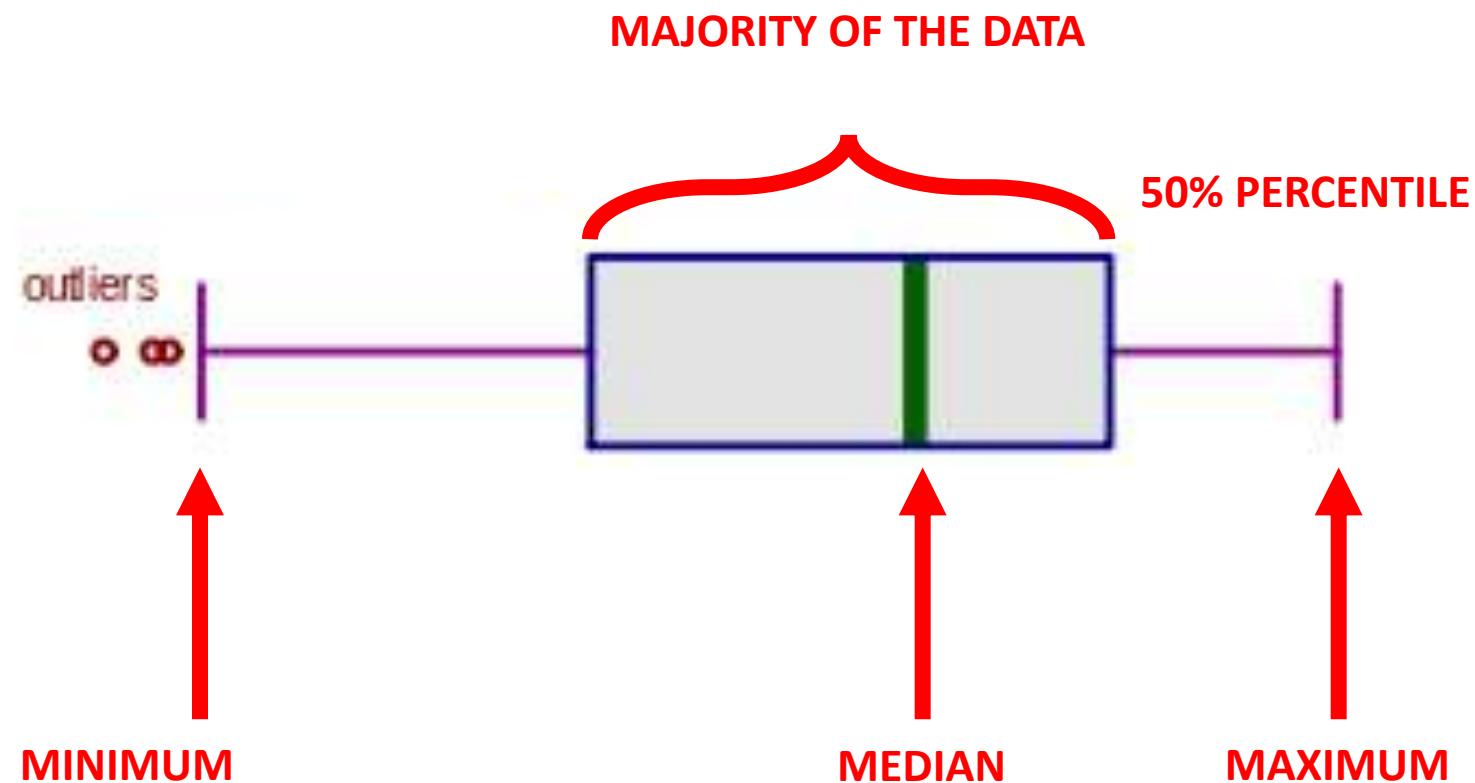
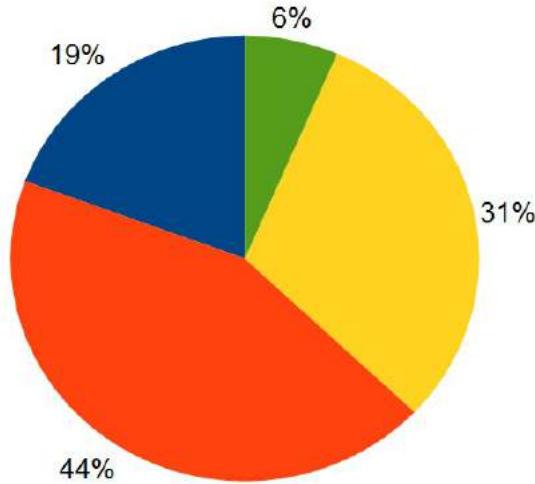


Photo Credit: https://commons.wikimedia.org/wiki/File:Elements_of_a_boxplot_en.svg

COMPOSITIONS

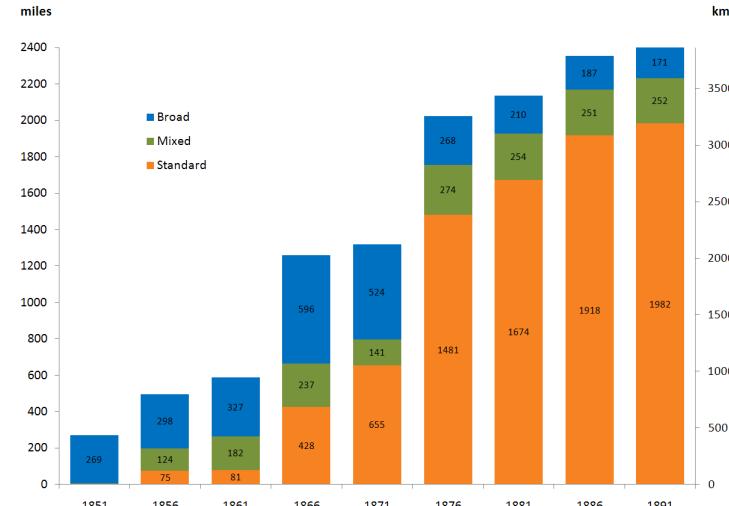


PIE CHART



STACKED BAR CHART

Broad and standard mileage operated by GWR



STACKED AREA CHART

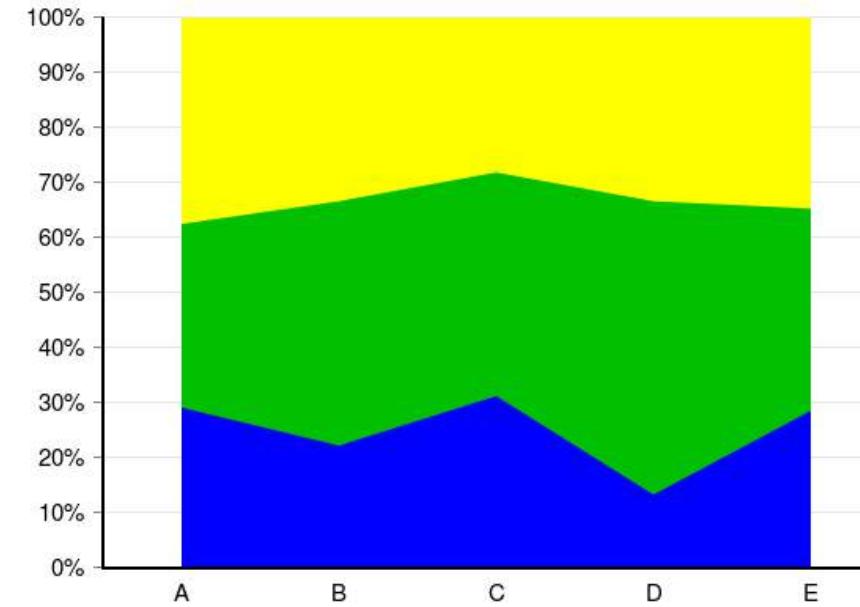


Photo Credit: https://commons.wikimedia.org/wiki/File:Broad_and_standard_mileage_operated_by_GWR.png

Photo Credit: https://commons.wikimedia.org/wiki/File:Charts_SVG_Example_12_-_Stacked_100%25_Area_Chart.svg

Photo Credit: <https://commons.wikimedia.org/wiki/File:Pie-chart.jpg>



DISTRIBUTIONS



1. NORMAL DISTRIBUTION



- A normal distribution is known as the bell curve because it looks like a bell!
- The density curve is symmetrical.
- Normal distribution is defined by its mean and standard deviation.
- Normal distribution is centered about its mean, with standard deviation indicating its spread.
- At point x , the height is represented as follows:

$$\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

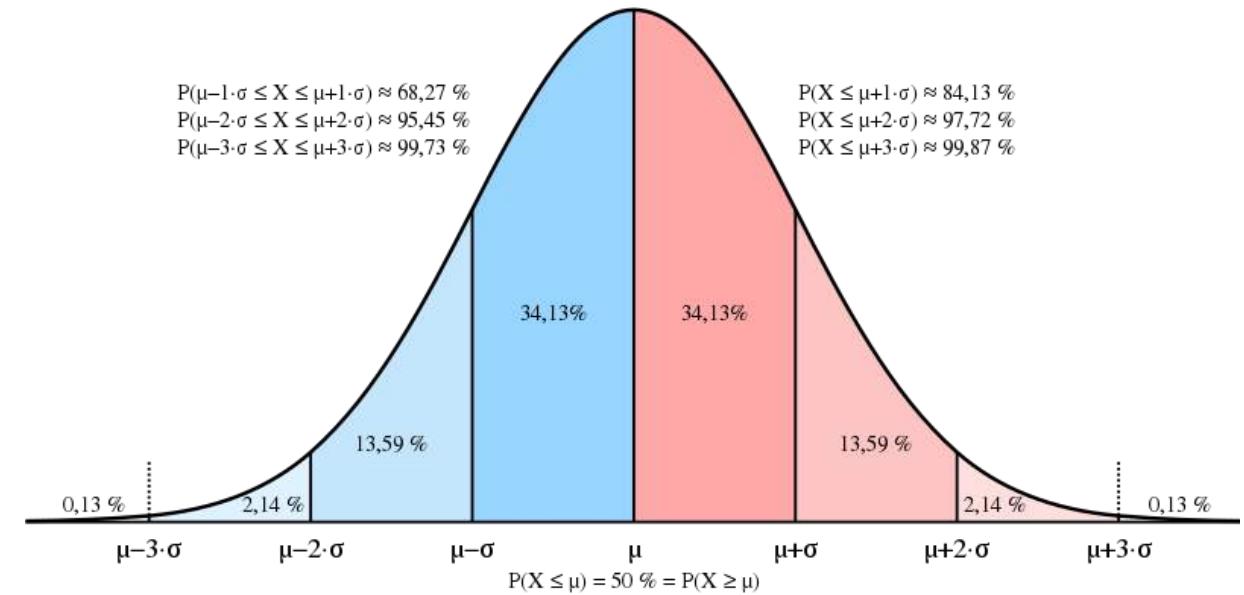


Photo Credit: https://commons.wikimedia.org/wiki/File:Normal_Distribution_Sigma.svg

1. STANDARD NORMAL DISTRIBUTION



- The Standard Normal distribution curve has:
 - Mean = 0
 - Standard deviation = 1
- We can convert data that is normally distributed to make it follow a standard normal by subtracting the mean and dividing by the standard deviation.

$$Z = \frac{X - \mu}{\sigma}$$

- For normally distributed data:
 - 68.3% of observations are within 1 standard deviation from the mean (-1,1).
 - 95% of observations are within 2 standard deviations of the mean (-2,2).
 - 99.7% of observations are within 3 standard deviations of the mean, interval (-3,3).

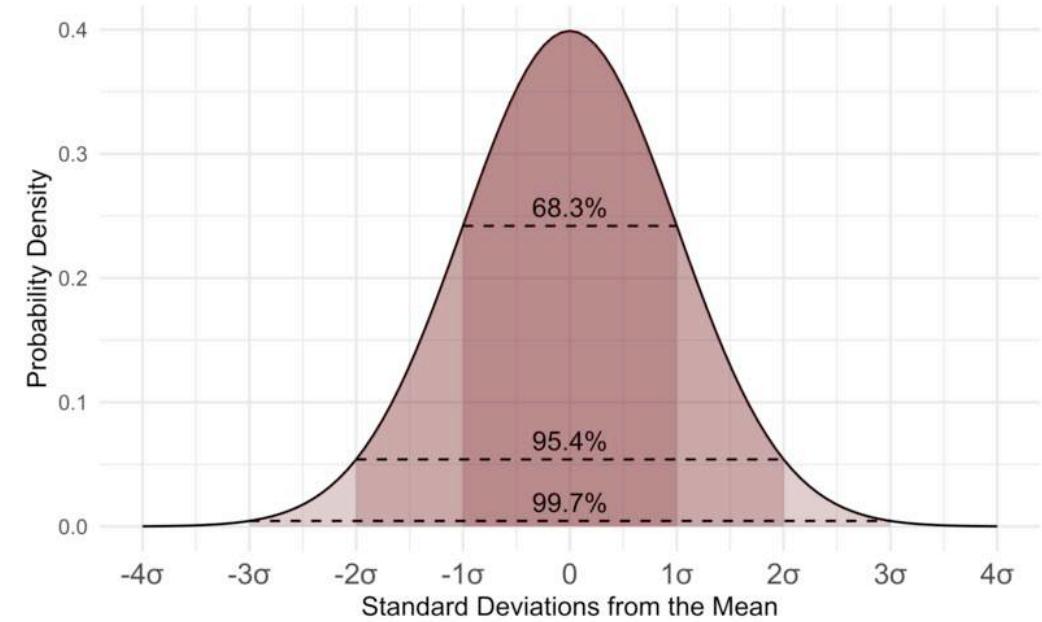


Photo Credit: https://commons.wikimedia.org/wiki/File:Standard_Normal_Distribution.png



2. POISSON DISTRIBUTION

- Poisson distribution is the discrete probability distribution of the number of events that occur in a specified period of time.
- Poisson distribution is extremely helpful for planning purposes as it enable managers to analyze customer behaviour as they visit a restaurant or store for example.
- Example: a restaurant manager want to know how many customers visit the store. He knows that the average visitors are 5 but actual number can change drastically.
- A Poisson distribution enable the manager to analyze the probability of various events.
 - Probability of 0 customers coming to the restaurant.
 - Probability of 7 or more customers visiting the restaurant.

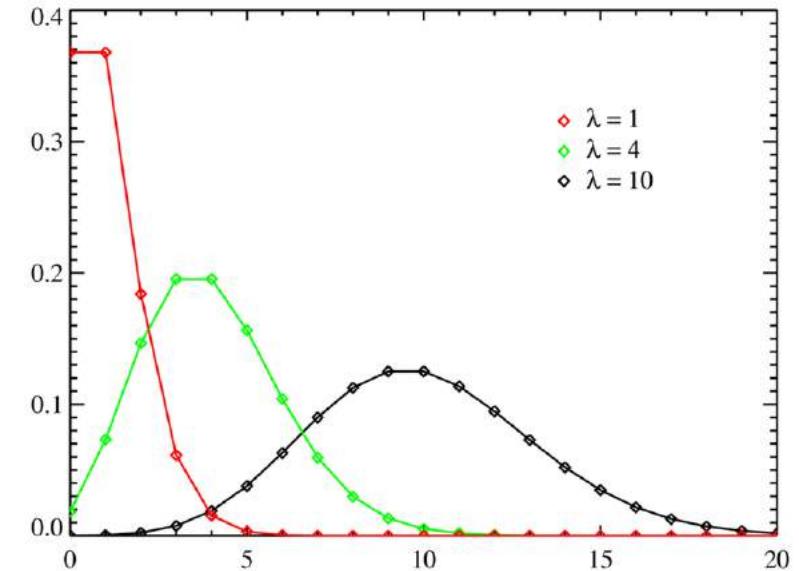


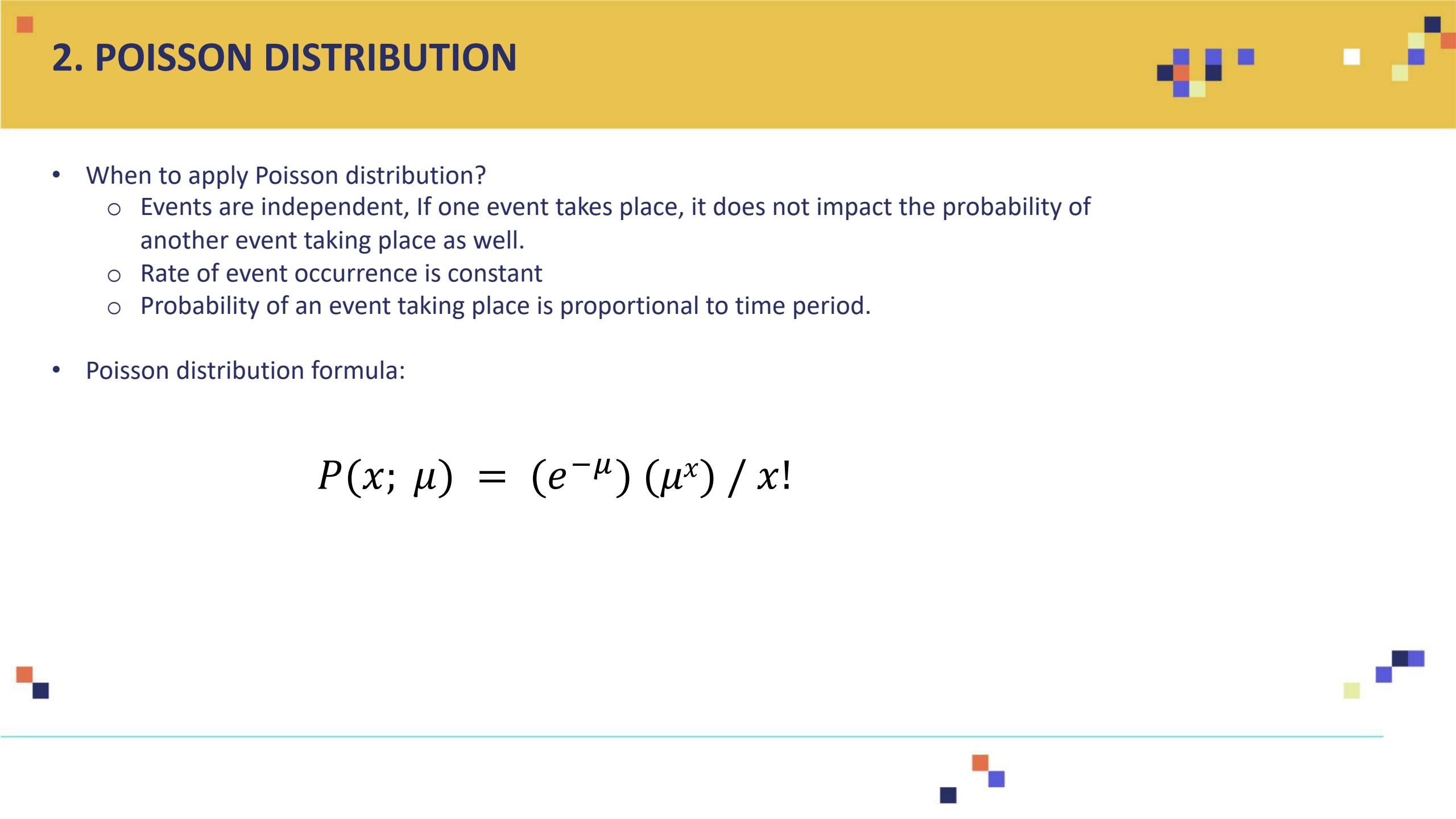
Photo Credit: https://commons.wikimedia.org/wiki/File:Poisson_distribution_PMF.png

2. POISSON DISTRIBUTION



- When to apply Poisson distribution?
 - Events are independent, If one event takes place, it does not impact the probability of another event taking place as well.
 - Rate of event occurrence is constant
 - Probability of an event taking place is proportional to time period.
- Poisson distribution formula:

$$P(x; \mu) = (e^{-\mu}) (\mu^x) / x!$$



2. POISSON DISTRIBUTION



EXAMPLE:

You are a manager of car dealership and the average number of cars sold is 2 cars per day. What is the probability that exactly 5 cars will be sold tomorrow?

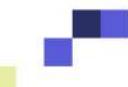
Solution

- Since we have 2 cars sold per day, $\mu = 2$.
- Since we want to know the likelihood that 5 cars being sold tomorrow, $x = 5$.
- We know that $e = 2.71828$ (constant).

Substitute in Poisson formula as follows:

$$\begin{aligned}P(x; \mu) &= (e^{-\mu}) (\mu^x) / x! \\P(5; 2) &= (2.71828^{-2}) (2^5) / 5! \\P(5; 2) &= 0.036\end{aligned}$$

Thus, the probability of selling 5 cars tomorrow is 0.036.



3. BINOMIAL DISTRIBUTION

- A binomial distribution measures the probability of success or failure outcome when the experiment is repeated several times (ex: outcomes of taking the AWS Machine Learning exam is: pass or fail).
- There are only two possible outcomes with fixed probabilities summing to one.
- For example, a coin toss has only two possible outcomes: heads or tails where the probability of each event is exactly = 0.5.
- For N trials, and probability = π , the binomial distribution is calculated as follows:

$$P(x) = \frac{N!}{x!(N-x)!} \pi^x (1-\pi)^{N-x}$$

- where $P(x)$ is the probability of x successes out of N trials, N is the number of trials, and π is the probability of success.

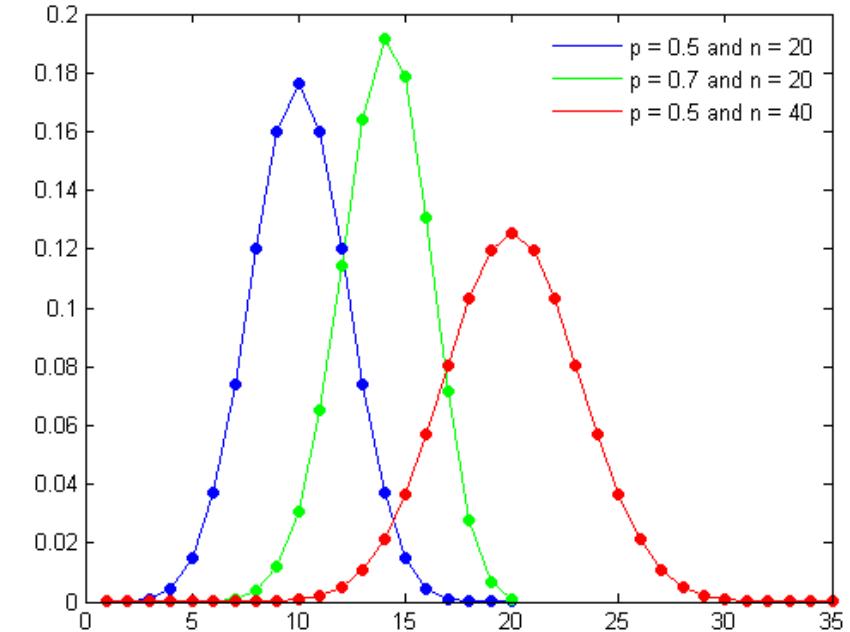


Photo Credit: https://en.wikipedia.org/wiki/File:Binomial_distribution_pdf.png

4. BERNOULLI DISTRIBUTION

- Bernoulli distribution is a special case of the binomial distribution for $n = 1$.
- Simply put, it is a binomial distribution with a single trial (one coin toss).
- Bernoulli distribution is a discrete probability distribution has only two outcomes (“Success” or a “Failure”).
- For example, when coin flipping:
 - Probability of head (success) = 0.5
 - Probability of tail (failure) = $1 - P = 0.5$
- The probability of a failure is labeled on the x-axis as 0 and success is labeled as 1.
- As shown in figure, the probability of success (1) is 0.7, and the probability of failure (0) is 0.3:
-

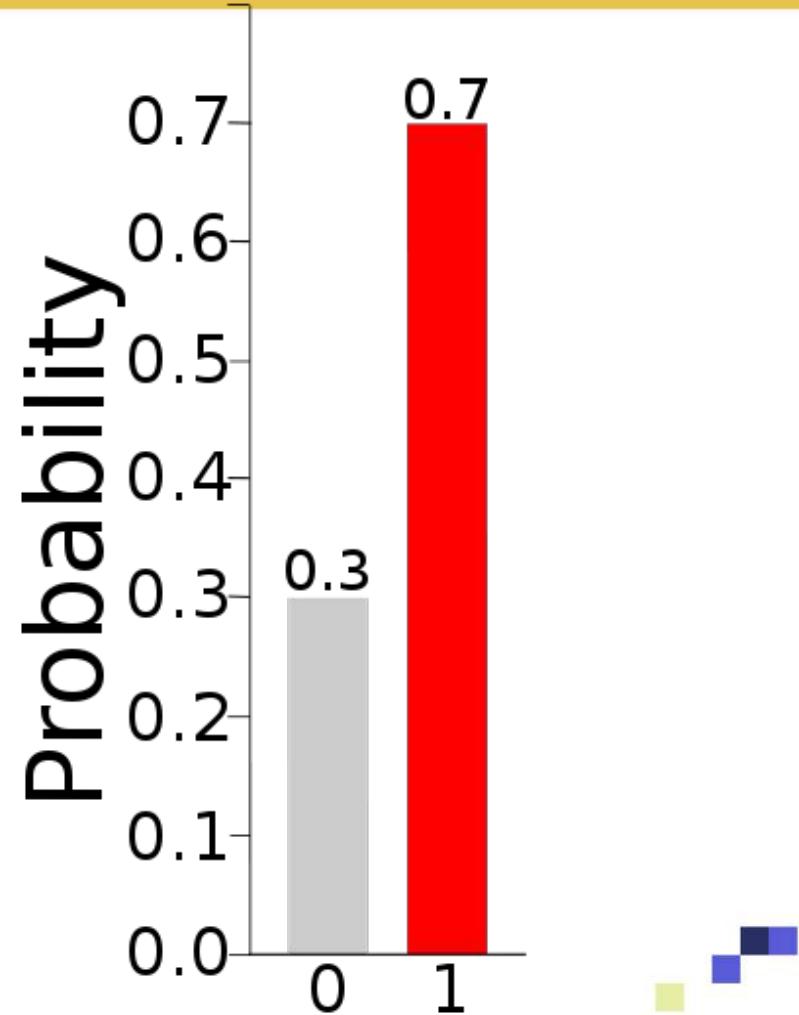
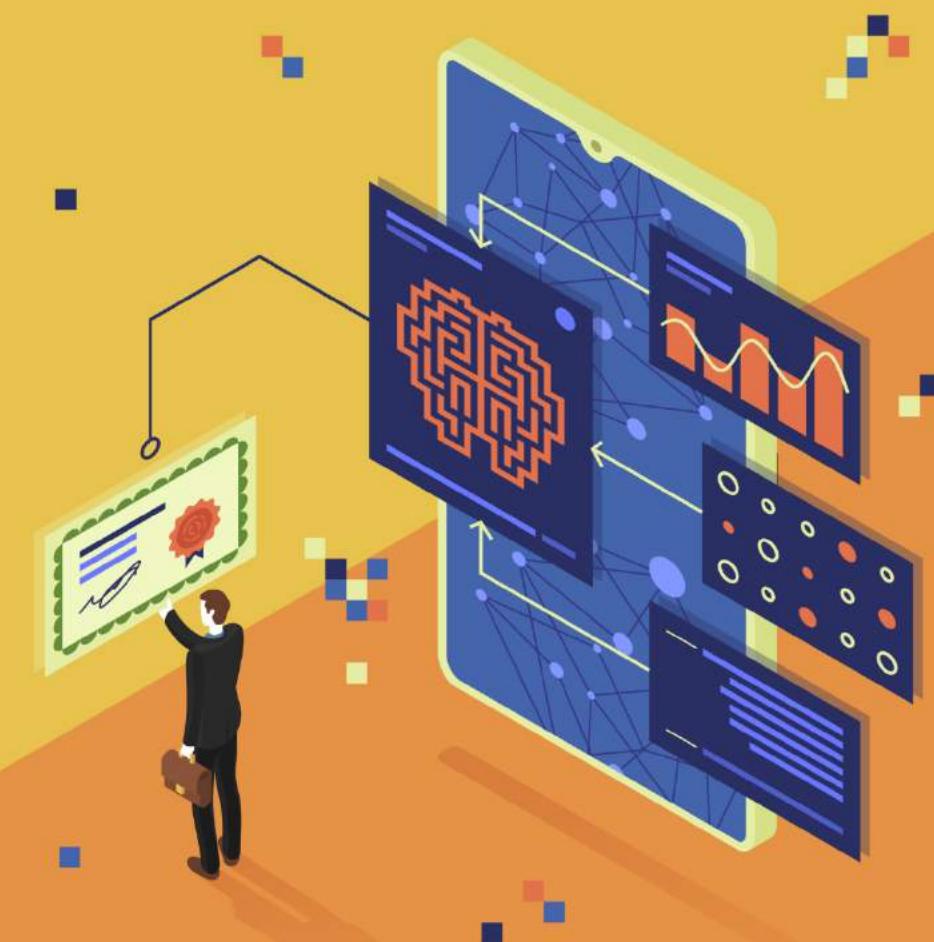


Photo Credit: https://commons.wikimedia.org/wiki/File:Bernoulli_0.7.svg

TIME SERIES



TIME SERIES: TRENDS

- In any time series, there are 4 important components:
 1. **Level:** average value in the series
 2. **Trend:** if the series is increasing or decreasing in value
 3. **Seasonality:** repeating short-term cycle in the series
 4. **Noise:** non-systematic random variation component



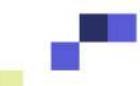
TIME SERIES: MULTIPLICATIVE VS. ADDITIVE MODELS



- There are two types of time series models:
additive and multiplicative.
 - In additive models, the seasonality, trend and error components are added.
 - In multiplicative models, these components are multiplied.



Photo Credit: <https://pixabay.com/illustrations/graph-diagram-growth-written-report-3033203/>



AWS MACHINE LEARNING CERTIFICATION



MODULE #2: EXPLORATORY DATA ANALYSIS (24% EXAM)



AWS ML CERTIFICATION EXAM DOMAINS



Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #2 OVERVIEW: WHERE ARE WE NOW?!



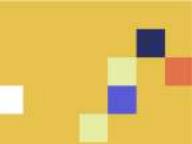
SECTION #5: JUPYTER NOTEBOOKS, SCIKIT LEARN, PYTHON PACKAGES, AND DISTRIBUTIONS

- Introduction
- Jupyter Notebooks and Scikit Learn
- Python Packages (Pandas, Numpy, Matplotlib and Seaborn)
- Distributions (Normal, Standard, Poisson, Bernoulli)
- Time Series



SECTION #6: AMAZON ATHENA, QUICKSIGHT AND ELASTIC MAP REDUCE

- Amazon Athena Features
- Amazon Athena Deep Dive (Security, Cost, and glue integration)
- Amazon QuickSight Features
- Amazon QuickSight (integration with AWS services)
- Amazon QuickSight ML insights and Use Cases
- Elastic Map Reduce (EMR)
- Apache Hadoop with EMR
- Apache Spark with EMR

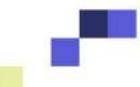


DOMAIN #1 OVERVIEW:



SECTION #7: FEATURE ENGINEERING

- Introduction to Feature Engineering
- Amazon SageMaker GroundTruth
- Feature Selection
- Scaling
- Imputation
- Outliers
- One Hot Encoding
- Binning
- Log Transformation
- Shuffling, Feature Splitting, Unbalanced Datasets
- Text Feature Engineering overview
- Bag of words, punctuation, and dates (easy ones!)
- Term Frequency Inverse Document Frequency (TF-IDF)
- N-Grams (Unigram vs. Bigram vs. Trigram)
- Orthogonal Sparse Bigram (OSB)
- Cartesian Product Transformation



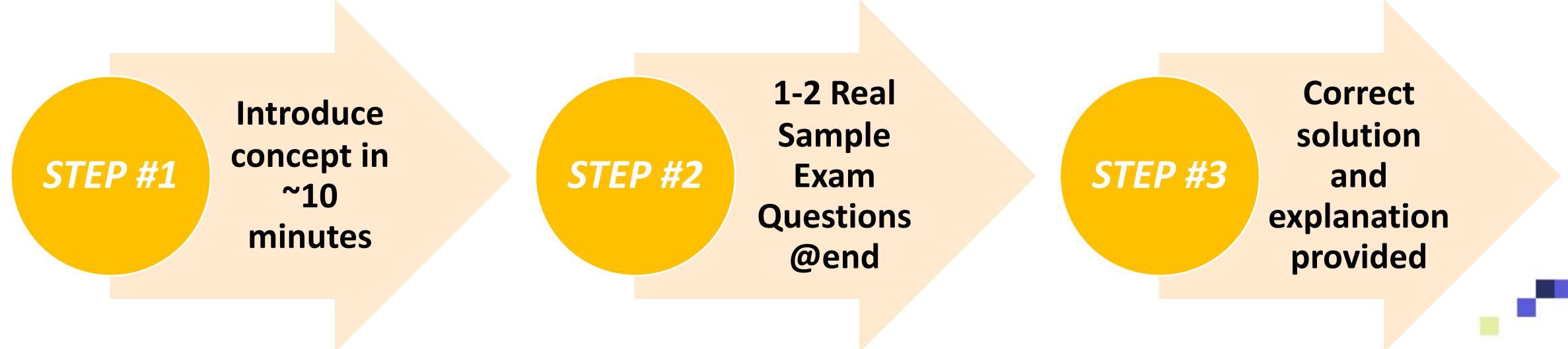
LECTURE DESIGN



- We know how hard it is to study for an exam especially if you have a busy schedule.
- This course is designed to be extremely on point and optimized to pass the exam.

No boring content. Zero unnecessary information.

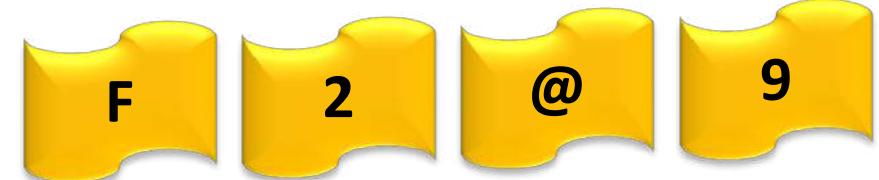
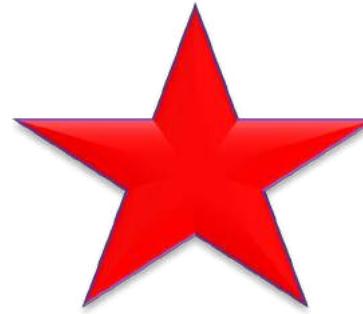
- Here's the lecture structure that we will follow:



RECALL OUR MINI CHALLENGE AND PRIZE!



- For those of you who will successfully complete the entire section and watch the videos till the end, they will receive a valuable prize!



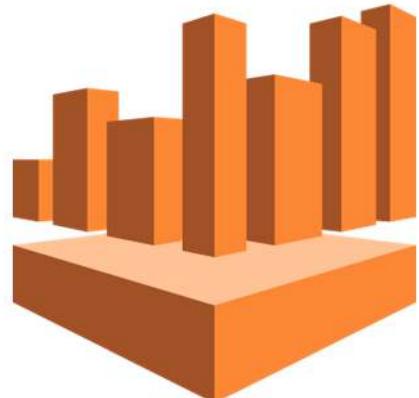
AMAZON ATHENA – PART #1



AMAZON ATHENA: FEATURES

- The Amazon Athena is a flexible, cost-effective query service.
- Amazon Athena is used for data analysis simply by accessing the data available in Amazon S3 using standard SQL requests. So if you know basic SQL, you can start using Athena now to analyze large scale datasets!
- Athena eliminates the need to have complex and expensive ETL* jobs to analyze your data.
- Athena is serverless which makes it extremely easy to use.
- Athena is very fast, results are retrieved within seconds.
- It is very cost effective, you only pay per queries you choose to run.

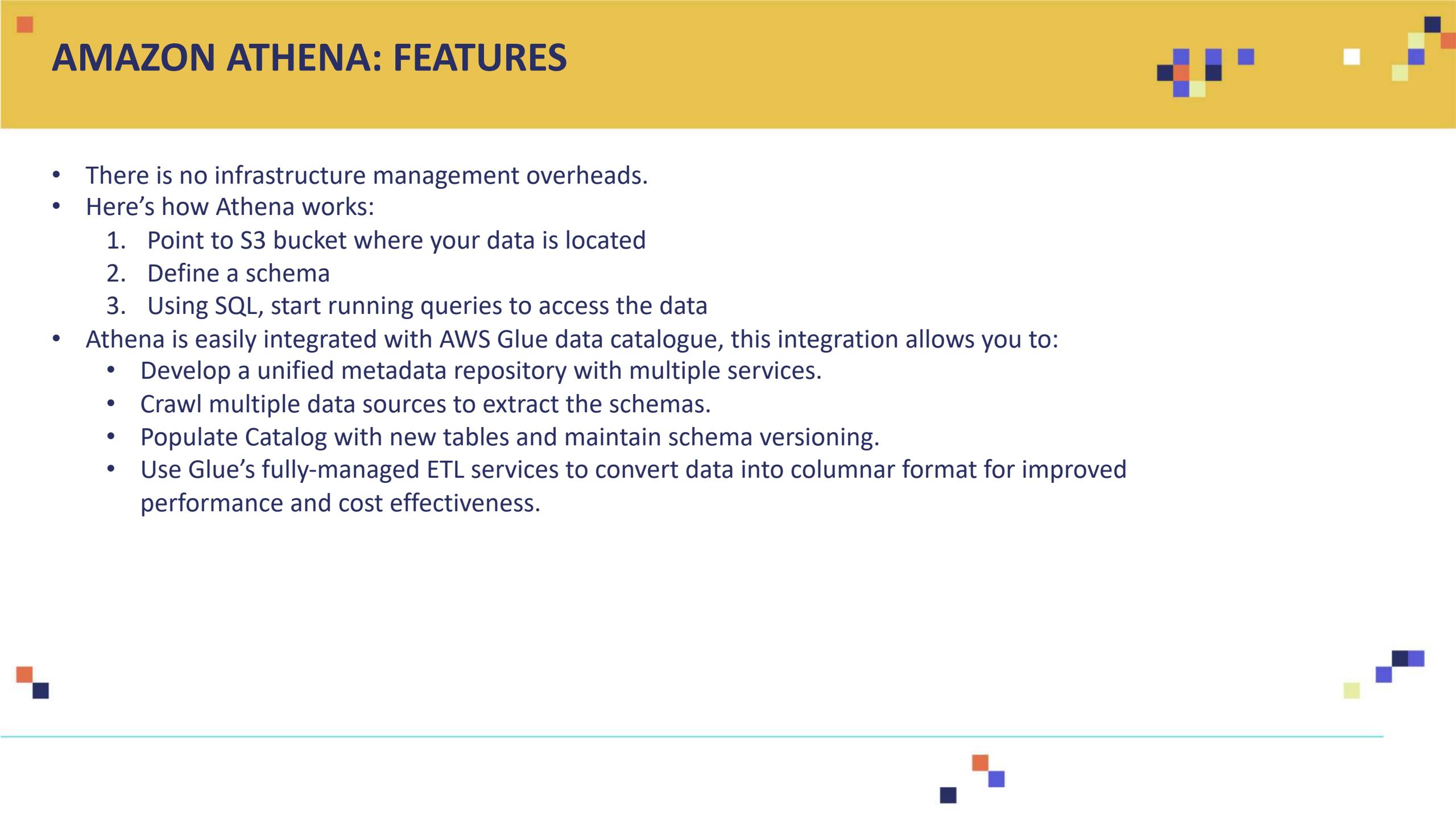
- *ETL stands for extract, transform, load.*
- *ETL combines multiple functions to extract data from one database and put it in another one.*



AMAZON ATHENA: FEATURES



- There is no infrastructure management overheads.
- Here's how Athena works:
 1. Point to S3 bucket where your data is located
 2. Define a schema
 3. Using SQL, start running queries to access the data
- Athena is easily integrated with AWS Glue data catalogue, this integration allows you to:
 - Develop a unified metadata repository with multiple services.
 - Crawl multiple data sources to extract the schemas.
 - Populate Catalog with new tables and maintain schema versioning.
 - Use Glue's fully-managed ETL services to convert data into columnar format for improved performance and cost effectiveness.



AMAZON ATHENA: FEATURES



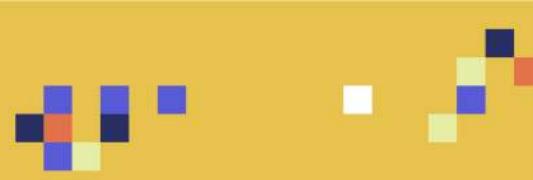
- Amazon Athena is extremely fast, it is capable for executing multiple queries using several compute resources across multiple machines.
- Since Athena relies on data available on Amazon S3, this makes the data highly available and durable.
- Amazon Athena uses Presto with ANSI SQL support.
- Athena supports several formats such as:
 - CSV
 - JSON
 - ORC
 - Avro
 - Parquet
- Athena works great with both:
 - quick querying
 - Complex analysis



Photo Credit: <https://commons.wikimedia.org/wiki/File:Data-transfer.svg>



AMAZON ATHENA: FEATURES AND BENEFITS



FAST DATA QUERY

- Serverless
- No ETL
- No server management and zero overheads.
- pay per query so very cost optimized
- Access all your data buckets in S3 with zero ETL overheads.

COST EFFECTIVE

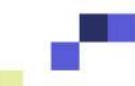
- You only pay per queries.

VERY HIGH SPEED

- Extremely fast so obtain results in seconds.
- No need to worry about having powerful compute resources to obtain fast speed. Athena manages that and executes queries in parallel.

STANDARD/OPEN

- Built on Presto.
- Runs standard SQL.



AMAZON ATHENA: FEATURES AND BENEFITS



DURABILITY & HIGH AVAILABILITY (S3 FOR DATA STORE)

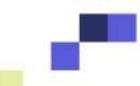
- Amazon Athena leverages multiple compute resources across several facilities so this makes it highly available.
- Athena uses Amazon S3 so the data becomes available, secure and durable.
- Athena inherits the durability of Amazon S3 which is set at 99.999999999% of objects. Your data is redundantly stored across multiple facilities and multiple devices in each facility.

VERY SECURE

- Athena is secure because it leverages:
 1. AWS Identity and Access Management (IAM) policies
 2. Access control lists (ACLs)
 3. Amazon S3 bucket policies

SEAMLESS INTEGRATION WITH OTHER AWS SERVICES (GLUE)

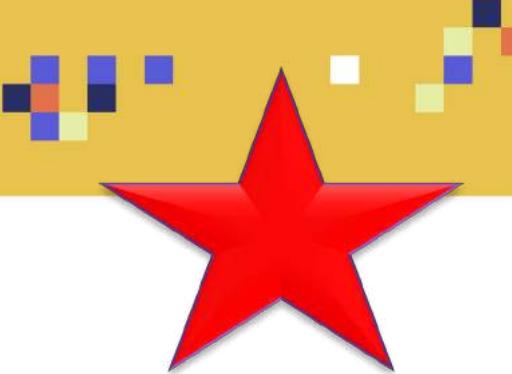
- Amazon Athena can be seamlessly and effectively integrated with AWS Glue.



AMAZON ATHENA – PART #2



ATHENA SECURITY



- Athena is secure because it leverages:
 - AWS Identity and Access Management (IAM) policies
 - Access control lists (ACLs)
 - Amazon S3 bucket policies
- Using S3 bucket polices, you can allow or prevent users from querying it using Athena.
- Using IAM policies, you can grant/deprive IAM users access to various buckets in amazon S3.
- Athena allows encryption for both client and server sides.
- Athena enable users to query encrypted data stored in Amazon S3.
- It also allows users to write encrypted results back to S3 buckets.
- Transport Layer Security (TLS) encrypts in transit data between Athena and Amazon S3.



Photo Credit: <https://www.flickr.com/photos/mikemacmarketing/35366000233>

ATHENA: COST MODEL



- Users are only charged for the queries they run.
- Users are charged based on data size scanned during each query.
- How do users reduce cost and improve performance?
 - Compressing, partitioning, or converting the data to a columnar format.
 - This dramatically reduces the time and resources Athena requires to scan and execute queries.
 - This could result in 30% to 90% cost reduction.
- Users are charged \$5 per terabyte scanned (rounded up to nearest megabyte, with a 10MB minimum per query).
- If you cancel a query, users will be charged based on the amount of data scanned.
- Athena supports Apache ORC and Apache Parquet.
- Note: Amazon S3 and Glue have separate charges.

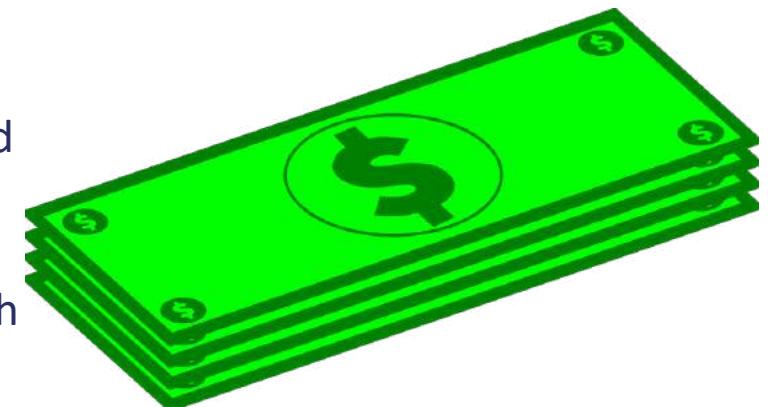
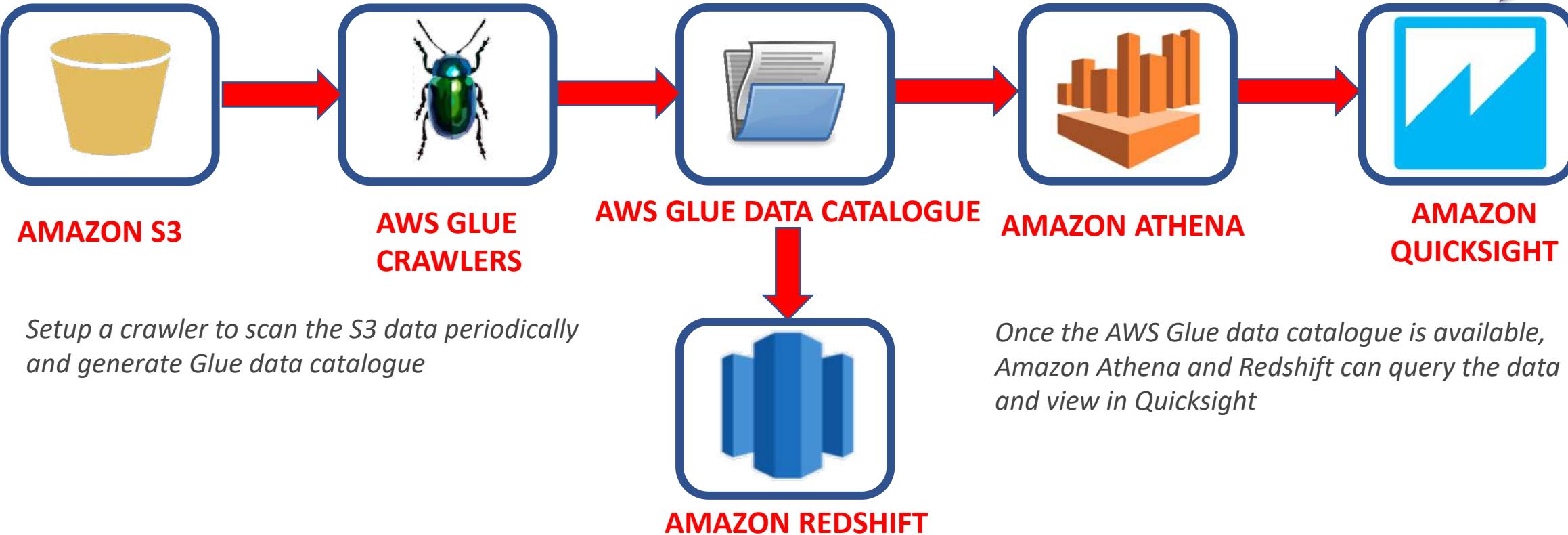
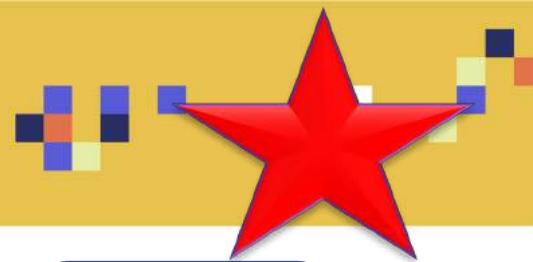


Photo Credit: <https://pixabay.com/vectors/cost-currency-dollars-four-green-151072/>



GLUE AND ATHENA



Setup a crawler to scan the S3 data periodically and generate Glue data catalogue

Once the AWS Glue data catalogue is available, Amazon Athena and Redshift can query the data and view in Quicksight

- Athena and AWS Glue Data Catalog work seamlessly together, AWS Glue can be used to create databases and tables (schema) so that Athena can use it to query the data.

Photo Credit: https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg

Photo Credit: <https://pixabay.com/illustrations/insect-insects-insect-perfection-4470664/>

Photo Credit: <https://commons.wikimedia.org/wiki/File:Document-open.svg>

Photo Credit: https://commons.wikimedia.org/wiki/File:Magnifying_glass_01.svg

Photo Credit: <http://pgfplots.net/tikz/examples/plot-markers/>

Photo Credit: <https://publicdomainvectors.org/en/free-clipart/Image-of-electricity-spark-orange-icon/31132.html>

ATHENA: IN ACTION



INSERT QUERY HERE

The screenshot shows the Amazon Athena landing page. At the top, there's a navigation bar with 'Services' and 'Resource Groups'. Below the header, the 'Amazon Athena' logo is displayed, followed by a brief description: 'Amazon Athena is a fast, cost-effective, interactive query service that makes it easy to analyze petabytes of data in S3 with no data warehouses or clusters to manage.' A prominent blue 'Get Started' button is centered. Below the button, there are three main sections: 'Select a data set' (with a database icon), 'Create a table' (with a table icon), and 'Query data' (with a person icon). Each section has a brief description and a 'Learn more' link. At the bottom, there's a 'Athena documentation and support' section with links to 'User Guide' and 'Report an issue'.

The screenshot shows the Amazon Athena Query Editor. The top navigation bar includes 'Services', 'Resource Groups', and the user 'RyanAhmed'. The main interface has tabs for 'Athena' (selected), 'Query Editor', 'Saved Queries', and 'History'. It shows 'Data sources' (awsdatacatalog), 'Database' (sampledb), and 'Tables (1)' (eb_logs). A message at the top right says: 'Before you run your first query, you need to set up a query result location in Amazon S3. Learn more'. Below the tables, there's a 'Views (0)' section with a note: 'You have not created any views. To create a view, run a query and click "Create view from query"'. At the bottom, there are buttons for 'Run query', 'Save as', and 'Create'. A red arrow points from the 'INSERT QUERY HERE' text in the previous screenshot to the 'Run query' button here.

ATHENA VS. REDSHIFT SPECTRUM



- Do you remember Amazon Redshift Spectrum?
- Redshift spectrum was used to generate queries as well directly into S3 which seem similar to what Athena does!

ATHENA	REDSHIFT SPECTRUM
Send Queries directly to Amazon S3	Send Queries directly to Amazon S3
Designed for easy ad-hoc queries into S3	Designed for users of Redshift
Does not require redshift clusters	You will require redshift clusters.

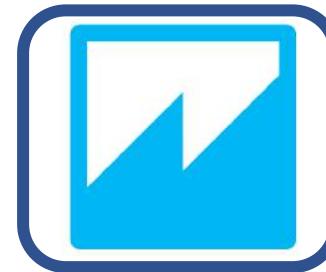
AMAZON QUICKSIGHT – PART #1



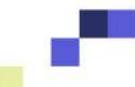
AMAZON QUICKSIGHT



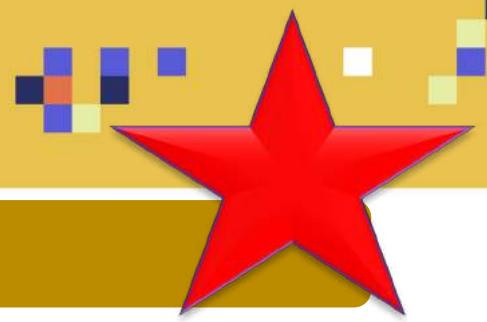
- Amazon QuickSight is cloud business intelligence (BI) tool that is used to visually share data across the entire organization.
- QuickSight is a fully managed service, and therefore it is extremely easy to use.
- It allows anyone to easily develop interactive dashboards along with ML Insights (will be introduced in details shortly!).
- Users can view the dashboard from anywhere on any device.
- Users can also embed the dashboard into any application or websites.
- Quicksight follows a per-use model so you only pay for what you use.



AMAZON QUICKSIGHT



AMAZON QUICKSIGHT FEATURES



COST EFFECTIVE, PAY PER USE

- QuickSight's follows a pay-per-session model.
- Charges are applied only when users access dashboards or reports.
- No annual subscription and zero upfront costs.
- Zero charges for users who are inactive.

HIGHLY SCALABLE, SCALE FROM 10 TO 10,000 USERS

- Serverless architecture, zero servers to manage, and no installation or setup required
- No need for capacity planning or any infrastructure cost.
- QuickSight scales automatically to 10,000 users.

ALLOW FOR EASY DATA ANALYTICS EMBEDDING

- Quicksight allows for embedding of dashboards and charts into applications.
- QuickSight visuals can be securely embedded in the application with authentication and powerful APIs.

END-TO-END BUSINESS INTELLIGENCE SOLUTION

- QuickSight works seamlessly with other AWS services available on the cloud such as RedShift, S3, Athena, Aurora, RDS, IAM, CloudTrail, Cloud Directory.
- This allows for building an end to end complete BI solution across any organization.

AMAZON QUICKSIGHT: HOW IT WORKS?

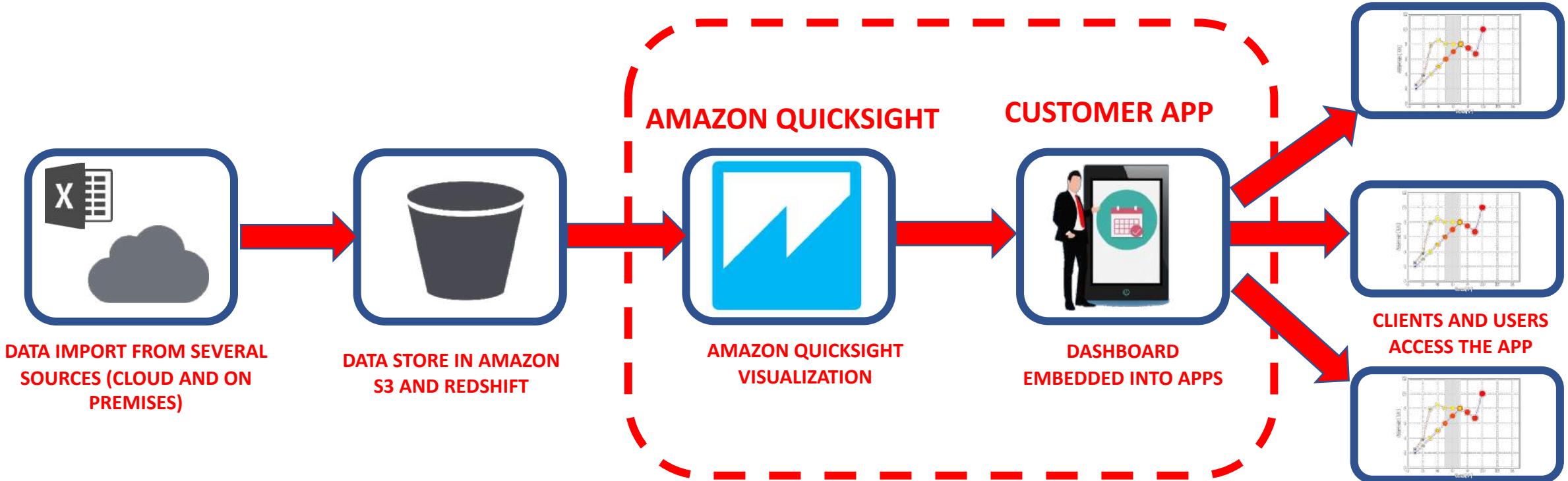


Photo Credit: https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg

Photo Credit: <https://commons.wikimedia.org/wiki/File:Document-open.svg>

Photo Credit: <http://pgfplots.net/tikz/examples/plot-markers/>

Photo Credit: https://commons.wikimedia.org/wiki/File:Microsoft_Excel_2013_logo.svg

Photo Credit: <https://pixabay.com/vectors/cloud-cloud-computing-3331240/>

Photo Credit: <https://pxhere.com/en/photo/1439573>

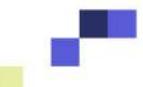
AMAZON QUICKSIGHT: DATA SOURCES



There are several sources of data available for consumption and visualization by Amazon Quicksight:

- Amazon Redshift
- Athena
- S3 or on-premises files in the following formats:
 - Excel
 - CSV
 - TSV
- EC2-hosted databases
- Aurora/RDS

- *Let's take a look at an integration example with Amazon Athena!*



AMAZON QUICKSIGHT + ATHENA + S3?

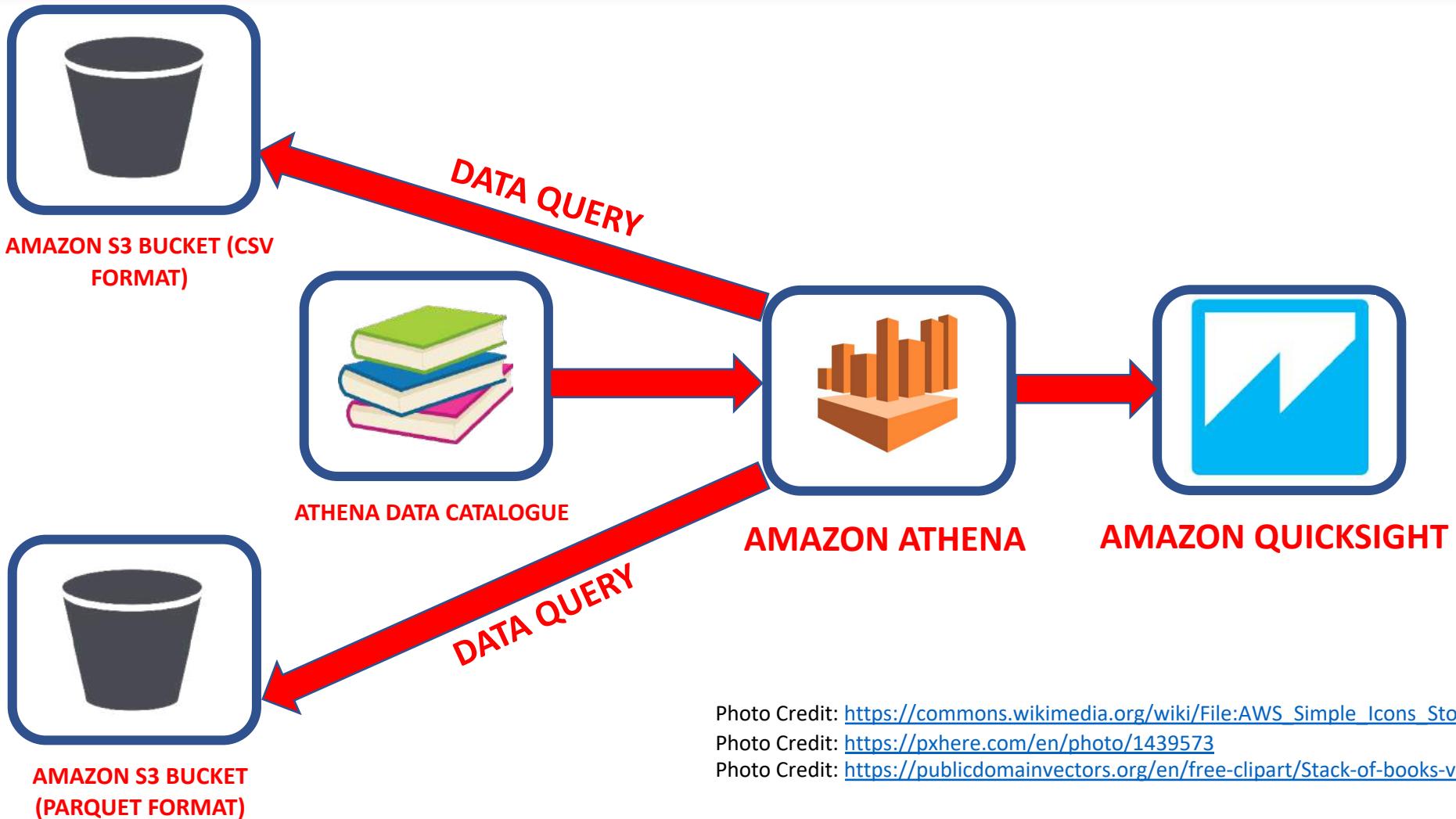


Photo Credit: https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg

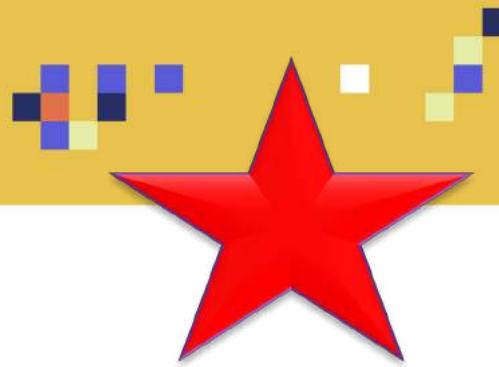
Photo Credit: <https://pxhere.com/en/photo/1439573>

Photo Credit: <https://publicdomainvectors.org/en/free-clipart/Stack-of-books-vector-clip-art/75624.html>

AMAZON QUICKSIGHT – PART #2

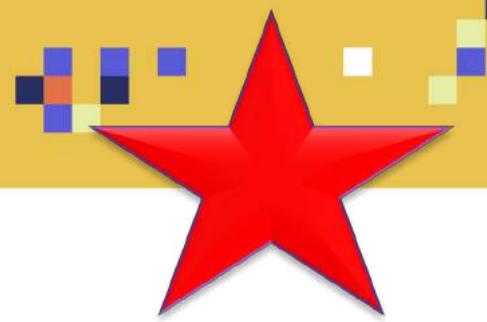


AMAZON QUICKSIGHT: SPICE



- SPICE is a fast, optimize, in-memory calculation engine for Amazon QuickSight.
- SPICE is Highly available and durable, and could be scaled to hundreds of thousands of users.
- SPICE can be used for fast, ad-hoc data visualization.
- SPICE stores the data in a system that enables fast access, data is saved until it is deleted by user.
- Once you have a QuickSight account, you can:
 - For paid users, automatically get 10 GB of SPICE capacity
 - Free users get 1 GB of SPICE capacity.
 - Purchase additional capacity.
- Instead of sending direct queries to the database, you can import data into SPICE for great performance improvements.
- If you do not use data in SPICE, you can simply delete unused data.

AMAZON QUICKSIGHT: USE CASES



PACKAGED DATA PRODUCTS (SELL DATA IN A PACKAGED FORMAT)

- QuickSight can enable enterprises to share reports and dashboards with customers.
- Data Monetization by offering a packaged product.

IMPROVE APPS BY OFFERING ANALYTICS

- Improve app experience by offering customers rich dashboards/reports while integrating Quicksight ML Insights features.

INTEGRATE DATA INTO WORKFLOWS

- By embedding Quicksight's rich dashboards/reports into portals and company sites.

"Amazon QuickSight will allow us to quickly build fast, interactive dashboards that will seamlessly integrate with our Next Gen Stats applications. With the Amazon QuickSight Readers and pay-per-session pricing, we are able to extend these secure, customized and easy to use dashboards for each Club without having to provision servers or manage infrastructure – all while only paying for actual usage. We love the direction, and look forward to expanding use of Amazon QuickSight."

Matt Swensson, VP Emerging Products - NFL

Source: <https://aws.amazon.com/quicksight/features-embedding/?nc=sn&loc=3>

AMAZON QUICKSIGHTS: EMBEDDED ANALYTICS



- QuickSight embedding allows for easy integration of analytics in apps and websites.
- It is cost effective and fully managed service.

PROVIDE A COMPLETE ANALYTICS EXPERIENCE

- Allows for creating amazing, modern dashboards that uses QuickSight's visualization and analytics capabilities(ML Insights) and Auto-Narratives.

RICH APIs AND SDK*

- Powered by JavaScript SDK, it could allow for integration between the application and the embedded dashboards powered by QuickSight.

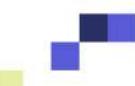
A SCALABLE, GLOBAL PLATFORM

- QuickSight is a fully managed, secure service.
- Supports 10 languages with exceptional end-to-end data security.
- Fast performance powered by SPICE.

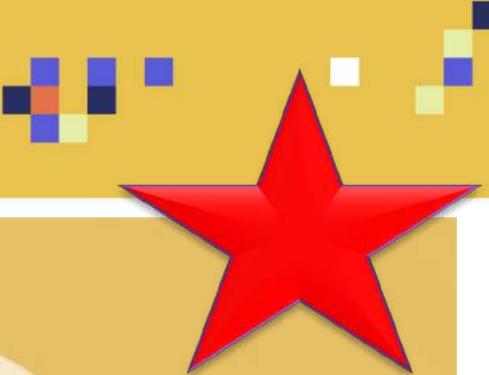
EASY TO DEVELOP AND MAINTAIN

- Easily develop dashboard templates that you can embed into your application.

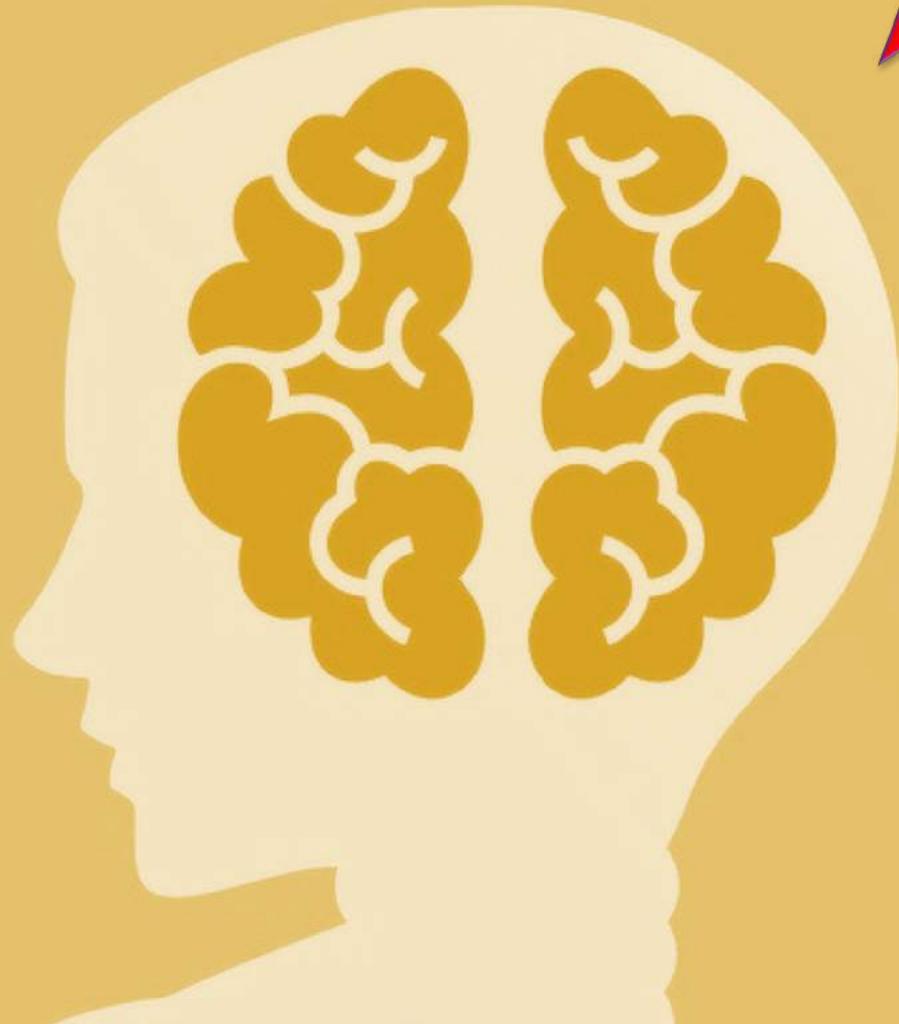
**SDK is a collection of software used for app development targeted at an operating system such as Windows 10 SDK, Mac OS X SDK, and iPhone SDK*



AMAZON QUICKSIGHT: ML INSIGHTS



- Amazon QuickSights offer an integrated ML insights features.
- ML insights offer the following:
 1. Find data insights:
 - QuickSight's ML offers an anomaly detection tool.
 - The tool can perform anomaly detection and notify management.
 2. Forecasting:
 - Quicksight's ML tool can perform accurate forecasting to predict critical business metrics.
 3. Generate Auto-Narratives:
 - Powered by NLP, QuickSight's ML can offer narratives and tell a story!



AMAZON QUICKSIGHT: PRICING



- Check out pricing here: <https://aws.amazon.com/quicksight/pricing/?nc=sn&loc=4>
- The pricing model varies if you are an author or reader.
- For authors:
 - Annual subscription
 - Standard: \$9/user/month
 - Enterprise: \$18/user/month
 - Additional SPICE capacity more than 10GB
 - \$0.25 (standard) / GB / month
 - \$0.38 (enterprise) / GB / month
- For the ML-powered Anomaly Detection, there is a pricing model.

VOLUME TIER	\$ PER THOUSAND METRICS PROCESSED PER MONTH
First 1,000,000 metrics processed	\$0.50
Next 9,000,000 metrics processed	\$0.25
Next 90,000,000 metrics processed	\$0.10
> 100,000,000 metrics processed	\$0.05

WHAT DOES QUICKSIGHT LOOK LIKE?

The screenshot shows the Amazon QuickSight interface. At the top, there is a navigation bar with the QuickSight logo, a search bar containing "Search for analyses, data sets, and dashboards", and a location indicator for "N. Virgi...". Below the navigation bar, there are buttons for "New analysis" and "Manage da". A horizontal menu bar contains four items: "All analyses" (which is selected and highlighted in blue), "All dashboards", "Favorites", and "Tutorial videos".

The main content area is titled "All analyses" and displays four sample analyses as cards:

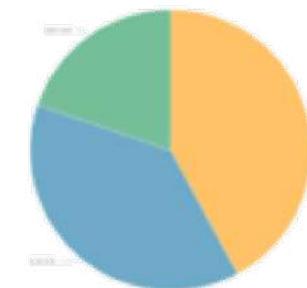
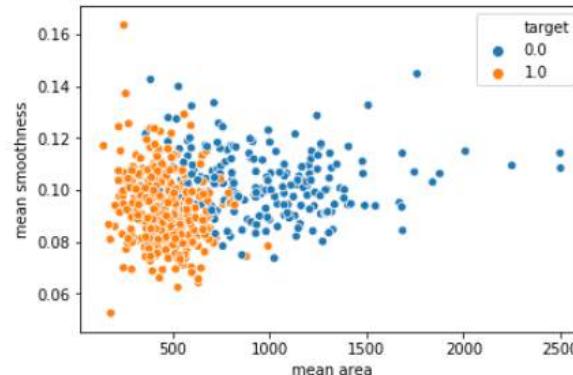
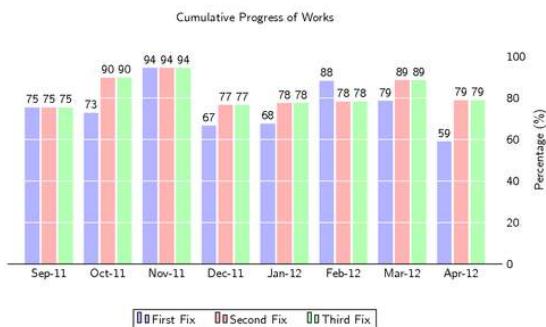
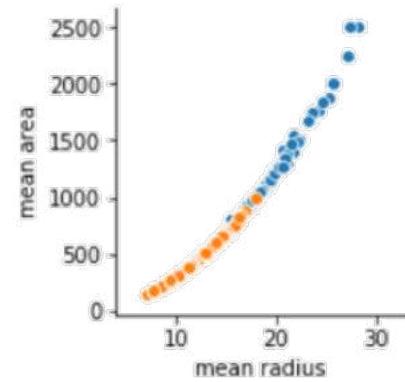
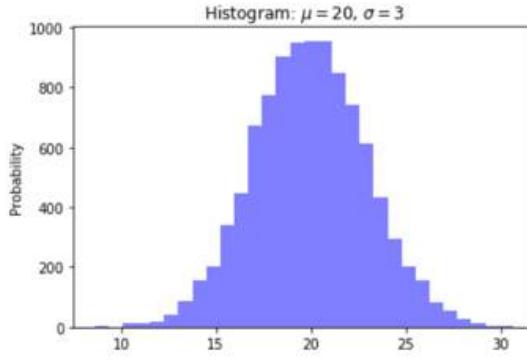
- Web and Social Media Analytics analysis**: Shows a bar chart with three orange bars. A "SAMPLE" button is at the bottom left.
- People Overview analysis**: Shows a pie chart divided into three segments (blue, green, yellow).
- Sales Pipeline analysis**: Shows a horizontal bar chart with several bars of varying lengths.
- Business Review analysis**: Shows a stacked area chart with multiple layers of different colors (teal, yellow, blue, green).

A dropdown menu labeled "Last updated (newest first)" is visible on the right side of the analysis list.

[Sample Dashboard #1: https://aws.amazon.com/blogs/big-data/embed-interactive-dashboards-in-your-application-with-amazon-quicksight/](https://aws.amazon.com/blogs/big-data/embed-interactive-dashboards-in-your-application-with-amazon-quicksight/)

[Sample Dashboard #2: https://aws.amazon.com/blogs/big-data/amazon-quicksight-updates-multiple-sheets-in-dashboards-axis-label-orientation-options-and-more/](https://aws.amazon.com/blogs/big-data/amazon-quicksight-updates-multiple-sheets-in-dashboards-axis-label-orientation-options-and-more/)

WHAT VISUALIZATIONS CAN WE MAKE?



	first	last	email	postal	gender	dollar
0	Joseph	Patton	daafeja@boh.jm	M6U 5U7	Male	\$2,629.13
1	Noah	Moran	guutodi@bigwoc.kw	K2D 4M9	Male	\$8,626.96
2	Nina	Keller	azikez@gahew.mr	S1T 4E6	Male	\$9,072.02

- **Bar Charts:** Comparison
- **Line graphs:** trends with time
- **Scatter plots/heat maps:** correlation
- **Pie chart:** aggregation
- **Pivot tables:** tabular data

Photo Credit: <http://pgfplots.net/tikz/examples/multi-series-bar-chart/>

QUICKSIGHT SECURITY



- Amazon QuickSight offers a secure service to allow users to access interactive dashboards from any device.
- Amazon QuickSight offers multiple security features such as:
 - Role-based access control
 - Microsoft Active Directory integration
 - AWS CloudTrail auditing
 - Single sign-on using AWS Identity and Access Management (IAM) and third-party solutions
 - Private VPC subnets (Elastic Network Interface, AWS Direct Connect)
 - Data backup
 - Multifactor authentication on the user account
 - Row level security
- Amazon QuickSight can also support FedRAMP, HIPAA, PCI DSS, ISO, and SOC compliance.



Photo Credit: <https://pixabay.com/vectors/lock-padlock-green-locked-33495/>

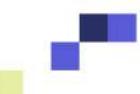
QUICKSIGHT SECURITY



- There are two levels of security of the cloud and security in the cloud:
 - Security of the cloud:
 - ❖ AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud.
 - ❖ Security levels are in compliance and being regularly tested by third-party auditors.
 - Security in the cloud:
 - ❖ Enterprises are responsible for the data sensitivity, the requirements, and applicable laws and regulations.



Photo Credit: <https://pixabay.com/vectors/lock-padlock-green-locked-33495/>



ELASTIC MAP REDUCE (EMR) – PART #1



WHAT IS EMR?

- EMR Stands for Elastic Map Reduce.
- Amazon EMR is big data platform that allows for data processing in a fast, easy and cost optimized way.
- Leverages Apache Spark, Apache Hive, Apache HBase, Apache Flink, and Presto.
- EMR empower developers to use:
 - Short Single-purpose clusters that are scalable based on demand.
 - Long term highly available clusters.
- EMR allows for:
 - Dynamic scalability using Amazon EC2
 - Storage of Amazon S3
- Using Jupyter-based EMR Notebooks, developers can work with data anywhere in AWS such as Amazon S3, Amazon DynamoDB, and Amazon Redshift.
- Works great with machine learning, data transformations (ETL), and deep learning.



WHAT IS EMR?

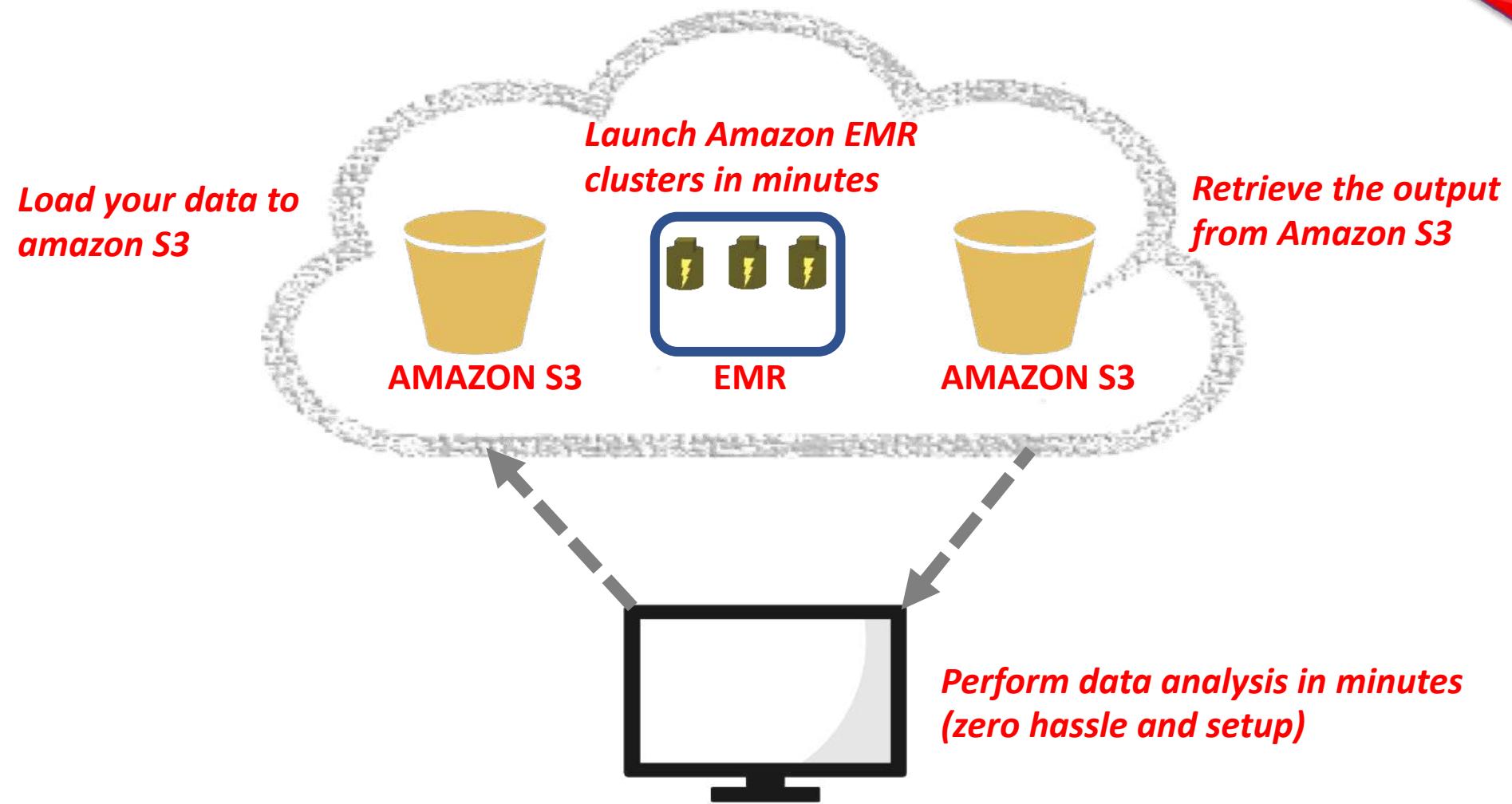


Photo Credit: <https://publicdomainvectors.org/en/free-clipart/Grey-cloud-icon-vector-image/19199.html>

Photo Credit: <https://www.needpix.com/photo/824215/computer-pc-monitor-screen-computer-monitor-computer-screen-personal-computer-technology-laptop>

EMR FEATURES – PART #1



EASE OF USE

- Developers could launch clusters in a very short period of time.
- Does not require any infrastructure setup or node provisioning.
- Works with serverless Jupyter notebooks (EMR Notebooks) so developers can analyze/visualize data easily.

COST EFFECTIVE

- EMR pricing model is effective, pay per instance per time. .
- Example: Launch multi-node EMR clusters and run applications such as Apache Spark, and Apache Hive and pay \$0.15 per hour.
- Great savings can be achieved by leveraging Amazon EC2 Spot and Reserved Instances.

ELASTIC

- EMR gives you the flexibility to elastically change the compute and storage offering extremely cost effective way.
- EMR allows for instantiating whatever number of compute instances (you can increase/decrease at anytime) to work with massive amount of data at any scale.
- Auto Scaling feature allows for automatically managing cluster sizes based on utilization).



EMR FEATURES – PART #2



ENHANCED RELIABILITY

- No need to waste time monitoring instances.
- Multiple master nodes are available to ensure high reliability.
- EMR automatically and periodically monitors instances, assesses their health and replace failed ones.
- EMR manages software updates resulting in minimum issues and less maintenance.

IMPROVED SECURITY

- EC2 firewall settings are automatically configured by EMR.
- Launches clusters Amazon Virtual Private Cloud (VPC)
- Server-side and client side encryption in S3 used with EMRFS (an object store for Hadoop on S3).
- AWS Key Management Service.
- Allows for In-transit and at-rest encryption
- Authentication with Kerberos

INCREASED FLEXIBILITY

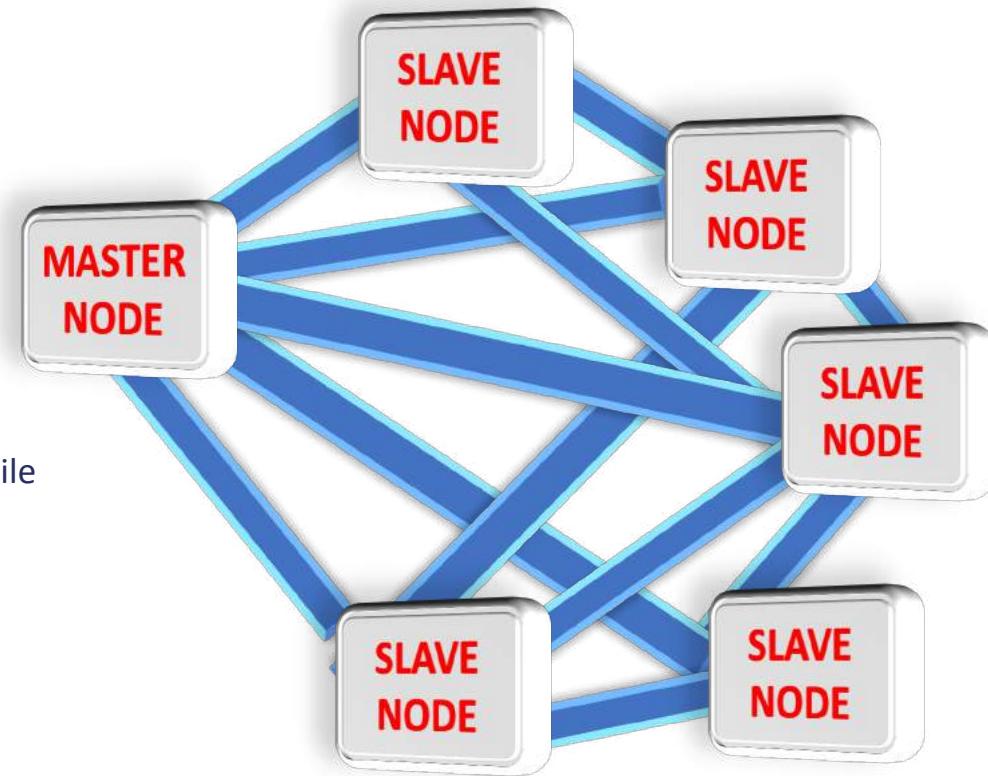
- Increased flexible Cluster Control
- Developers have root access to instances which allow for customization with bootstrap actions.
- Launch EMR clusters with custom Amazon Linux AMIs, and reconfigure running clusters on the fly.

LOREM IPSUM



EMR CLUSTER

- A cluster consists of a group of Amazon Elastic Compute Cloud (Amazon EC2) instances.
- Instances are known as nodes.
- Several node type are available (runs different SW):
 - **Master node:**
 - Big boss! Node that manages the cluster.
 - Distribute data and tasks
 - Tracks status of tasks and monitors health cluster.
 - **Core node:**
 - A slave node
 - Runs tasks and store data in the Hadoop Distributed File System (HDFS) on the cluster.
 - **Task node:**
 - A slave node.
 - Optional
 - It only run tasks.
 - Spot instances.
 - No risk of data losses if task node is removed



**“CLUSTER WITH ONE MASTER NODE
AND FIVE SLAVE NODES”**

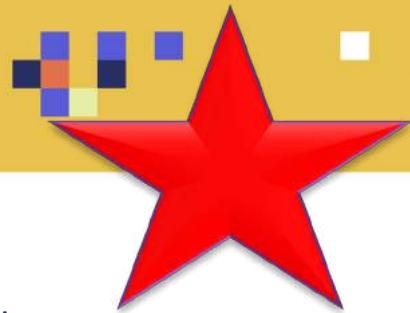
EMR NOTES:

- EMR leverages several AWS services such as:
 - **Amazon EC2**: run instances (nodes) in the cluster
 - **Amazon S3**: data storage and data output.
 - **Amazon CloudWatch**: cluster performance monitoring and generating alarms
 - **IAM**: grant users permissions
 - **AWS CloudTrail**: audit requests made to the service
 - **AWS Data Pipeline**: clusters scheduling and initiating
 - **Amazon VPC**: virtual network configurations for enhanced security.
- When configuring EMR, you can:
 - Use Spot instances and run task nodes
 - Use reserved instances with long running clusters

ELASTIC MAP REDUCE (EMR) – PART #2



WHAT IS SPOT INSTANCE?



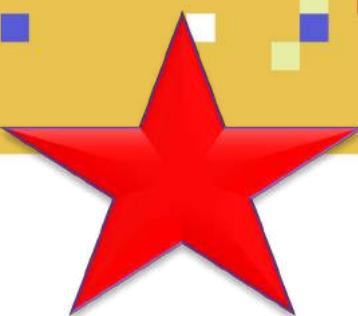
- A Spot offers a lower price compared to an on-Demand instance.
- “Spot price” is the price you pay for the spot instance and is adjusted based on availability zone and demand.
- The spot instance will run when:
 - When capacity permits.
 - When the maximum hourly price set is more than the Spot price, the instance will run.
- Choose Spot instances when:
 - You have limited resources
 - When you have some degrees of flexibility. Note that your applications can be interrupted.
 - You want to perform optional tasks, data analysis, and batch jobs.



Photo Credit: <https://pxhere.com/en/photo/747848>

EMR DATA STORAGE

- Amazon EMR and Hadoop offer several file systems when processing a cluster (you can use multiple of them).
- HDFS and EMRFS are the most common with Amazon EMR.

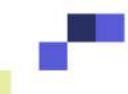


File System	Prefix	Description
HDFS	hdfs://	<ul style="list-style-type: none">HDFS is a distributed, scalable, and portable file system for Hadoop.FastStorage is reclaimed when cluster ends.
EMRFS (EMR FILE SYSTEM)	s3://	<ul style="list-style-type: none">EMRFS is an implementation of the Hadoop file system used for reading and writing regular files from Amazon EMR directly to Amazon S3.EMRFS allows for storing files in S3 for use with Hadoop while allowing features like Amazon S3 server-side encryption.EMRFS allows for EMRFS consistent view which is a feature that leverages DynamoDB to ensure data consistency.
local file system		<ul style="list-style-type: none">locally connected disk.
Amazon S3 block file system	s3bfs://	<ul style="list-style-type: none">legacy file storage system.Not recommended.

EMRFS CONSISTENT VIEW



- EMRFS consistent view overcomes some of the issues encountered due to S3 Data Consistency model.
- Scenario: If data is added to Amazon S3 followed by another operation that include “list objects”, you may encounter issues such as incomplete list. More common in multistep ETL (Extract transform Load) operations.
- EMRFS consistent view overcomes this issue by allowing the EMR cluster to check for list and read after write consistency.
- EMRFS Consistent view ensures data consistency by using an Amazon DynamoDB database to store metadata and ensure consistency with S3.
- There is a fee associated with EMRFS consistent view.



EMR SECURITY



- EC2 firewall settings are automatically configured by EMR.
- Launches clusters Amazon Virtual Private Cloud (VPC)
- Server-side and client side encryption in S3 used with EMRFS (an object store for Hadoop on S3).
- AWS Key Management Service and IAM polices/roles.
- Allows for In-transit and at-rest encryption
- Authentication with Kerberos

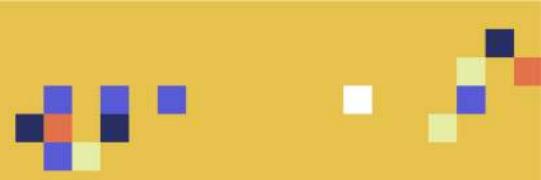


EMR NOTEBOOKS



- Apache allows for using Amazon EMR Notebooks which are serverless Jupyter notebooks.
- EMR notebooks are launched via AWS EMR console.
- EMR notebooks are used with Amazon EMR clusters running Apache Spark.
- EMR notebooks contents such as equations, models, and code are saved on Amazon S3 separately from the cluster that runs the code.
- An Amazon EMR cluster is needed to run codes in an EMR notebook, but notebooks are not locked with a specific cluster.
- EMR Notebooks allow for cluster provisioning and could be hosted inside a VPC.
- Similar to Zeppelin.

EMR SPOT INSTANCES



- Spot instances could be used for task nodes to save money.
- Spot instances are not recommended for core and master since it might result in data loss.

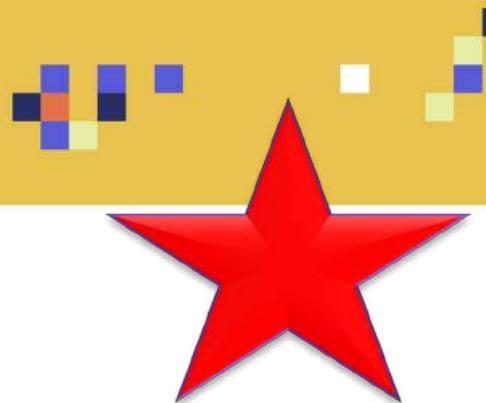
Master node:

- m4.large or m4.xlarge

Core & task nodes:

- m4.large is recommended

WHEN SHOULD I USE AWS GLUE VS. AWS EMR?



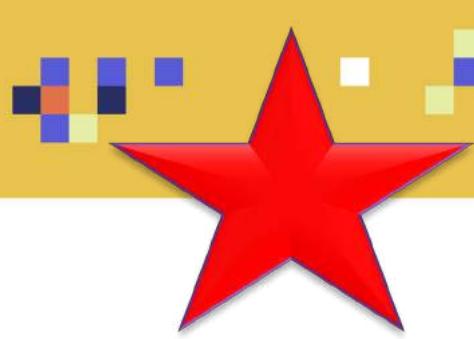
AWS GLUE:

- Glue is an ETL service that runs on a serverless Apache Spark environment.
- As a user, you do not have to configure or manage resources.
- Glue contains a data catalogue for ETL that could be used with Athena and Redshift Spectrum.
- AWS Glue ETL jobs uses Scala or Python.

AWS EMR:

- Amazon EMR allows for direct access to Hadoop environment.
- Amazon EMR allows for flexible, lower-level access to tools beyond Spark.

WHEN TO USE ATHENA COMPARED TO AMAZON EMR AND REDSHIFT?



Redshift:

- Data warehousing services
- Offers fastest query performance for enterprise reporting and business intelligence workloads
- Can be used with extremely complex SQL with several sub-queries

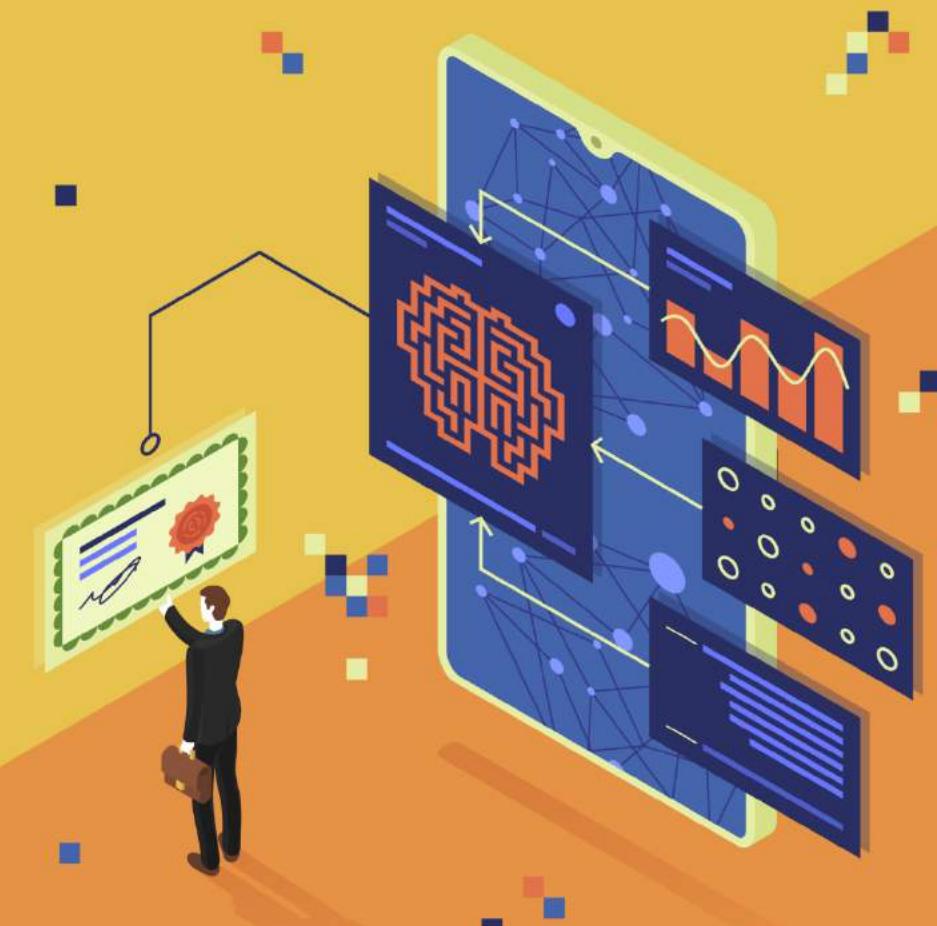
EMR:

- Sophisticated data processing framework
- Allows for running highly distributed processing frameworks such as Hadoop, Spark, and Presto in a simple and cost effective ways.
- Offers large flexibility, users are able to specify memory, compute and storage requirements to meet their specific needs.

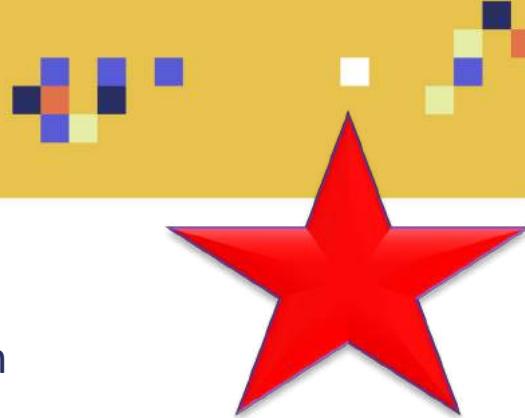
Athena:

- Query service
- Provides easiest way to run ad-hoc serverless queries for data available in S3 buckets.
- There is absolutely no need to manage any servers.

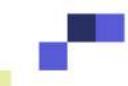
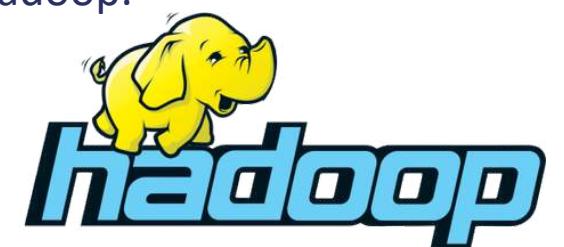
APACHE HADOOP ON AMAZON EMR



APACHE HADOOP ON AMAZON EMR



- Apache Hadoop is an open source tool to process large data efficiently.
- Hadoop allows for hardware clustering to process many datasets in parallel (instead of relying on one computer).
- Amazon EMR allows for launching elastic clusters of Amazon EC2 instances running Hadoop.
- Hadoop includes:
 - **MapReduce:** execution framework
 - **YARN:** resource manager
 - **HDFS:** distributed storage
- Amazon EMR includes EMRFS which allows Hadoop to use Amazon S3 for storage.
- Amazon EMR could be used to install tools such as Hive, Pig, Hue, Ganglia, Oozie, and HBase on your cluster.



APACHE HADOOP ON AMAZON EMR: HADOOP AND BIG DATA



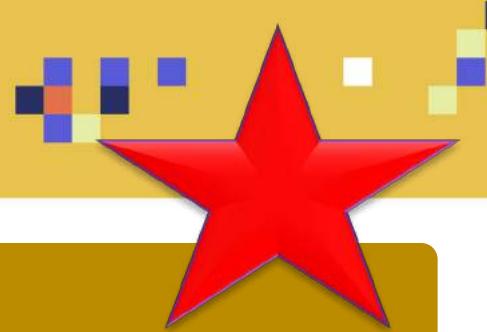
- Hadoop is extremely scalable and process data in parallel which makes it perfectly suited for big data processing.
- Hadoop is very durable and available.
- Scalability is achieved by adding more servers to the Hadoop cluster.
- Amazon EMR could be used to create and configure a cluster of Amazon EC2 instances that run Hadoop.



Photo Credit: <https://www.needpix.com/photo/514304/big-data-data-analysis-information-data-analysis-analytics-data-analytics>



APACHE HADOOP ON AMAZON EMR: ELEMENTS



MAPREDUCE AND YARN

- Hadoop MapReduce execution engine works by dividing a job into smaller ones and distribute them across many nodes in Amazon EMR Cluster.
- YARN is a resource manager that tracks resources in a cluster, and it ensures that these resources are dynamically allocated.

STORAGE USING AMAZON S3 AND EMRFS

- EMR File System (EMRFS) could be used to leverage S3 storage for Hadoop. S3 is scalable and decoupled from compute resources.
- EMRFS is optimized for Hadoop to read and write in parallel to Amazon S3.
- Allows for object encryption with Amazon S3 for both server-side and client-side.

ON-CLUSTER STORAGE WITH HDFS

- Hadoop offers a distributed storage system named HDFS (the Hadoop Distributed File System) that stores data on disk on clusters in large blocks.
- HDFS offers a 3x replication factor for improved speed and availability.

APACHE HADOOP ON AMAZON EMR: UNIQUE FEATURES



HIGH SETUP SPEED

- Hadoop clusters could be launched in minutes using Amazon EMR cluster

MINIMUM ADMIN WORK

- Amazon EMR manages all the hassle associated with configuring and ensuring security of Hadoop.

SEAMLESS INTEGRATION WITH OTHER AWS SERVICES

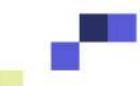
- Hadoop is integrated with Amazon S3, Kinesis, Redshift, and DynamoDB.
- AWS Glue Data Catalog is used as a metadata repository for Apache Hive and Apache Spark.

COST EFFECTIVE

- Amazon EMR allows for launching dynamic instances (with auto scaling) to address varying workloads as opposed to the classic expensive capacity planning before deploying a Hadoop environment.

HIGH AVAILABILITY

- Hadoop on Amazon EMR could be launched in many availability zones and therefore ensures high availability.



APACHE HADOOP ON AMAZON EMR: USE CASES



CLICKSTREAMS DATA ANALYTICS

- Hadoop is used for clickstreams data analytics to track customers behaviour (targeted ad campaign).

LOG DATA ANALYSIS

- Hadoop is used for log data analysis by converting petabytes of un-structured data into key metrics/insights.

MASSIVE DATA PROCESSING

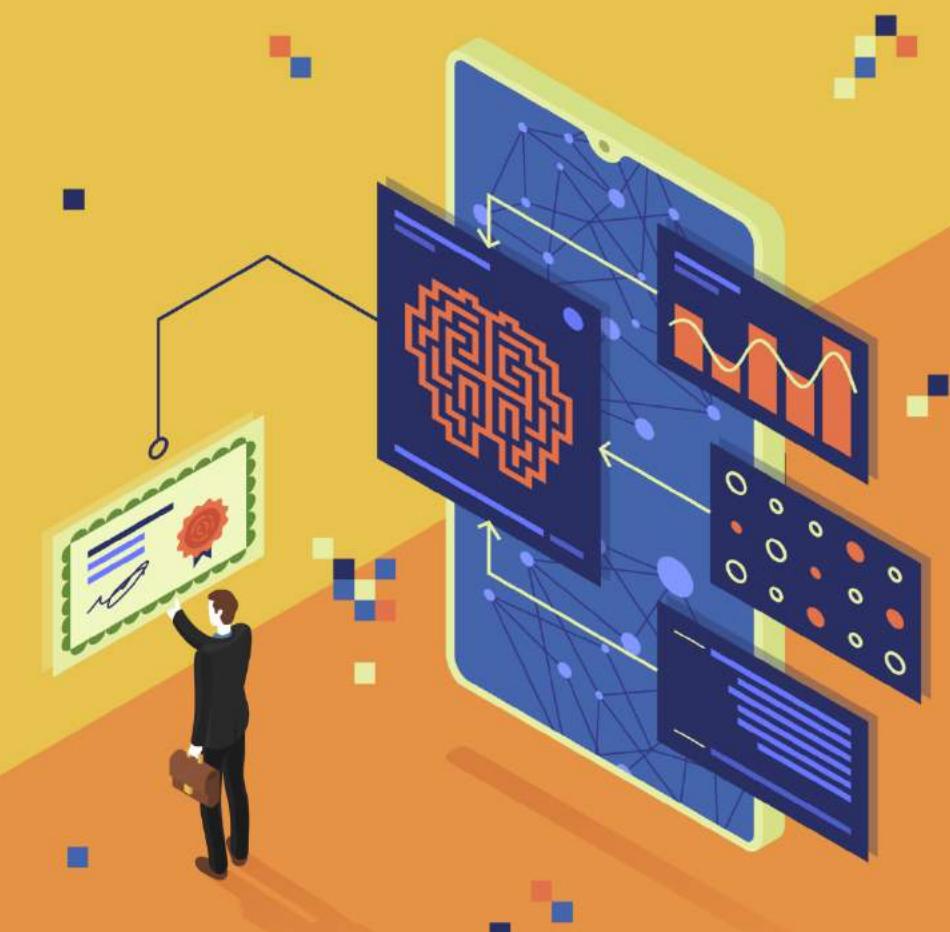
- Hive application can allow for massive scale data analytics by leveraging MapReduce using a SQL interface.

ETL JOBS

- Hadoop works great for ETL jobs such as sorting, joining, and aggregating big data.



APACHE SPARK ON AMAZON EMR



APACHE SPARK



- Apache Spark is an open-source, widely adopted big data processing system.
- It performs batch processing, optimization and in-memory caching for extreme performance and minimum latency.
- Spark is supported in Amazon EMR
- You can launch Apache Spark Clusters in minutes from the console besides leveraging the following features:
 - S3 storage and connectivity using Amazon EMR File System (EMRFS)
 - EC2 spot instances
 - AWS Glue to store Spark SQL table metadata
 - Auto scaling to account for dynamic demand
- Spark encryption and authentication is configured with Kerberos using an Amazon EMR security configuration.
- Amazon EMR installs and manages Apache Spark on Hadoop YARN.



APACHE SPARK: FEATURES



SUPER FAST

- Apache Spark can perform extremely fast data transformation.
- Apache Spark achieves fast processing by removing I/O cost. This is achieved by storing inputs and outputs in-memory as resilient distributed datasets (RDDs).

EASY TO USE

- Apache Spark works with many languages such as Java, Scala, and Python.
- SQL or HiveQL queries could be sent to Apache Spark via Spark SQL module.
- Zeppelin could be used to develop interactive notebooks to visualize your data (similar to jupyter notebooks).

DIVERSE WORKFLOWS

- Several libraries are included such as machine learning (MLlib), stream processing (Spark Streaming), and graph processing (GraphX).
- Apache MXNet could be used as well for Deep Learning applications.

EASY EMR INTEGRATION

- Submit Apache Spark jobs with the Amazon EMR Step API, use Apache Spark with EMRFS to directly access data in Amazon S3, save costs using Amazon EC2 Spot capacity.



APACHE SPARK



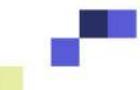
- Apache Spark includes several libraries such as machine learning (MLlib), stream processing (Spark Streaming), and graph processing (GraphX).

**Machine
learning
(MLlib)**

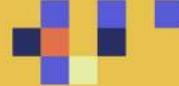
**Spark Stream
processing**

Spark SQL

**Graph
processing
(GraphX)**



APACHE ZEPPLIN



- Apache Zeppelin is a web-based notebook that allows for interactive data analytics and collaborative documents with SQL, Scala.
- Zeppelin is a Multi-purpose Notebook that could be used for:
 - Discovery
 - Ingestion
 - Analytics
 - Visualization
- Apache Zeppelin could be used for data exploration by creating interactive notebooks using Apache Spark.
- Zeppelin could be used as well with deep learning frameworks like Apache MXNet with Spark applications.



APACHE SPARK: USE CASES



DATA STREAMING

- Apache spark could be integrated with Amazon Kinesis and Apache Kafka to stream and analyze data in Realtime.
- Results could be stored to Amazon S3 or on-cluster HDFS.

MACHINE LEARNING

- MLlib library is available with Apache Spark on Amazon EMR to train Machine learning models

SQL

- Spark SQL could send queries with SQL or HiveQL with very low-latency.
- Apache Spark on Amazon EMR can leverage EMRFS to access data on Amazon S3.
- Zeppelin notebooks.
- BI tools via ODBC and JDBC connections.



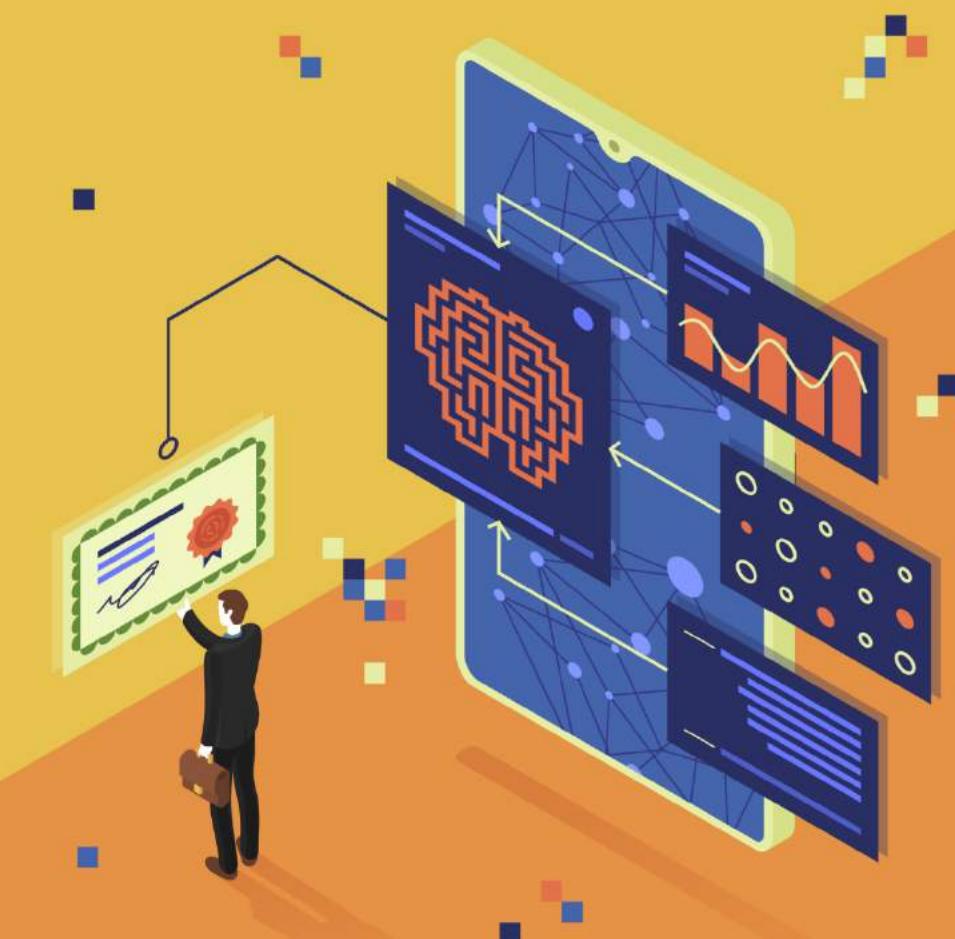
AWS MACHINE LEARNING CERTIFICATION



DOMAIN #2: EXPLORATORY DATA ANALYSIS (24% EXAM)



INTRODUCTION



AWS ML CERTIFICATION EXAM DOMAINS

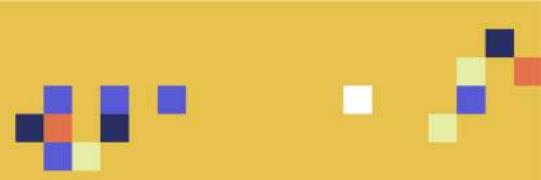


Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #2 OVERVIEW: WHERE ARE WE NOW?!



SECTION #5: JUPYTER NOTEBOOKS, SCIKIT LEARN, PYTHON PACKAGES, AND DISTRIBUTIONS

- Introduction
- Jupyter Notebooks and Scikit Learn
- Python Packages (Pandas, Numpy, Matplotlib and Seaborn)
- Distributions (Normal, Standard, Poisson, Bernoulli)
- Time Series

SECTION #6: AMAZON ATHENA, QUICKSIGHT AND ELASTIC MAP REDUCE

- Amazon Athena Features
- Amazon Athena Deep Dive (Security, Cost, and glue integration)
- Amazon QuickSight Features
- Amazon QuickSight (integration with AWS services)
- Amazon QuickSight ML insights and Use Cases
- Elastic Map Reduce (EMR)
- Apache Hadoop with EMR
- Apache Spark with EMR

DOMAIN #1 OVERVIEW:



SECTION #7: FEATURE ENGINEERING

- Introduction to Feature Engineering
- Amazon SageMaker GroundTruth
- Feature Selection
- Scaling
- Imputation
- Outliers
- One Hot Encoding
- Binning
- Log Transformation
- Shuffling, Feature Splitting, Unbalanced Datasets
- Text Feature Engineering overview
- Bag of words, punctuation, and dates (easy ones!)
- Term Frequency Inverse Document Frequency (TF-IDF)
- N-Grams (Unigram vs. Bigram vs. Trigram)
- Orthogonal Sparse Bigram (OSB)
- Cartesian Product Transformation

WE ARE HERE!



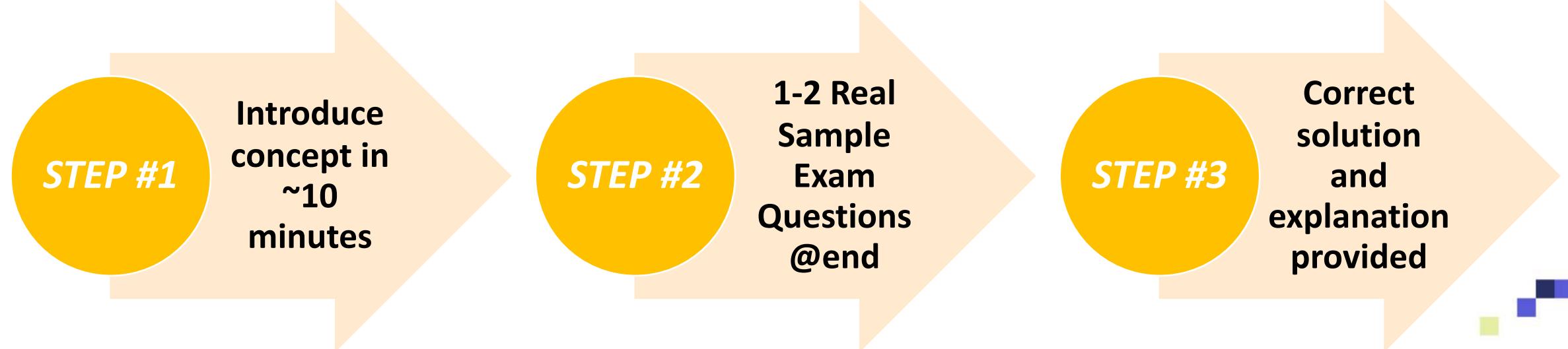
LECTURE DESIGN



- We know how hard it is to study for an exam especially if you have a busy schedule.
- This course is designed to be extremely on point and optimized to pass the exam.

No boring content. Zero unnecessary information.

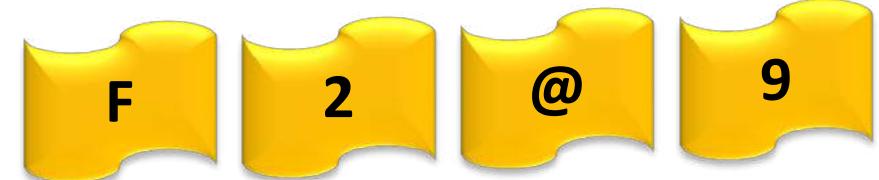
- Here's the lecture structure that we will follow:



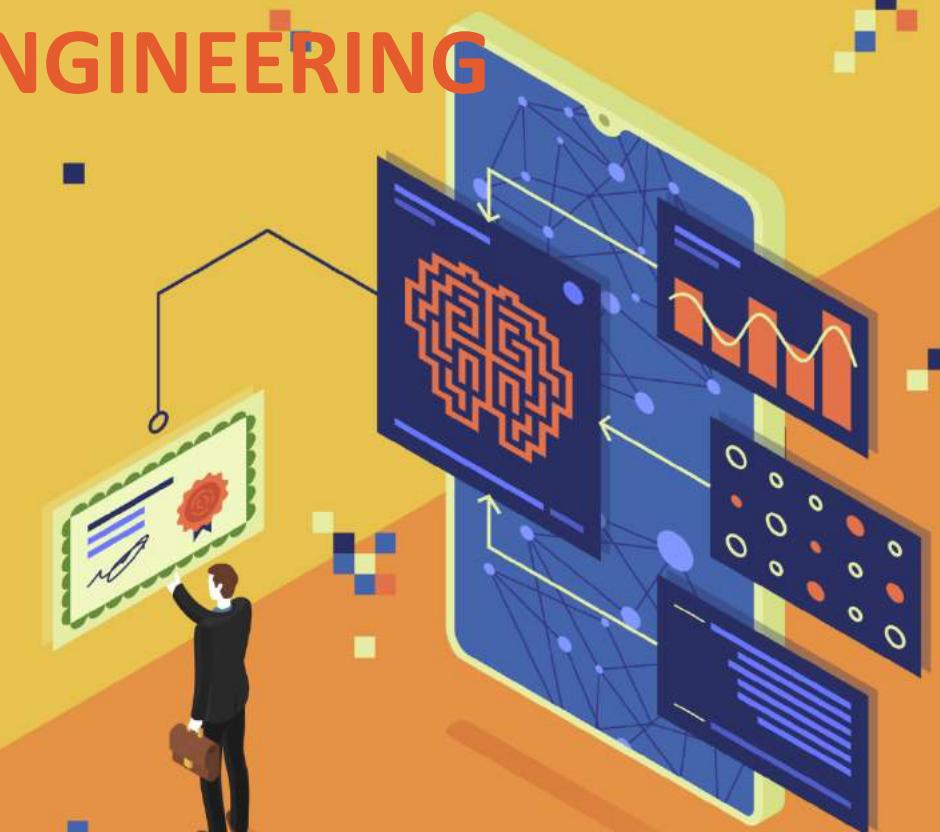
RECALL OUR MINI CHALLENGE AND PRIZE!



- For those of you who will successfully complete the entire section and watch the videos till the end, they will receive a valuable prize!



INTRODUCTION TO FEATURE ENGINEERING



WHAT IS FEATURE ENGINEERING?



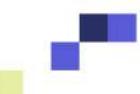
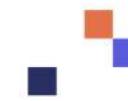
- Machine Learning Algorithms require training data to train.
- Feature engineering is a critical task that data scientists have to perform prior to training the AI/ML models.
- As a data scientist, you may need to:
 - Highlight important information in the data
 - Remove/isolate unnecessary information (e.x.: outliers).
 - Add your own expertise and domain knowledge to alter the data.
- Feature engineering is an art of introducing new features that weren't existing before.
- Data scientists spend 80% of their time performing feature engineering.
- The remaining 20% is the easy part which includes training the model and performing hyperparameters optimization.
- Performing proper feature engineering is crucial to improve AI/ML model performance.



FEATURE ENGINEERING: PROPER QUESTIONS TO ASK?



- As a data scientist, you need to answer the following questions:
 - *What are the capabilities of the ML model I have?*
 - *Which features should I select?*
 - *Can I add my domain knowledge to use less features?*
 - *Can I come up with new features from the data I have at hand?*
 - *What should I put in the missing data locations?*
- It is important to choose features that are most relevant to the problem.
- Adding new features that are unnecessary will increase the computational requirements needed to train the model (curse of dimensionality).
- There are numerous techniques that could be used to reduce the number of features (compress/encode the data) such as Principal Component Analysis (PCA) – This will be covered in the next sections.



FEATURE ENGINEERING: EXAMPLE



- Let's take a look at this data and see what's wrong with it!

CUSTOMER ID	CUSTOMER NAME	LOCATION	CLICK ON AD?
1	Steve	USA	Yes
2	Mitch	Canada	1
3	Chanel	France	0
4	Bird		1
5	Cynthia	Netherlands	0
6	Chanel	France	0

ENTIRE COLUMN REQUIRES ENCODING

MISSING INFORMATION

REQUIRES FORMATTING

DUPLICATE ENTRY

Red annotations highlight several issues with the data:

- A red arrow points from the text "ENTIRE COLUMN REQUIRES ENCODING" to the "LOCATION" column header.
- A red circle highlights the entire "CLICK ON AD?" column, with the text "MISSING INFORMATION" pointing to the empty cell for customer 4.
- A red circle highlights the "CLICK ON AD?" column, with the text "REQUIRES FORMATTING" pointing to the "Yes" entry for customer 1.
- A red circle highlights the entire "LOCATION" column, with the text "DUPLICATE ENTRY" pointing to the duplicate entries for customers 3 and 6.

FEATURE ENGINEERING TECHNIQUES



Imputation

Handling
Outliers

Binning

Log
Transform

One-Hot
Encoding

Feature
Split

Scaling



FEATURE ENGINEERING: TOOLS



SAGEMAKER AND
JUPYTER
NOTEBOOKS
(ADHOC)

GLUE ETL JOB
(REPEATABLE OR
REUSABLE
APPLICATIONS)



JUPYTER
NOTEBOOKS

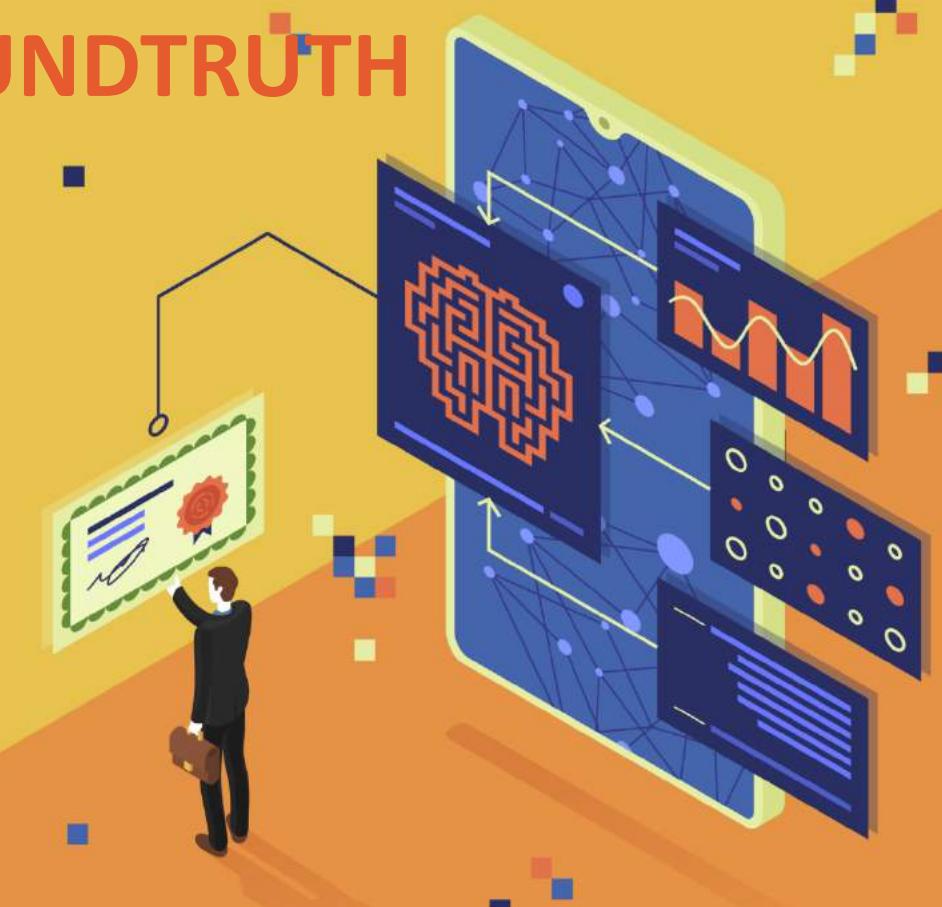


AMAZON
SAGEMAKER



AWS GLUE

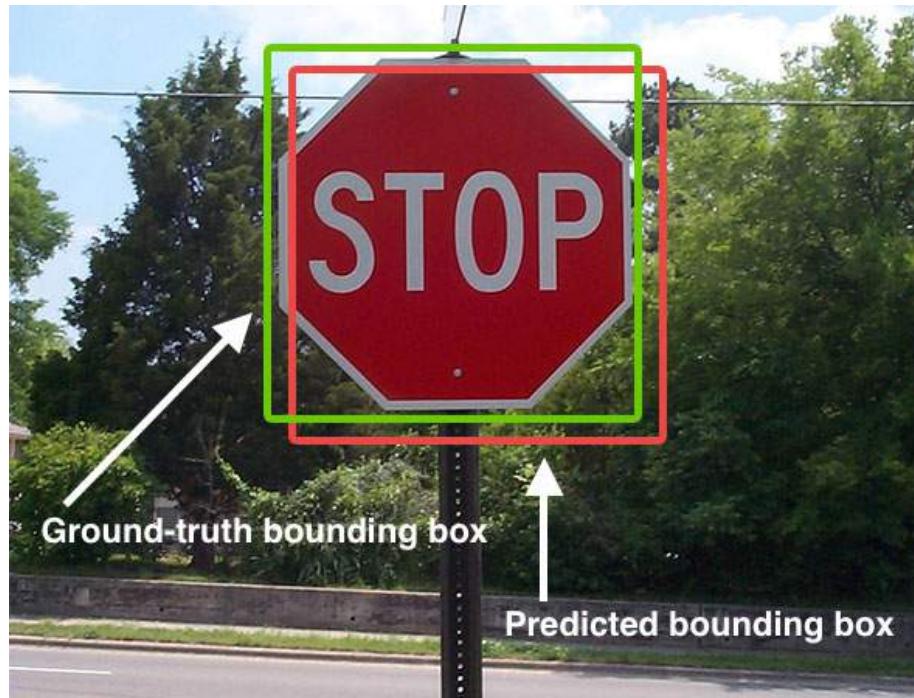
AMAZON SAGEMAKER GROUNDTRUTH



SAGEMAKER GROUND TRUTH



- In order to perform supervised training, labelled input/output data is required.
- Data labeling is a very expensive and time consuming job.
- Amazon SageMaker Ground Truth allows for creating datasets quickly and at low cost.
- Amazon Sagemaker provide developers and data scientists access to thousands of labelers around the globe.
- SageMaker ground truth can reduce the cost by 70% using automatic labeling.
- Automatic labeling works by training ground truth using the data that has already been labelled manually.

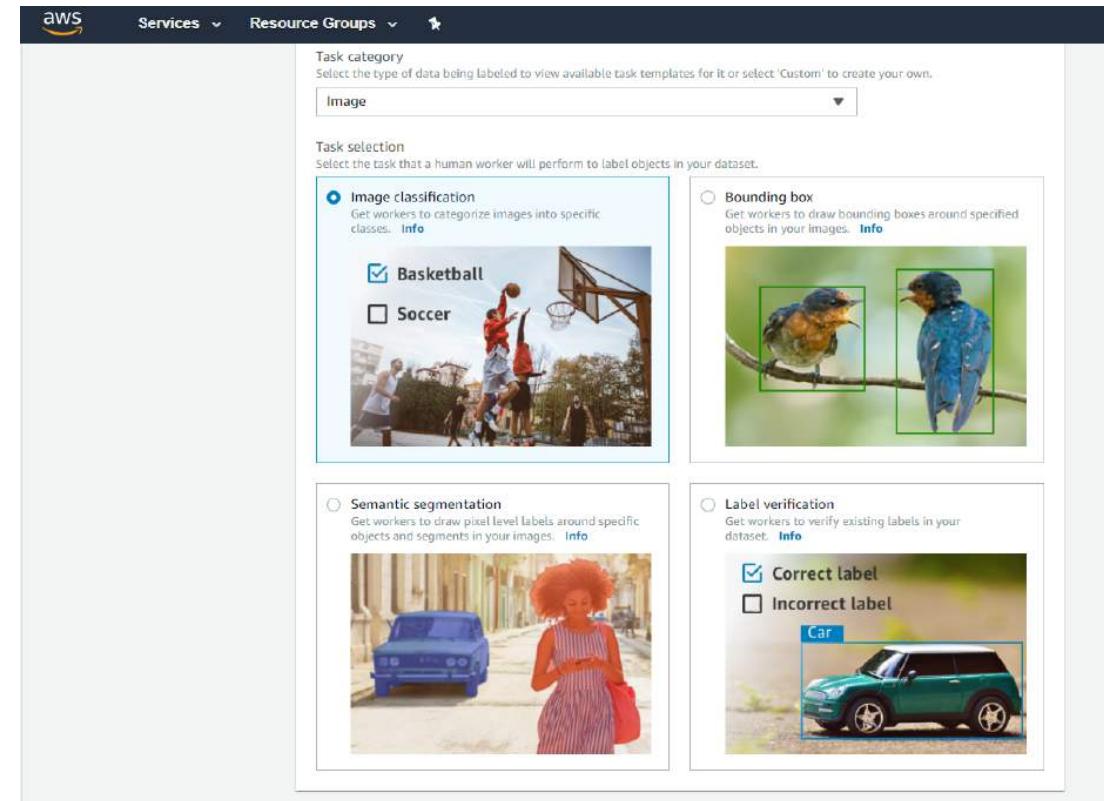


https://commons.wikimedia.org/wiki/File:Intersection_over_Union_-_object_detection_bounding_boxes.jpg

SAGEMAKER GROUND TRUTH



- Amazon SageMaker can save time and cost by using auto labeling service. The model learns from experience and continuously learn from human-labelled data.
- If model has high confidence in the label, it will automatically label raw data (produce final product).
- If the model has low confidence in the label, it will pass the image to human labellers for review.
- This dataset are then fed back to the model to improve and learn from experience (so it becomes better next time and learn from its mistakes which will lead to higher confidence next time).



SAGEMAKER GROUND TRUTH: BENEFITS



MASSIVE COST SAVINGS

- SageMaker Ground Truth can result in 70% savings by automating the labeling process.
- This is achieved by using machine learning to automatically label images.
- Human labelers are needed only if the confidence is not high.

ACCESS TO THOUSANDS OF LABELERS WORLDWIDE

- SageMaker Ground Truth allows you to:
 - (1) Choose your own private labelers.
 - (2) Outsource to labelers outside of your company (500,000 labelers with Amazon Mechanical Turk).
 - (3) For sensitive information, Amazon provides professional labeling service with companies verified by Amazon.

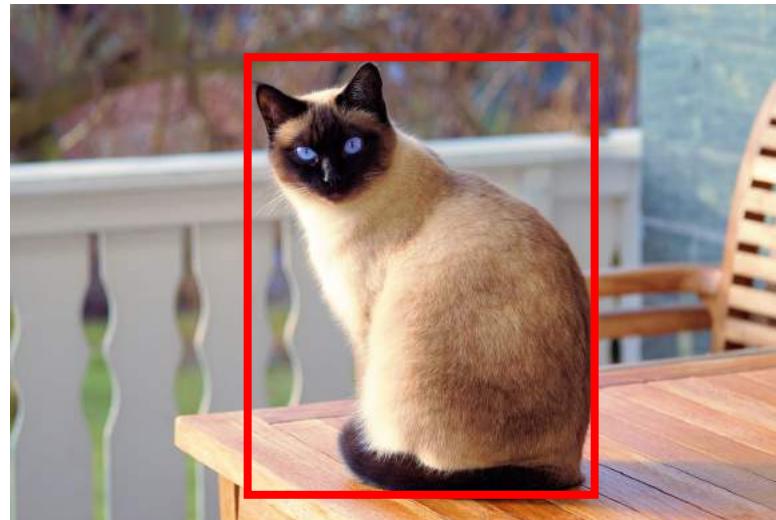
FAST

- SageMaker Ground Truth is fast and efficient.
- Labeling services provided manually are being automatically assessed to ensure quality.

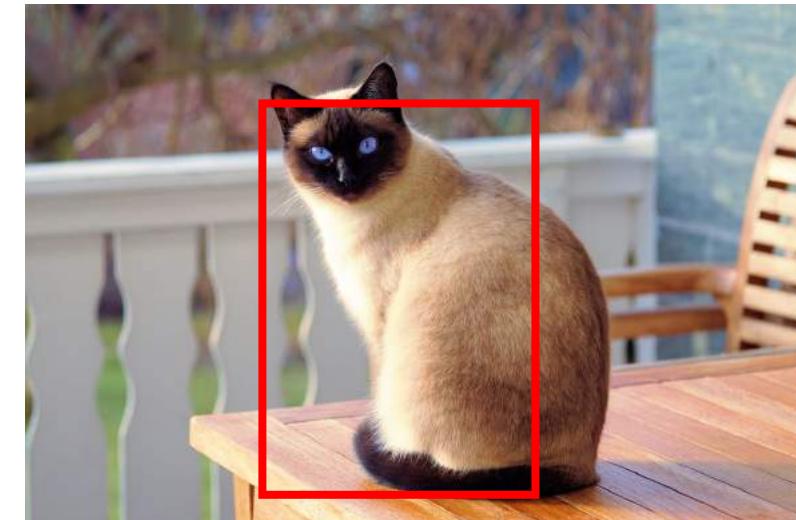
SAGEMAKER GROUND TRUTH: LABELING GUIDELINES



- SageMaker Ground Truth allows for data labeling of many kinds such as: images, audio, and text.
- As a user, you can provide guidelines to the human labeler as shown below.



GOOD LABEL



BAD LABEL

<https://www.pexels.com/photo/cat-outdoors-326875/>

SAGEMAKER GROUND TRUTH: PRICING

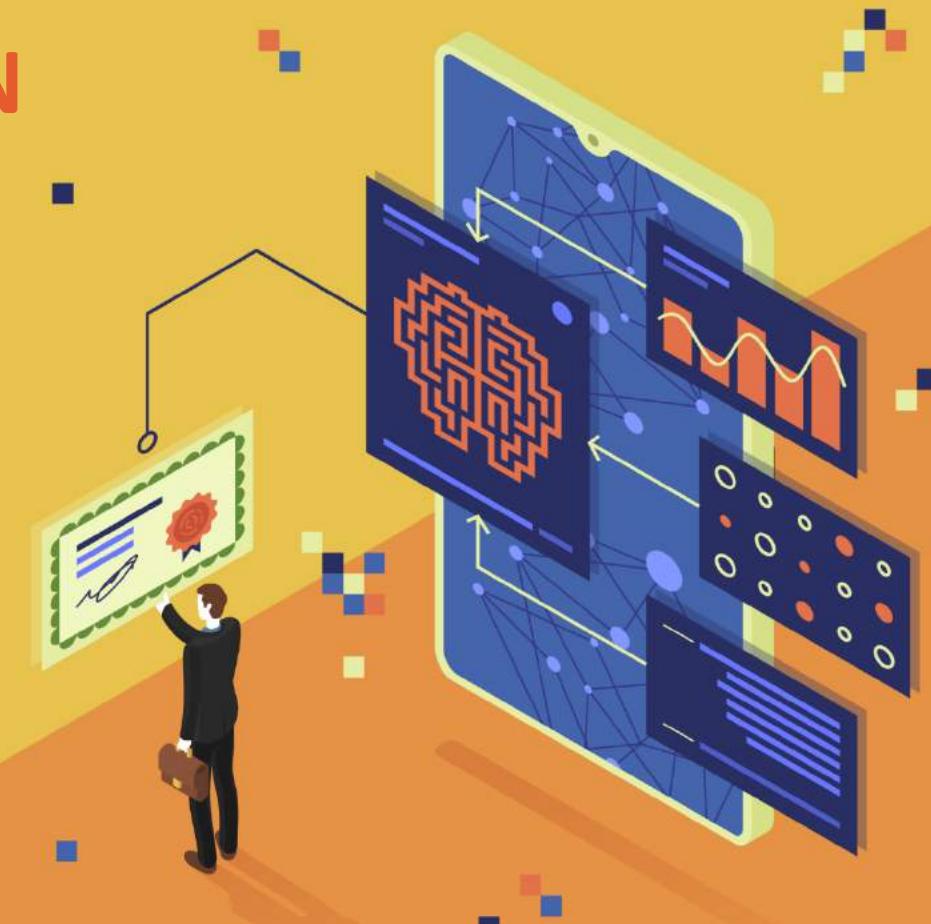


- The pricing model is per object.
- It does not matter if the object is labeled automatically or manually.
- Additional cost is paid with Amazon Mechanical Turk.
- The cost differs based on the workflow: (1) classification, (2) bounding box, (3) semantic segmentation.

NUMBER OF LABELED OBJECTS	PRICE PER LABELED OBJECT
Less than 50,000 objects	\$0.08
50,000 to 1,000,000 objects	\$0.04
Greater than 1,000,000 objects	\$0.02

WORKFLOW	SUGGESTED PRICE PER LABELER
Image classification	\$0.012
Text classification	\$0.012
Bounding box	\$0.036
Semantic Segmentation	\$0.84

FEATURE SELECTION

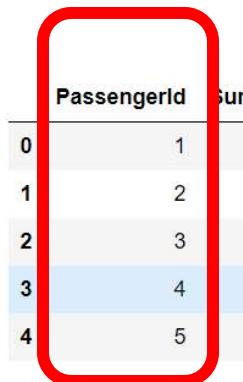


FEATURE SELECTION: GET RID OF USELESS DATA



- Feature selection is the process of selecting relevant features only from the available dataset.
- Feature selection enhances model performance by removing the “noise” and therefore enabling the model to focus on important features only.
- Useless data is a type of data that is discrete and has no relationship whatsoever with the output.
- We usually drop the useless features from the dataset before training the model
- Example include: random customer IDs at a store or a bank

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	Cumings, Mrs. John Bradley (Florence Briggs Th... Heikkinen, Miss. Laina	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	Allen, Mr. William Henry	male	35.0	1	0	113803	53.1000	C123	S
4	5	0				0	0	373450	8.0500	NaN	S



USELESS
INFORMATION



Photo Credit: <https://www.maxpixels.net/Not-Useless-Town-Sign-Usefulness-Use-Useful-Road-822236>

FEATURE SELECTION: PRINCIPAL COMPONENT ANALYSIS

- PCA is an unsupervised machine learning algorithm that performs dimensionality reductions while attempting at keeping the original information unchanged.
- PCA works by trying to find a new set of features called components.
- Components are composites of the uncorrelated given input features.
- The first component accounts for largest data variability followed by second component and so on.

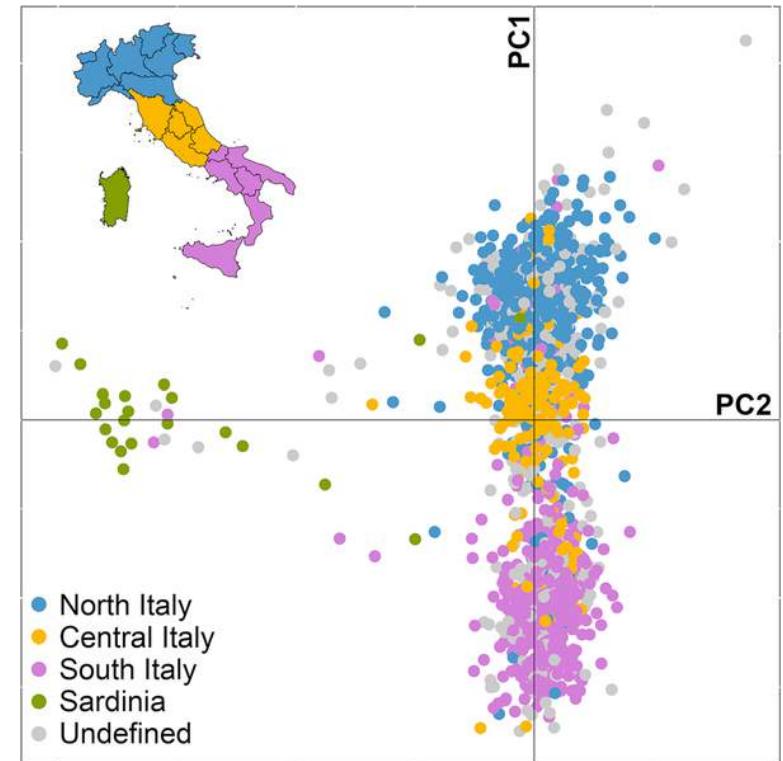


Photo Credit: https://commons.wikimedia.org/wiki/File:Principal_Component_Analysis_of_the_Italian_population.png

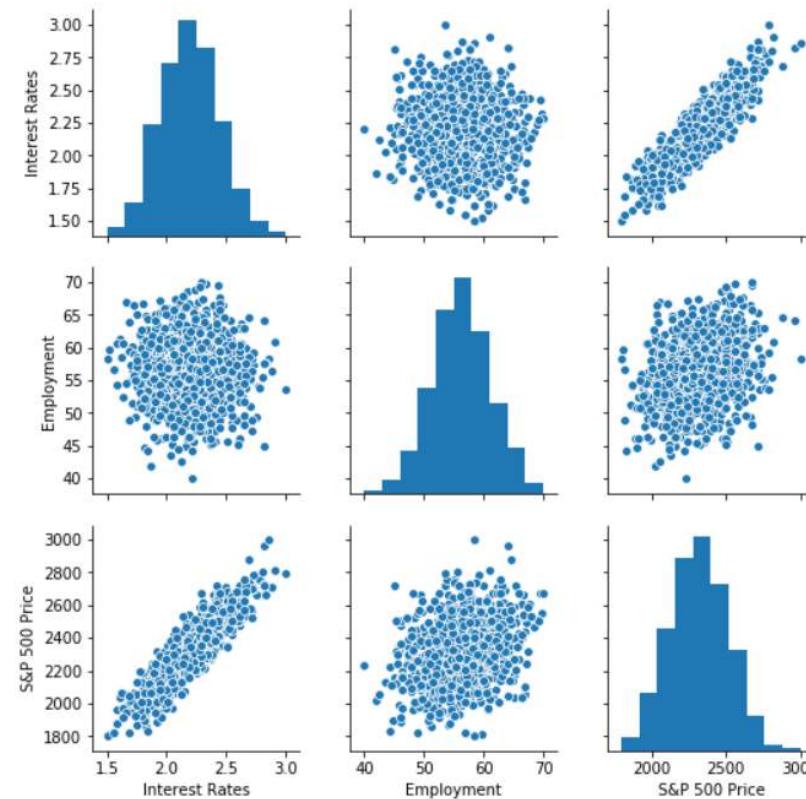
SCALING



SCALING



- Feature Scaling (normalization) is required prior to training of Artificial Neural Networks to ensure that features are within the same scale.
- Example: interest rate and employment rates are at a different scale. This will result in one feature dominating the other feature.
- Scikit Learn offers several tools to perform feature scaling.



	Interest Rates	Employment	S&P 500 Price
0	1.943859	55.413571	2206.680582
1	2.258229	59.546305	2486.474488
2	2.215863	57.414687	2405.868337
3	1.977960	49.908353	2140.434475
4	2.437723	52.035492	2411.275663
5	2.143637	56.060598	2187.344909
6	2.148647	51.513208	2263.049249
7	2.176184	53.475909	2281.496374
8	2.125352	63.668422	2355.163011
9	2.225682	56.993396	2326.330337
10	1.814688	55.361780	2078.553895
11	2.281897	58.484752	2337.504507
12	2.426738	55.709328	2485.774097

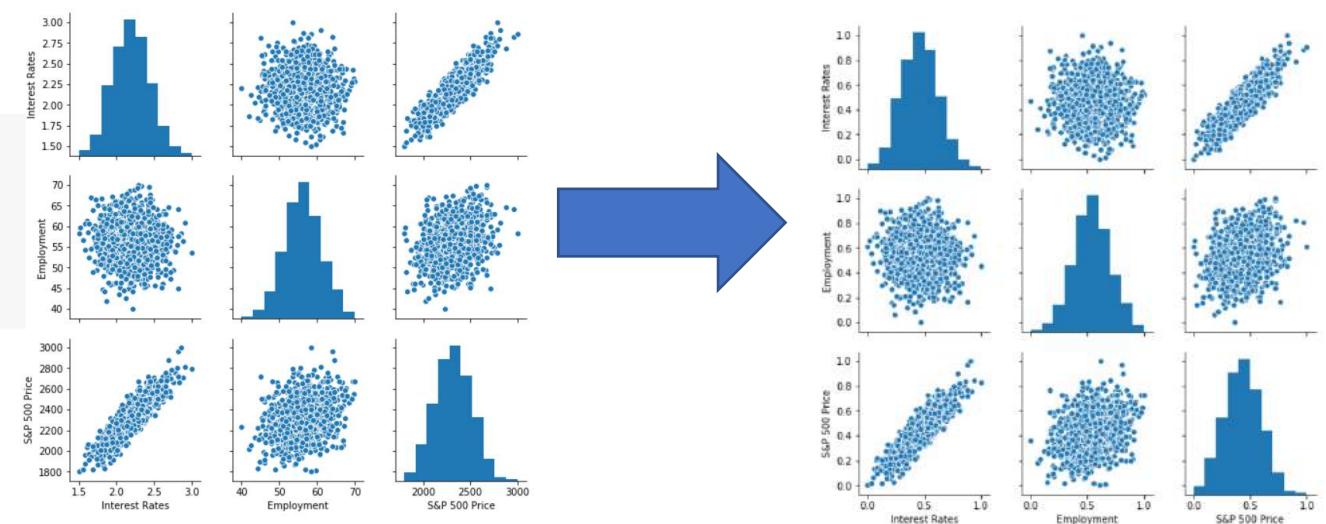
SCALING: NORMALIZATION



- Normalization is conducted to make feature values range from 0 to 1.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

```
from sklearn.preprocessing import MinMaxScaler  
scaler = MinMaxScaler()  
y = scaler.fit_transform(y)
```



SCALING: STANDARDIZATION



- Standardization is conducted to transform the data to have a mean of zero and standard deviation of 1.
- Standardization is also known as Z-score normalization.
- Standardization is preferred over normalization when there are a lot of outliers.
- You can revert back to the original data range by applying inverse standardization.

$$z = \frac{x - \bar{x}}{\sigma}$$

- *Note: You can get rid of outliers all together by using Random Cut Forest Technique!*

SCALING: NORMALIZATION Vs. STANDARDIZATION EXAMPLE



ORIGINAL DATASET

	Interest Rates	Employment	S&P 500 Price
0	1.943859	55.413571	2206.680582
1	2.258229	59.546305	2486.474488
2	2.215863	57.414687	2405.868337
3	1.977960	49.908353	2140.434475
4	2.437723	52.035492	2411.275663
5	2.143637	56.060598	2187.344909
6	2.148647	51.513208	2263.049249
7	2.176184	53.475909	2281.496374
8	2.125352	63.668422	2355.163011
9	2.225682	56.993396	2326.330337
10	1.814688	55.361780	2078.553895
11	2.281897	58.484752	2337.504507
12	2.426738	55.709328	2485.774097

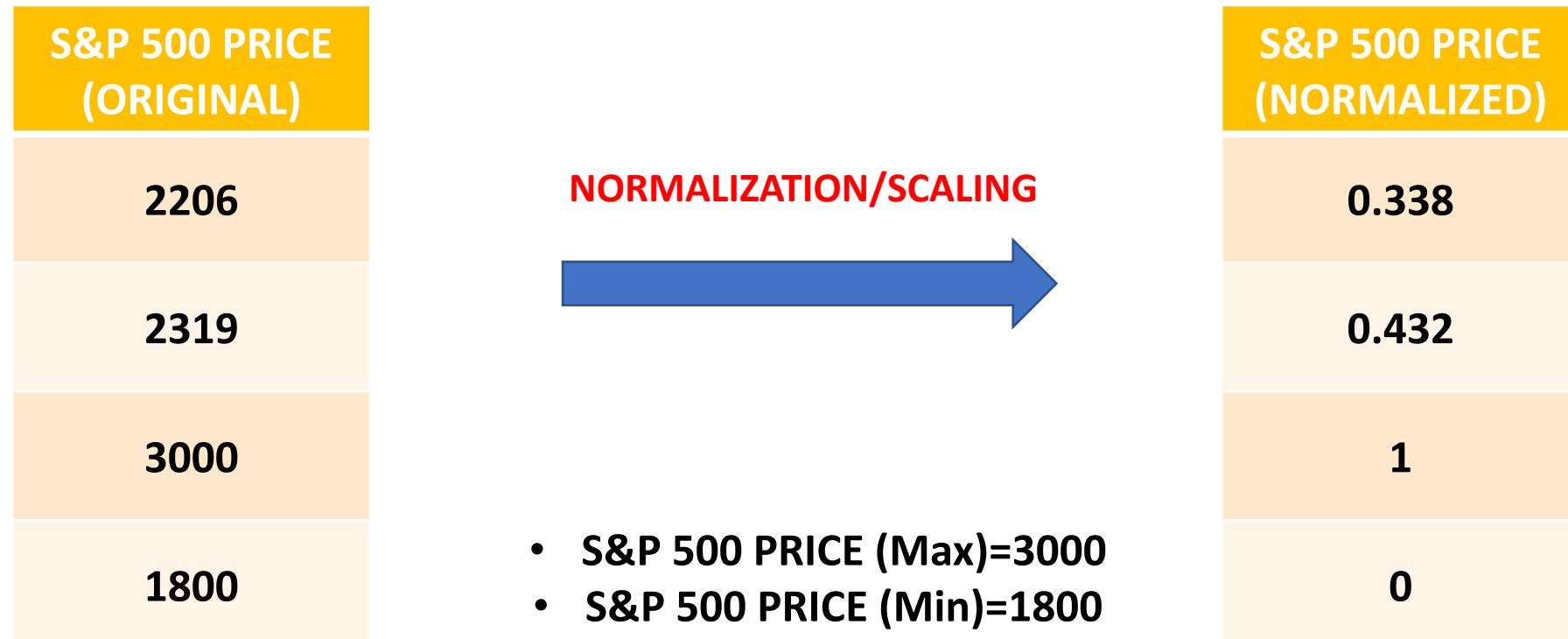
QUICK STATS!

	Interest Rates	Employment	S&P 500 Price
count	1000.000000	1000.000000	1000.000000
mean	2.195392	56.254855	2319.999936
std	0.241630	4.862178	193.854745
min	1.500000	40.000000	1800.000000
25%	2.035735	53.029784	2190.447901
50%	2.198214	56.160941	2312.443024
75%	2.359061	59.422633	2455.764327
max	3.000000	70.000000	3000.000000

SCALING: NORMALIZATION



$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} = \frac{2206 - 1800}{3000 - 1800} = 0.338$$



SCALING: STANDARDIZATION



$$z = \frac{x - \bar{x}}{\sigma} = \frac{2206 - 2319}{193.8} = -0.583$$

S&P 500 PRICE (ORIGINAL)
2206
2319
3000
1800

STANDARDIZATION



S&P 500 PRICE (STANDARDIZED)
-0.583
0
3.513
-2.67

ORIGINAL
DATA = MEAN

- S&P 500 PRICE (Mean)=2319
- S&P 500 PRICE (Std)=193.8

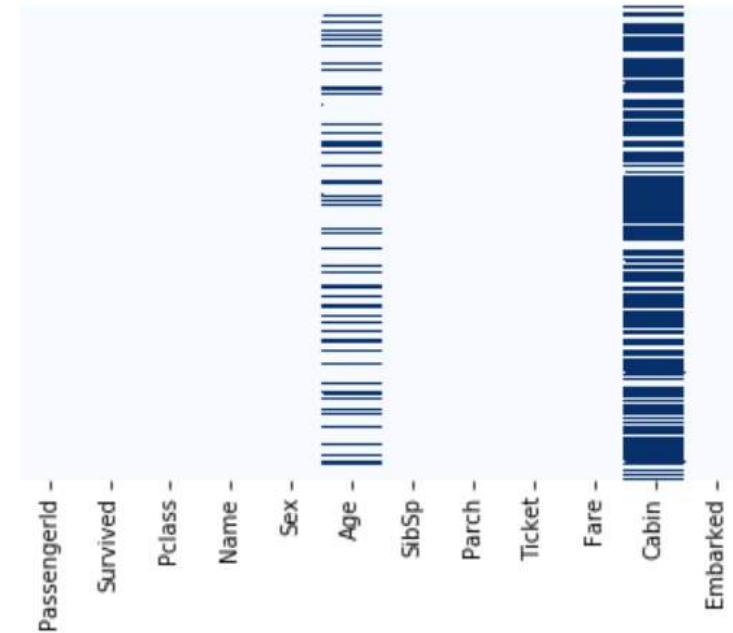
IMPUTATION



1. IMPUTATION: BEYOND REPAIR OR IS THERE HOPE?!



- As a data scientist, you will encounter data with missing values “NaN values”.
- Missing data might arise from a sensor or a server that might have broken down
- In order to feed this data a machine learning model, you will need to:
 - Replace the missing NaN values with something which could be mean, median or maximum value!
 - Remove the entire row (not recommended since row might contain valuable information).
- You can obtain the mean by replacing NaN values with feature mean for example.
- Let's take a look at the Titanic Dataset.



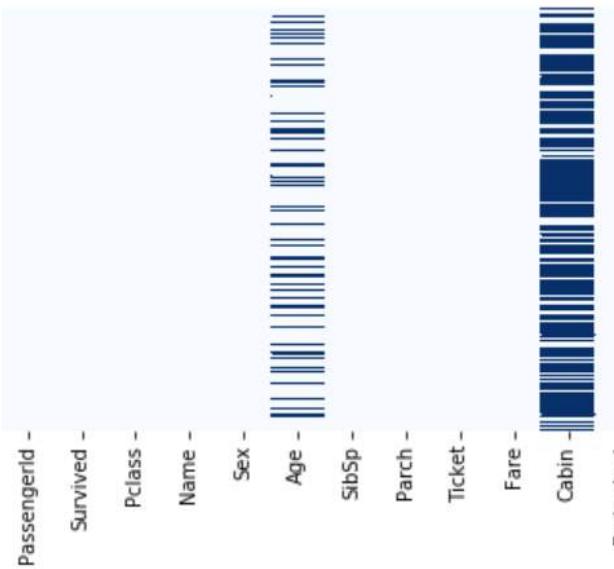
PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	Cumings, Mrs. John Bradley (Florence Briggs Th... Heikkinen, Miss. Laina	female	38.0 26.0	1 0	0	PC 17599 STON/O2. 3101282	71.2833 7.9250	C85 NaN	C S
2	3	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
3	4	0	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

1. IMPUTATION: DROPPING



- You might need to remove the “Cabin” column.
- It seems that there are a ton missing data beyond repair.

```
1 # Let's drop the cabin column  
2 training_set.drop('Cabin',axis=1,inplace=True)
```

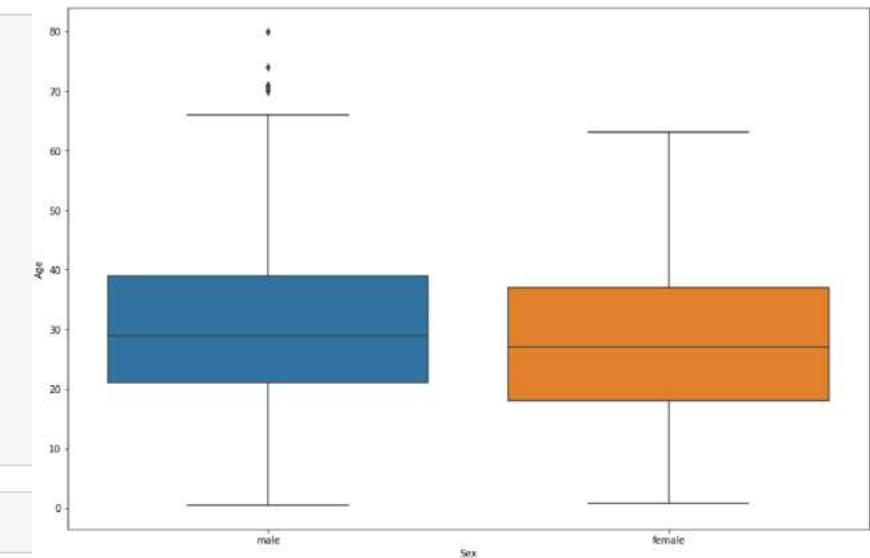


1. IMPUTATION: REPLACING MISSING VALUES WITH MEAN

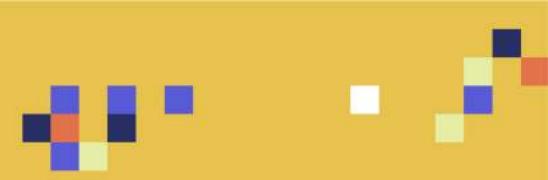
- For the “Age” Column, you might need to obtain the mean and replace missing values with the mean of the “Age” column.
- Another way of improving this, is to replace missing values with the average “age” depending on the “sex” of the passenger.

```
In [22]: 1 def Fill_Age(data):
2     age = data[0]
3     sex = data[1]
4
5     if pd.isnull(age):
6         if sex is 'male':
7             return 29
8         else:
9             return 25
10    else:
11        return age
12
```

```
In [23]: 1 training_set['Age'] = training_set[['Age', 'Sex']].apply(Fill_Age, axis=1)
```



1. IMPUTATION: REPLACING MISSING VALUES WITH MEAN



- Replacing missing values with the mean is simple and effective but lack accuracy.
- In the presence of outliers, using the median might lead to better results.
- What about categorical data?
 - You can replace missing value with the most frequently occurred value.
 - If values follows uniform distribution, use “other” instead.

1, 3, 3, **6**, 7, 8, 9

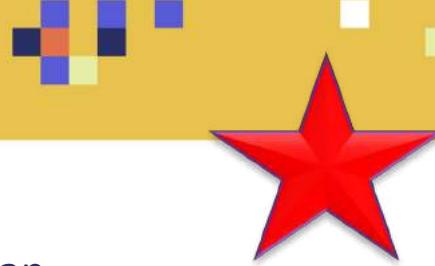
Median = **6**

1, 2, 3, **4**, **5**, 6, 8, 9

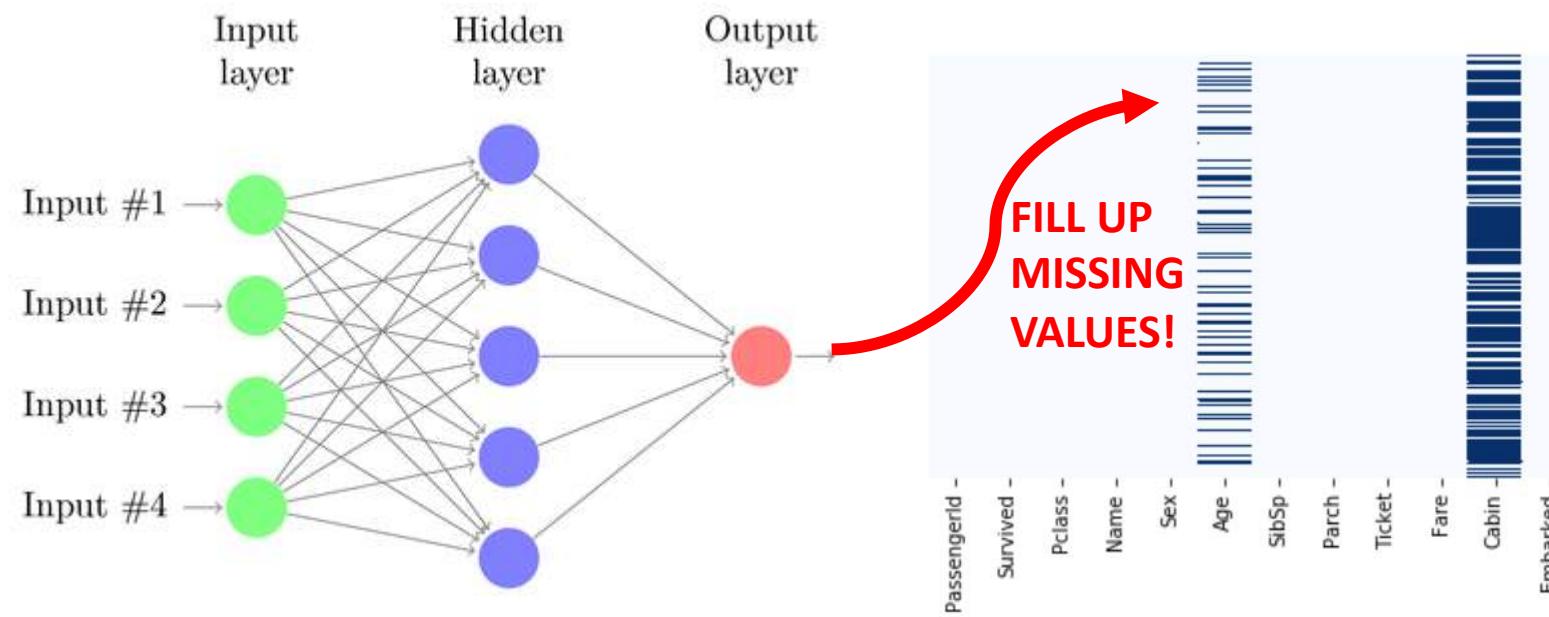
$$\begin{aligned}\text{Median} &= (4 + 5) \div 2 \\ &= \underline{\underline{4.5}}\end{aligned}$$



1. IMPUTATION: REPLACING MISSING VALUES WITH MACHINE LEARNING!



- Another way of handling missing values is by using machine learning techniques.
- These techniques will lead to most accuracy but they are much more complex than simple “mean replacement”.
- They could work well with numerical and categorical data.
- Deep Learning, KNN or Regression techniques could be used.
- Remember that you can always collect more data.



1. IMPUTATION: HANDLING MISSING VALUES IN AWS DEEPAR ALGORITHM



- AWS DEEPAR which is a time series forecasting algorithm provided by SageMaker.
- In timeseries, having missing data might have a huge impact on the forecasting accuracy.
- Imputing missing values with zeros, will bias the algorithm towards zero which is not desirable.
- Now DeepAR algorithm can work with missing data (directly in the model) so you do not have to perform any manual feature extraction.
- DeepAR relies on Recurrent Neural Network (deep learning) to perform the imputation resulting in a much accurate results.

RECURRENT NEURAL NETWORK

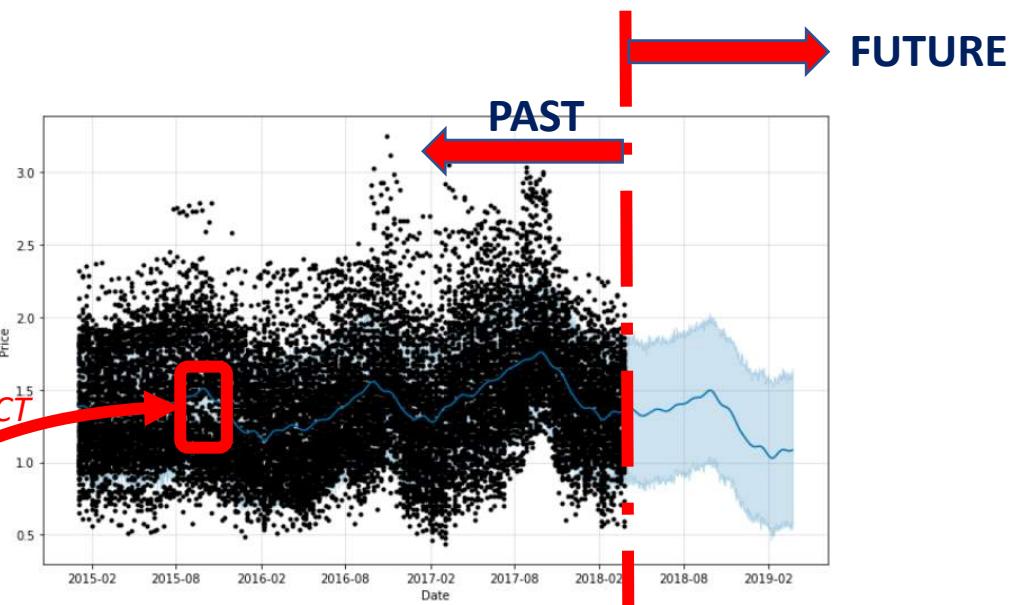
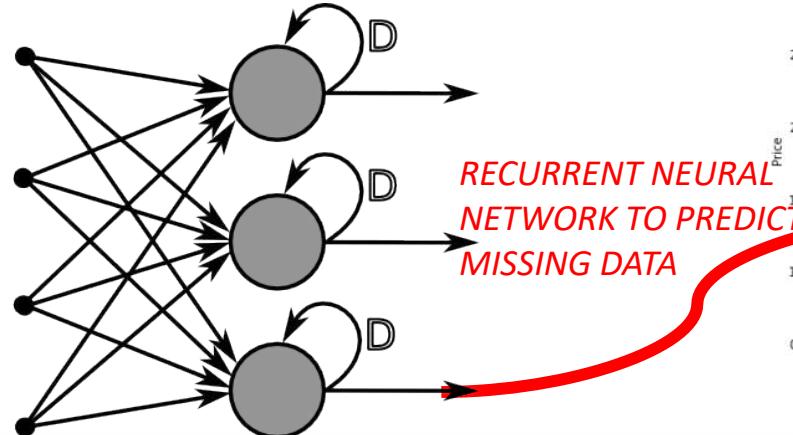
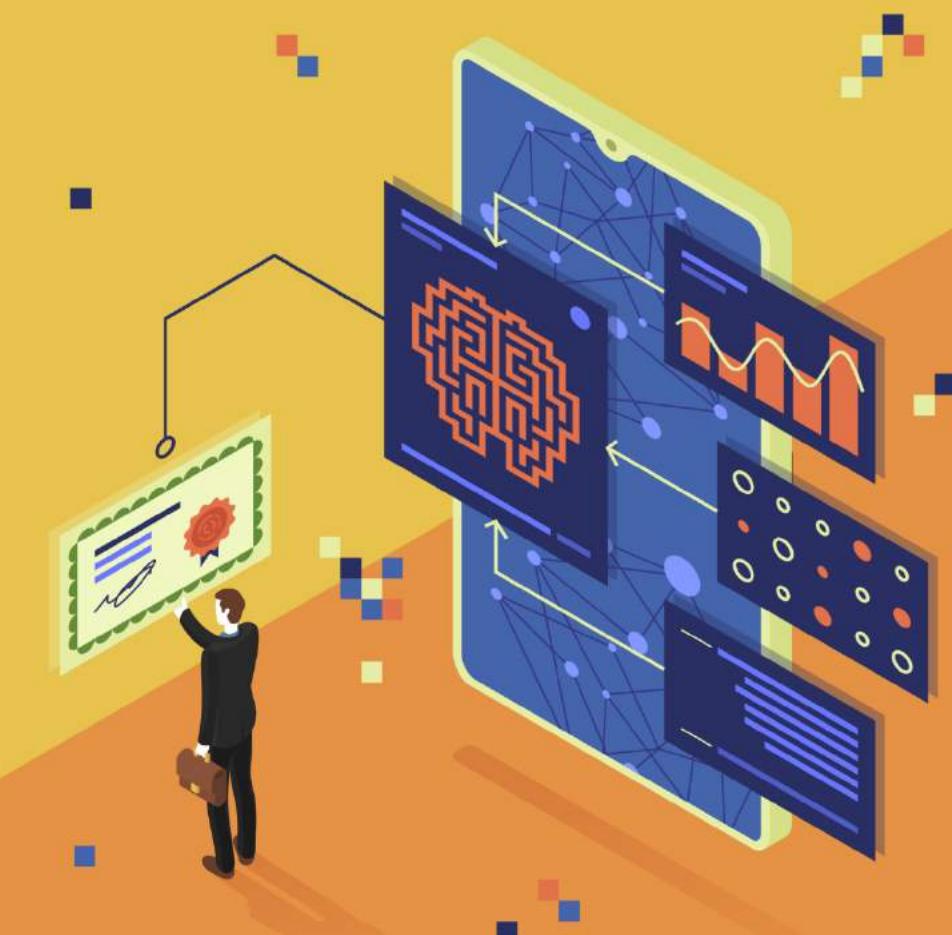


Photo Credit: https://commons.wikimedia.org/wiki/File:RecurrentLayerNeuralNetwork_english.png

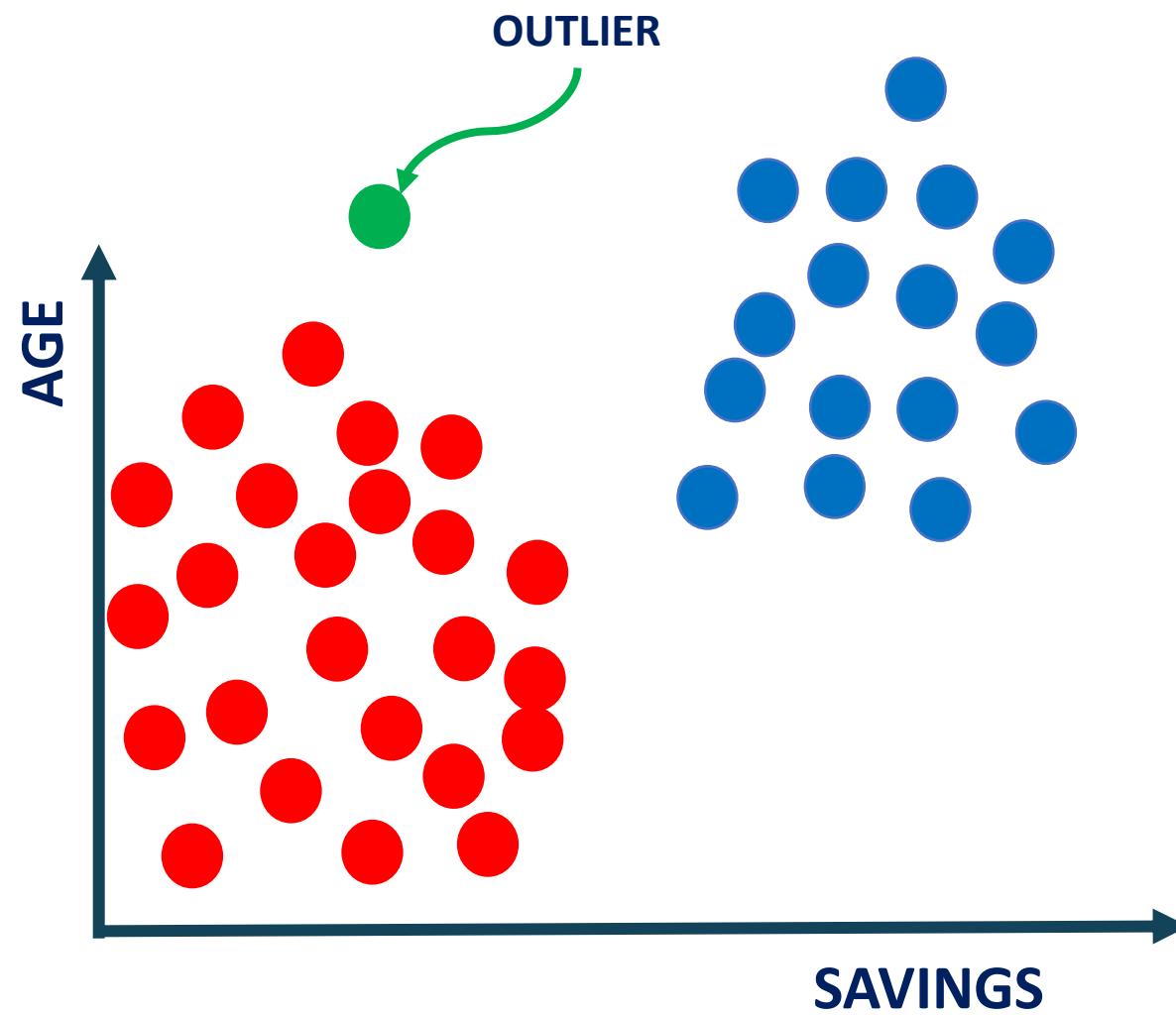
OUTLIERS



HOW TO HANDLE DATA OUTLIERS?



- The best way to handle outliers is by performing data visualization.
- The following 2 statistical methods could be used to detect outliers:
 - Standard deviation
 - Percentiles



VARIANCE



- The Variance is defined as the average of the squared differences from the Mean.
- Variance calculation steps:
 1. Calculate the mean (average) of all the numbers
 2. For each data point, subtract the Mean (calculated in step #1)
 3. Square the result to obtain the squared difference.
 4. Calculate the average of those squared differences.

$$Variance = \sigma^2 = \frac{1}{N} \left(\sum_{i=1}^N (x_i - \mu)^2 \right)$$

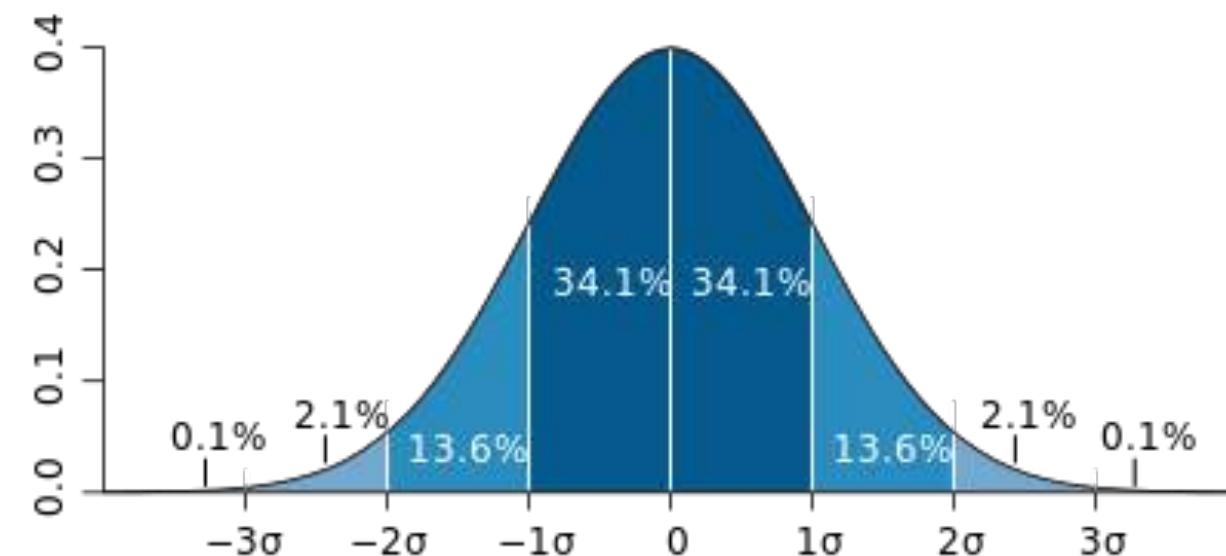
STANDARD DEVIATION



- Standard deviation is a measure of the dispersion from the mean
- Represented by the symbol is σ (sigma)
- Standard deviation is the square root of the variance

$$\text{Standard Deviation } (\sigma) = \sqrt{\text{variance}}$$

- Standard deviation is used to detect outliers.
- Data points that are greater than 2σ or 3σ or 4σ are considered outliers (tunable parameter).
- AWS Random Cut Forest algorithm is used for outlier detection.
- AWS Random Cut Forest is included in many services such as Amazon Kinesis analytics and QuickSight.



PERCENTILES



- Another way of detecting and removing outliers is by using percentiles.
- For example, you can assume that values that are more than 90% percentile will be removed from the data.

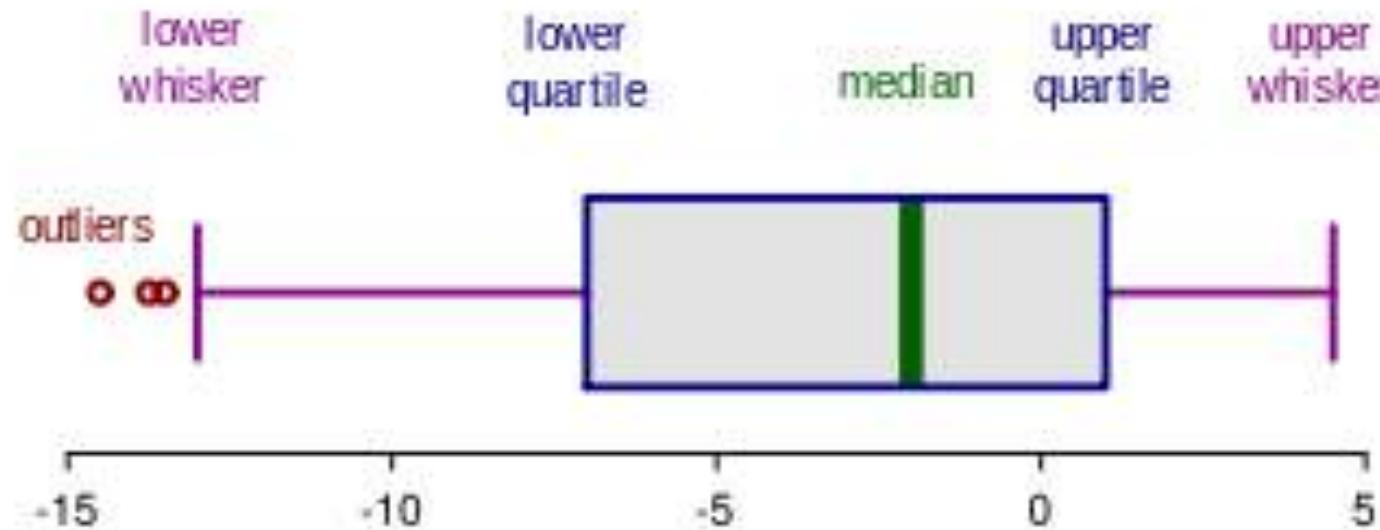


Photo Credit: https://commons.wikimedia.org/wiki/File:Elements_of_a_boxplot_en.svg

HOW TO DEAL WITH OUTLIERS?



- Is Elon musk an outlier?!



[Photo Credit: https://commons.wikimedia.org/wiki/File:Elon_Musk_Royal_Society.jpg](https://commons.wikimedia.org/wiki/File:Elon_Musk_Royal_Society.jpg)

ONE HOT ENCODING



ONE-HOT ENCODING: WHY DO WE NEED IT?



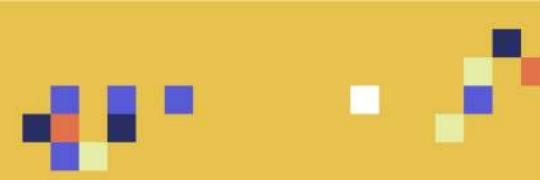
- Can we simply replace colors with integer values?
- The machine learning model will assume that:

GREEN > YELLOW > RED

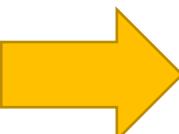
COLOR	ENCODED COLOR
RED	1
RED	1
YELLOW	2
GREEN	3
YELLOW	2

WRONG!

ONE-HOT ENCODING



- One hot encoding is widely used in machine learning.
- It works by converting values such as “color” into columns with 1’s and 0’s in them.
- Since machine learning models deal with numbers, we perform one hot encoding to convert from categorical data into numerical.
- If you have N categories, you will need N-1 binary columns to represent them.



COLOR	RED	YELLOW	GREEN
RED	1	0	0
RED	1	0	0
YELLOW	0	1	0
GREEN	0	0	1
YELLOW	0	1	0

ONE-HOT ENCODING: ORDINAL Vs. NOMINAL



- Recall the difference between nominal and ordinal data.
 - In ordinal data, order is important
 - In Nominal data, order is not important.

NOMINAL

Order of colors doesn't mean anything!

COLOR
RED
RED
YELLOW
GREEN
YELLOW

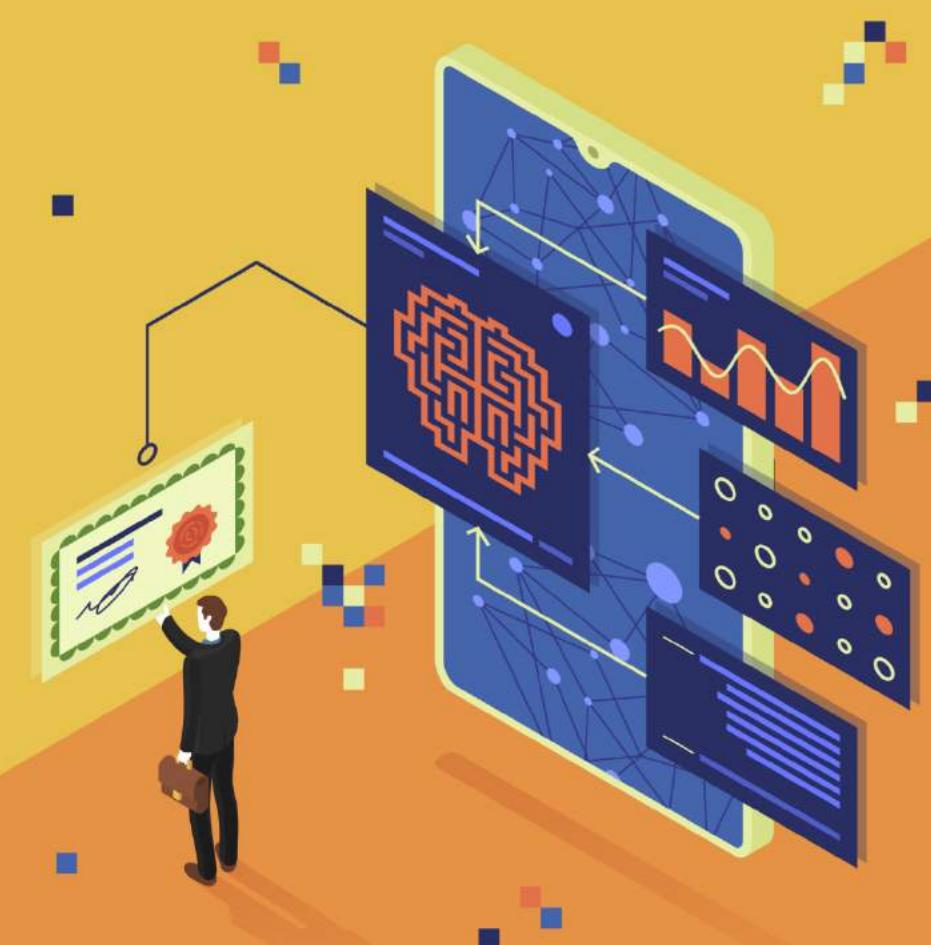
ORDINAL

Order is important!



- 1 star means poor quality course
- 5 star means great quality course

BINNING



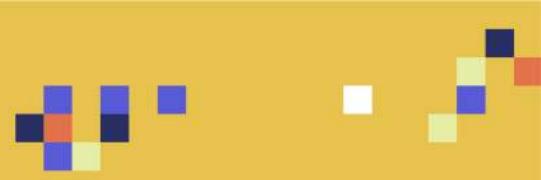
BINNING



- Binning could be used for nonlinear transformation from numeric values to categorical.
- Binning is used when the relationship between numeric feature (input) and output is not linear. (*no monotonic increase or decrease with the target*). Then by doing binning, each categorical feature (bin) might have a linear relationship with the target.
 - Let's assume that you have age as an input and the likelihood to purchase a cars an target (output).
 - Looking at the data, there is no linear relationship between the two.
 - However, by doing binning, you can now convert the numerical values into bins and this will lead to more accurate model.

VALUE	BIN
0-20	Young
20-70	Mid-age
70-100	Old

BINNING



- One of the key advantages of binning is to improve the model robustness and avoid overfitting.
- Binning is also important when data contains a lot of uncertainty.
- Quantile binning is used to bin the data based on its location in the distribution.
- Binning could be used for both numerical and categorical transformations.
- Let's take a look at a categorical binning example.

VALUE	BIN
Toronto	Ontario
Hamilton	Ontario
Vancouver	British Columbia
Burnaby	British Columbia

QUANTILE BINNING



- **Quantile Binning:** works by assigning same number of observations to each of the bins so each bin will end up having the same number of observations,

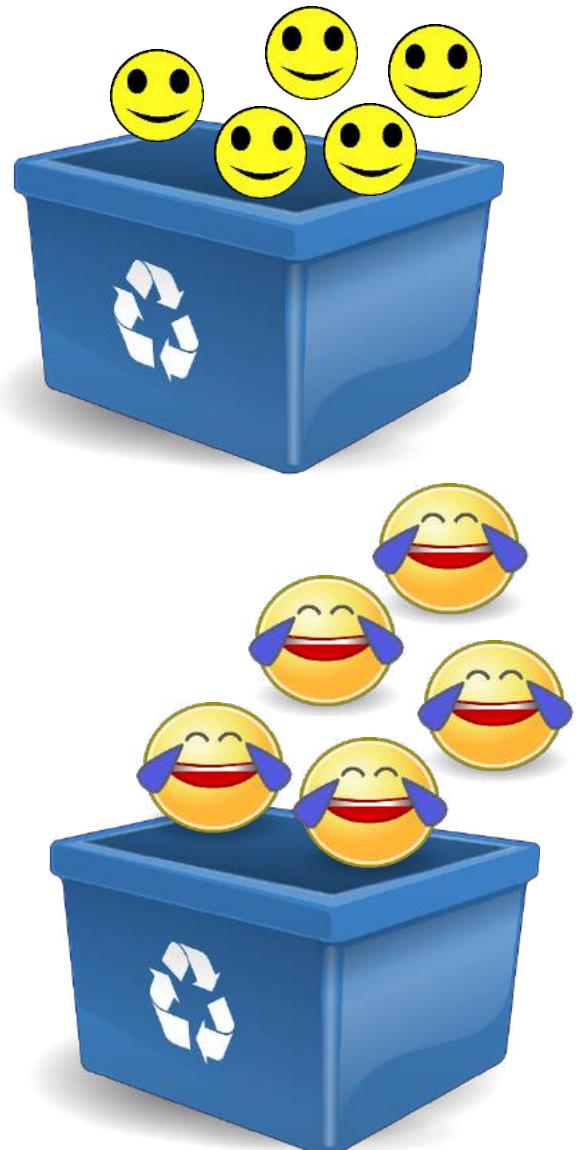


Photo Credit: <https://pixabay.com/vectors/recycling-container-bin-boxes-41078/>

Photo Credit: https://en.m.wikipedia.org/wiki/File:Happy_face.svg

Photo Credit: <https://pixabay.com/illustrations/sad-cry-tear-emotion-mood-face-1533965/>

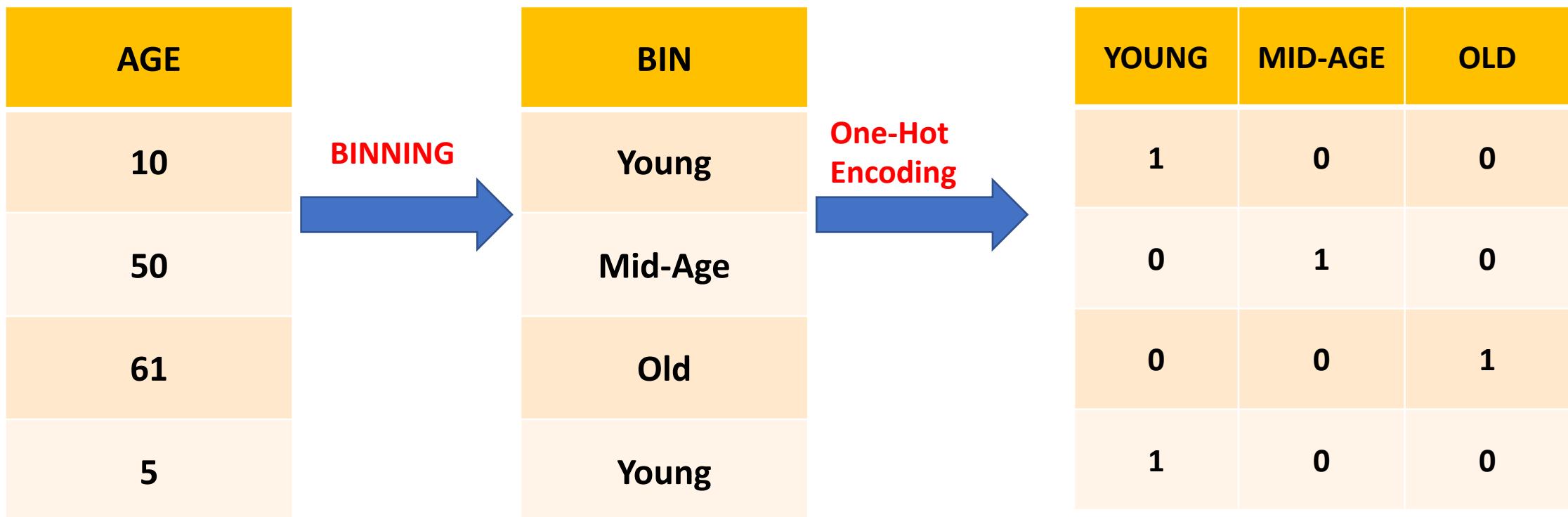
Photo Credit: <https://publicdomainvectors.org/en/free-clipart/Laughing-crying-face/82663.html>

BINNING AND ENCODING



BINNING CRITIREA

VALUE	BIN
0-20	Young
20-60	Mid-age
60-100	Old



LOG TRANSFORM



LOG TRANSFORM



- Log transform is widely used transformation in feature engineering.
- Log transform must be applied to positive values only to avoid getting error.
- Log transform enables data scientists to deal with skewed dataset.
- Log transform is used when the variables span many orders of magnitude such as “income”.
- Since the majority of incomes are very small and very few incomes are large.

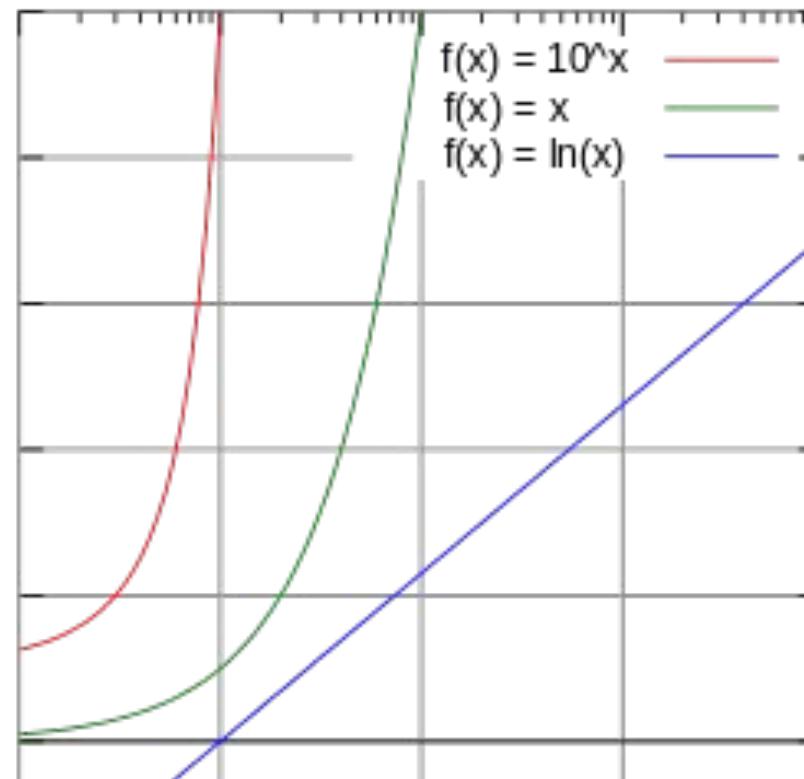


Photo Credit: https://commons.wikimedia.org/wiki/File:Logarithmic_Scales.svg

LOG TRANSFORM



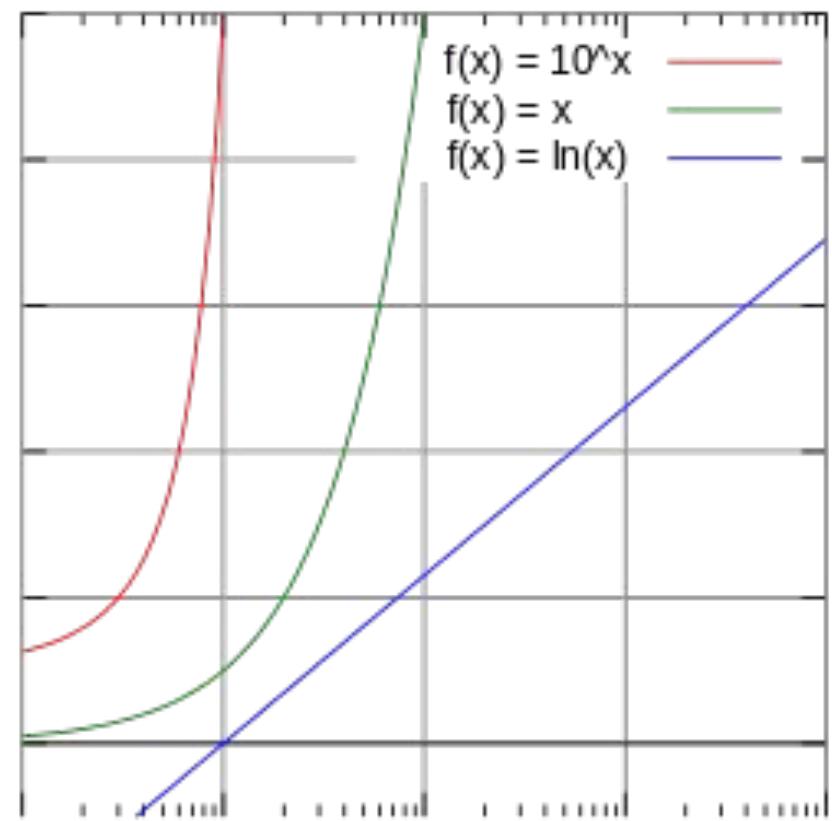
- So this type of data is best deal with in logarithmic scale
- Log transform reduces the negative effect of outliers since it normalizes the magnitude difference.
- Example:

$$\log(x^n) = n * \log(x)$$

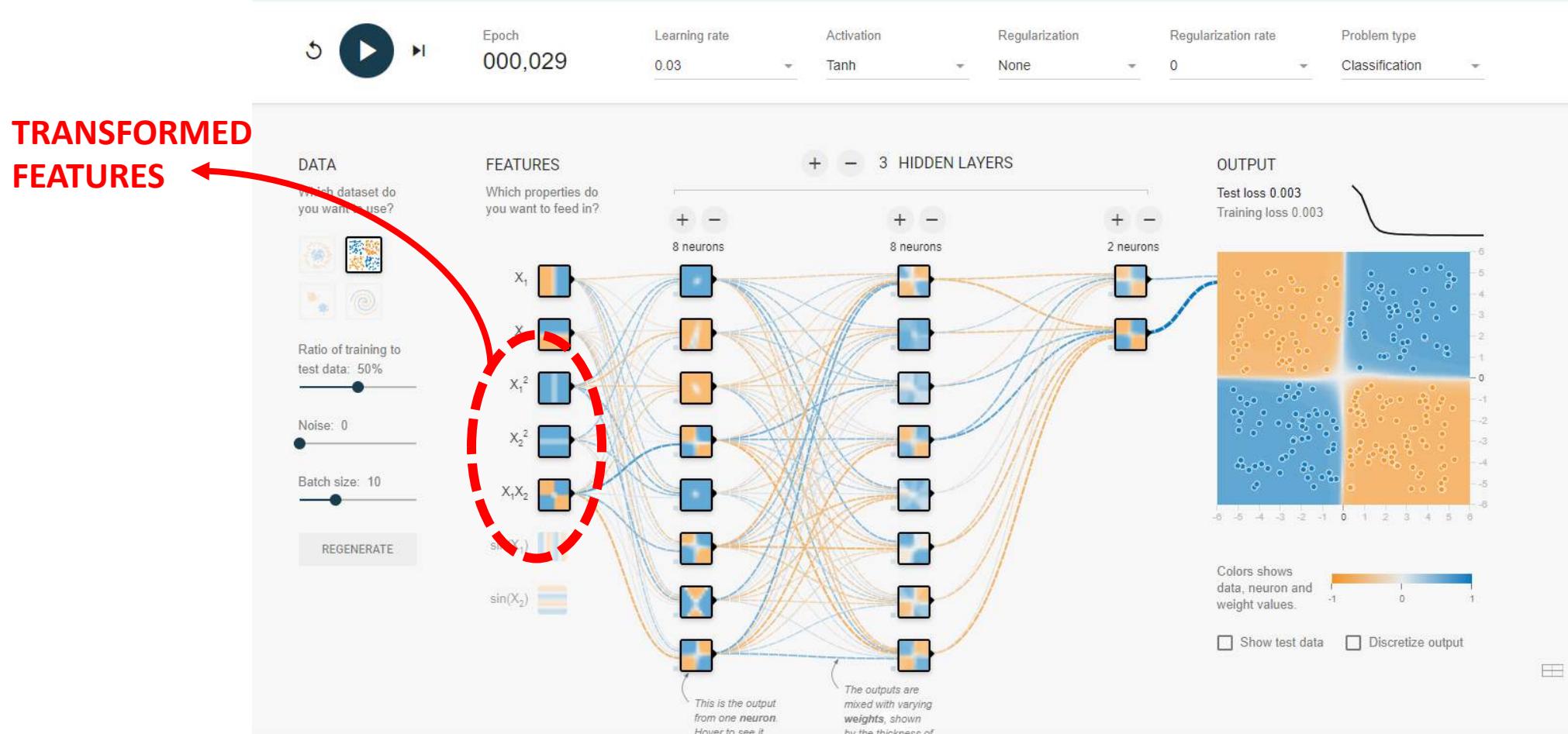
$$\log(10^4) = 4 * \log(10) = 4$$

$$\log(10^3) = 3 * \log(10) = 3$$

- Difference between $\log(10^4) - \log(10^3)$ is much bigger compared to (4-3)



ANOTHER TRANSFORMATION EXAMPLE



<https://playground.tensorflow.org/#activation=tanh&batchSize=10&dataset=circle®Dataset=reg-plane&learningRate=0.03®ularizationRate=0&noise=0&networkShape=4,2&seed=0.17464&showTestData=false&discretize=false&percTrainData=50&x=true&y=true&xTimesY=false&xSquared=false&ySquared=false&cosX=false&sinX=false&cosY=false&sinY=false&collectStats=false&problem=classification&initZero=false&hideText=false>

DATA SHUFFLING



SHUFFLING



- Shuffling is an important step prior to training of machine learning models.
- This is important to avoid any bias or pattern related to the order of the data.
- Especially since data is divided to training, testing and validation.
- Shuffling ensures:
 - Enhanced ML model quality/performance
 - Reduce tendency to overfit the training data

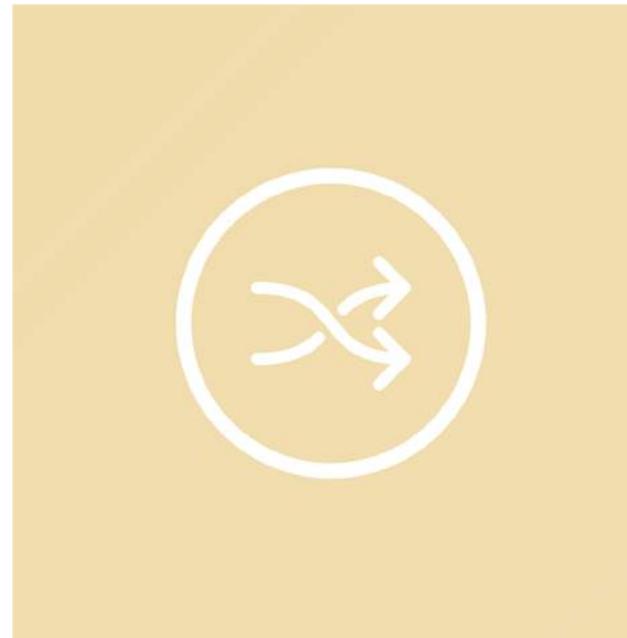
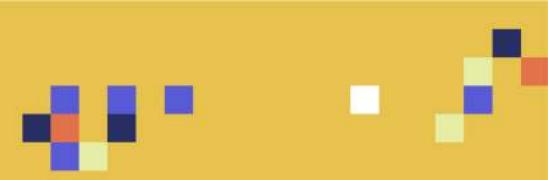


Photo Credit: <https://pixabay.com/vectors/shuffle-icon-player-button-outline-2297766/>

FEATURE SPLITTING

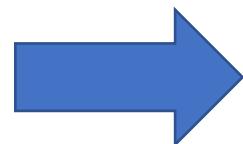


FEATURE SPLITTING



- Splitting features is used to split one feature into two.
- This might improve the performance of the machine learning model by extracting more information.

MOVIE
Titanic (1997)
Notebook (2004)
A Merry Christmas Match (2019)



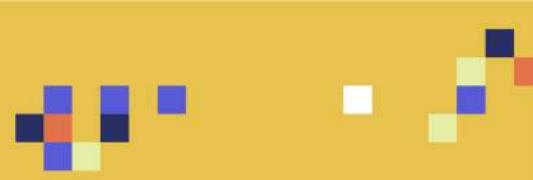
FEATURE #1
Titanic
Notebook
A Merry Christmas Match

FEATURE #2
1997
2004
2019

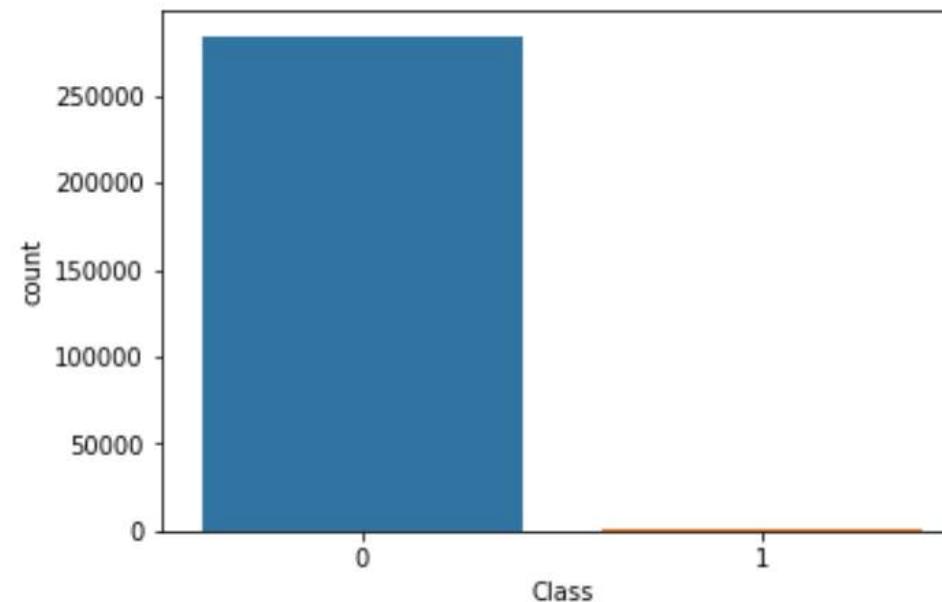
UNBALANCED DATASET



HOW TO DEAL WITH UNBALANCED DATASET?



- Let's take a look at an example data with unbalanced positive and negative classes.
 - Credit card companies need to have the ability to recognize fraudulent credit card transactions so that customers are not charged for items that they did not purchase.
 - Datasets contains transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions.
 - The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.
 - Link to the dataset: <https://www.kaggle.com/mlg-ulb/creditcardfraud/home>



UNBALANCED DATASET: SOLUTIONS?

- You can overcome the unbalanced dataset by doing one of the following:

1. UNDERSAMPLING

- Undersampling is the process of selecting some samples only from the majority class.
- Doing so will remove some of the “unbalance” seen in the data.
- Example: only select some samples that are “non-fraudulent”.

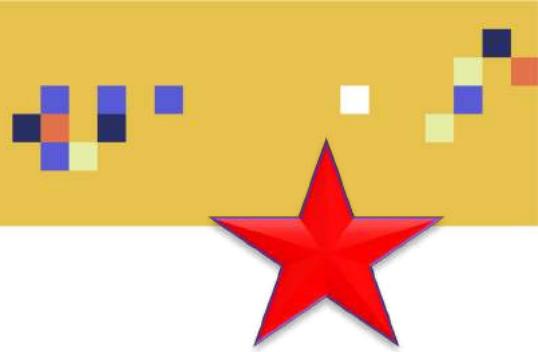
2. OVERSAMPLING

- Oversampling is the process of replicating data from the less representative (minority) class.
- Doing so will help alleviate the “unbalance” seen in the data.
- Example: duplicate samples that are “fraudulent”.

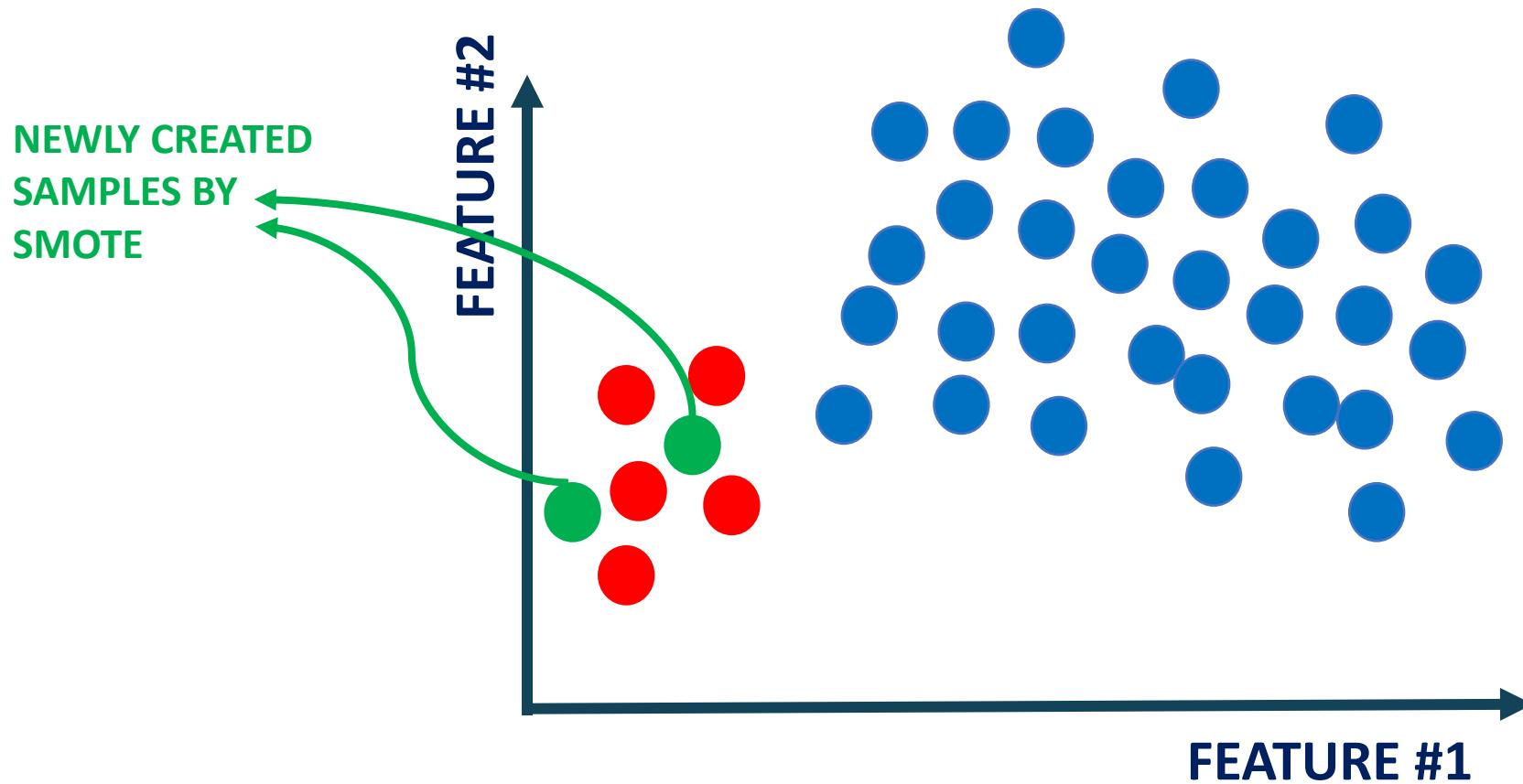
3. GENERATE SYNTHETIC DATASET

- Generate more artificial dataset from the minority class
- Example: Synthetic Minority Oversampling (SMOTE) technique

SYNTHETIC MINORITY OVERSAMPLING (SMOTE) TECHNIQUE



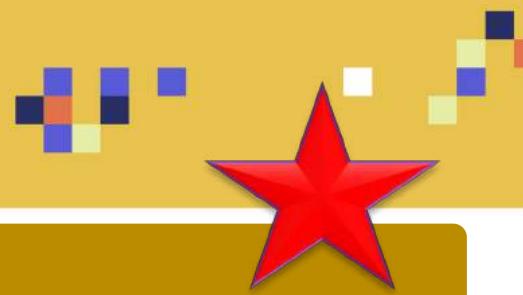
- SMOTE technique works by applying K nearest neighbour to add new dataset.
- For each point from the minority class, SMOTE will calculate k nearest neighbors.
- Then new samples are created based on the oversampling percentage (tunable parameter).



TEXT FEATURE ENGINEERING OVERVIEW



TEXT FEATURE ENGINEERING: OVERVIEW



BAG OF WORDS

- Bag of words is a text feature engineering technique used to tokenize text. Each key includes the word, and each value represents the number of occurrences of that word.

TERM FREQUENCY INVERSE DOCUMENT FREQUENCY (TF-IDF)

- TF-IDF is used to find important keywords in corpus of documents and to ignore less important (more frequent) words.

N-GRAM

- N-Gram is used to create an index of how often words follow one another. It is important for comparing text such as in spam emails detection.

PUNCTUATION REMOVAL

- Getting rid of punctuation from text.

DATES RETRIEVAL

- Extracting date-related information from text.

ORTHOGONAL SPARSE BIGRAMS (OSB)

- OSB is used for text transformation by encoding the words in a text along with how many words have been skipped as well (distance between words).

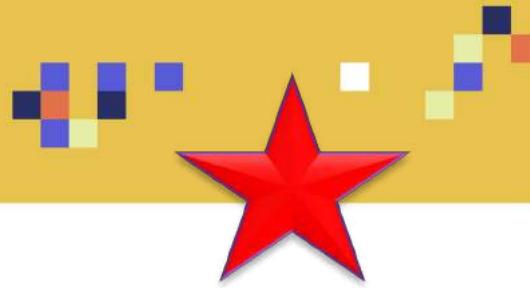
CARTESIAN PRODUCT

- Cartesian transformation works by creating permutations between variables.

TEXT FEATURE ENGINEERING: BAG OF WORDS



BAG OF WORDS



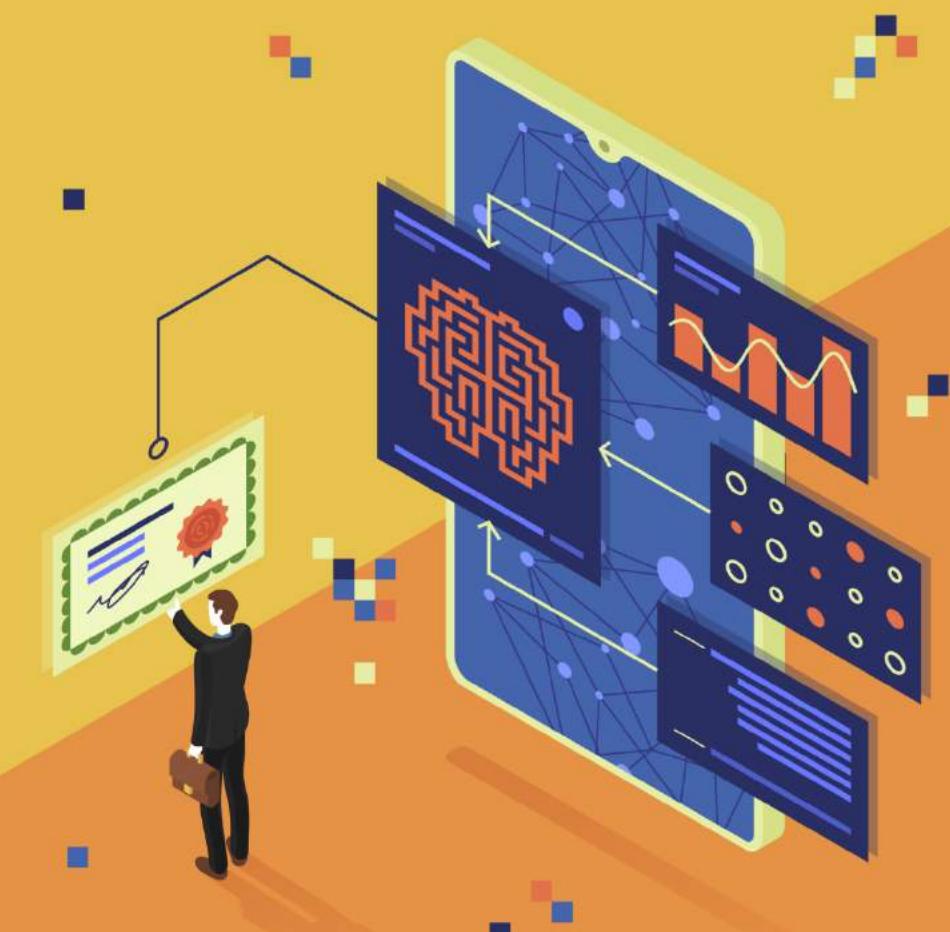
- Bag of words is a text feature engineering technique used to tokenize text.
- Each key includes the word, and each value represents the number of occurrences of that word.

I am happy today. I am learning SageMaker today.

Bag of words = {"I", "am", "happy", "today", "I", "am", "learning", "SageMaker", "today"}

WORD	COUNT
I	2
am	2
happy	1
Today	2
Learning	1
Sagemaker	1

TEXT FEATURE ENGINEERING: REMOVE PUNCTUATION



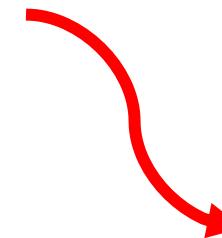
REMOVE PUNCTUATION



- Amazon machine learning uses whitespace as a separator between words (tokens).
- Applying punctuation removal step might be useful in some cases such as bag of words or N-gram transformation.
- Note that lower case transformation could be applied as well.

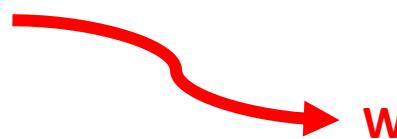
"Welcome to AML - please fasten your seat-belts!"

`{"Welcome", "to", "Amazon", "ML", "-", "please", "fasten", "your", "seat-belts!"}`



WITH PUNCTUATIONS

`{"Welcome", "to", "Amazon", "ML", "please", "fasten", "your", "seat-belts"}`

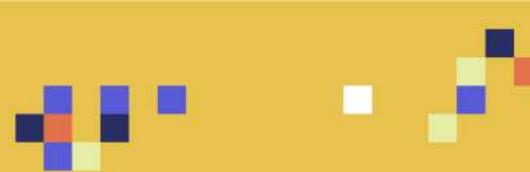


WITHOUT PUNCTUATIONS

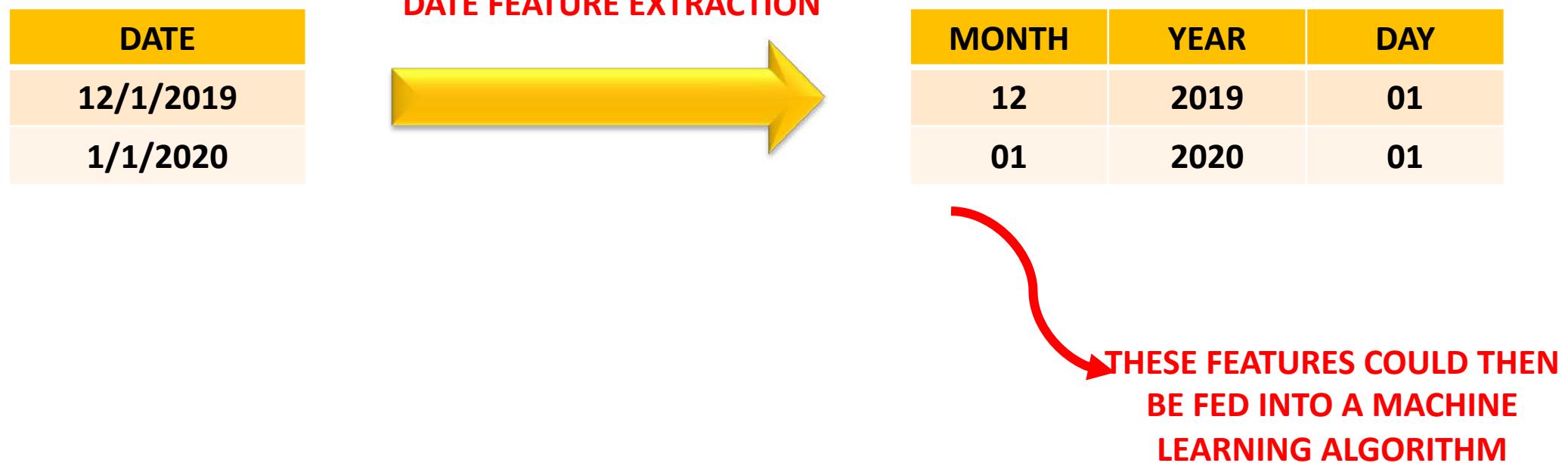
TEXT FEATURE ENGINEERING: DATE FEATURE ENGINEERING



DATE FEATURE ENGINEERING



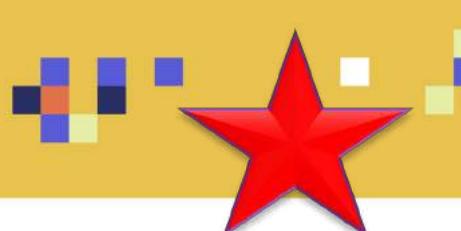
- Date feature engineering is used to extract date-related information as follows:



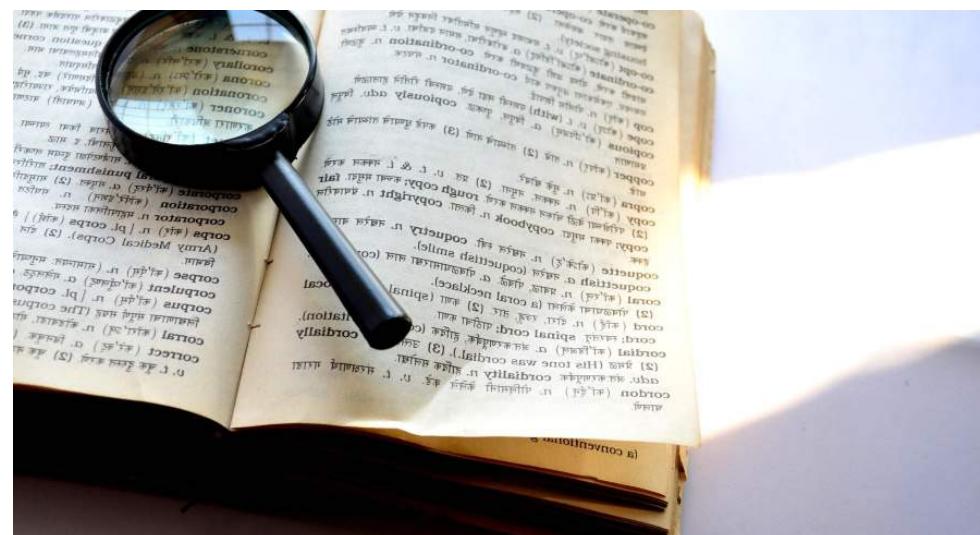
TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY (TF-IDF)



TERM FREQUENCY-INVVERSE DOCUMENT FREQUENCY (TF-IDF)



- TF-IDF stands for "Term Frequency–Inverse Document Frequency" is a numerical statistic used to reflect how important a word is to a document in a collection or corpus of documents.
- TF-IDF is used as a weighting factor during text search processes and text mining.
- The intuition behind the TF-IDF is as follows:
 - if a word appears several times in a given document, this word might be meaningful (more important) than other words that appeared fewer times in the same document.
 - However, if a given word appeared several times in a given document but also appeared many times in other documents, there is a probability that this word might be common frequent word such as 'I' 'am'..etc. (not really important or meaningful!).



<https://pixnio.com/objects/books/magnifying-glass-dictionary-document-paper-book-text>

TERM FREQUENCY-INVVERSE DOCUMENT FREQUENCY (TF-IDF)



- TF: Term Frequency is used to measure the frequency of term occurrence in a document:

$$TF(\text{word}) = \frac{\text{Number of times the 'word' appears in a document}}{\text{Total number of terms in the document}}$$

- IDF: Inverse Document Frequency is used to measure how important a term is:

$$IDF(\text{word}) = \log\left(\frac{\text{Total Number of Documents}}{\text{Number of documents with the term 'word' in it}}\right)$$

- TF-IDF is then calculated as follows:

$$TF - IDF = TF * IDF$$

- TF-IDF generated a matrix of the following shape:

(Number of documents, Number of unique N-Grams)

TERM FREQUENCY-INVVERSE DOCUMENT FREQUENCY (TF-IDF): EXAMPLE



- Let's assume we have a document that contains 1000 words and the term "John" appeared 20 times.
- The Term-Frequency for the word 'John' can be calculated as follows:

$$TF|john = 20/1000 = 0.02$$

- Let's calculate the IDF (inverse document frequency) of the word 'john' assuming that it appears 50,000 times in a 1,000,000 million documents (corpus).

$$IDF|john = \log(1,000,000/50,000) = 1.3$$

- Therefore the overall weight of the word 'john' is as follows"

$$TF - IDF|john = 0.02 * 1.3 = 0.026$$

TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY (TF-IDF): USE CASE



- TF-IDF could be used to develop a search algorithm as follows:
 - Step #1: calculate the TF-IDF for every word present in the corpus
 - Step #2: Once a user enter a word, the corpus is sorted by the TF-IDF for the given word inserted by the user.
 - Step #3: Show the output to user.

```
In [143]: 1 from sklearn.feature_extraction.text import TfidfTransformer  
2  
3 emails_tfidf = TfidfTransformer().fit_transform(spamham_countvectorizer)  
4 print(emails_tfidf.shape)
```

(5728, 37229)

```
In [144]: 1 print(emails_tfidf[:, :])  
2 # Sparse matrix with all the values of IF-IDF
```

(0, 3638)	0.017223322243491098
(0, 23369)	0.118508643434226
(0, 18841)	0.13854196464928686
(0, 10065)	0.07179540742040964
(0, 17696)	0.08994844691767893

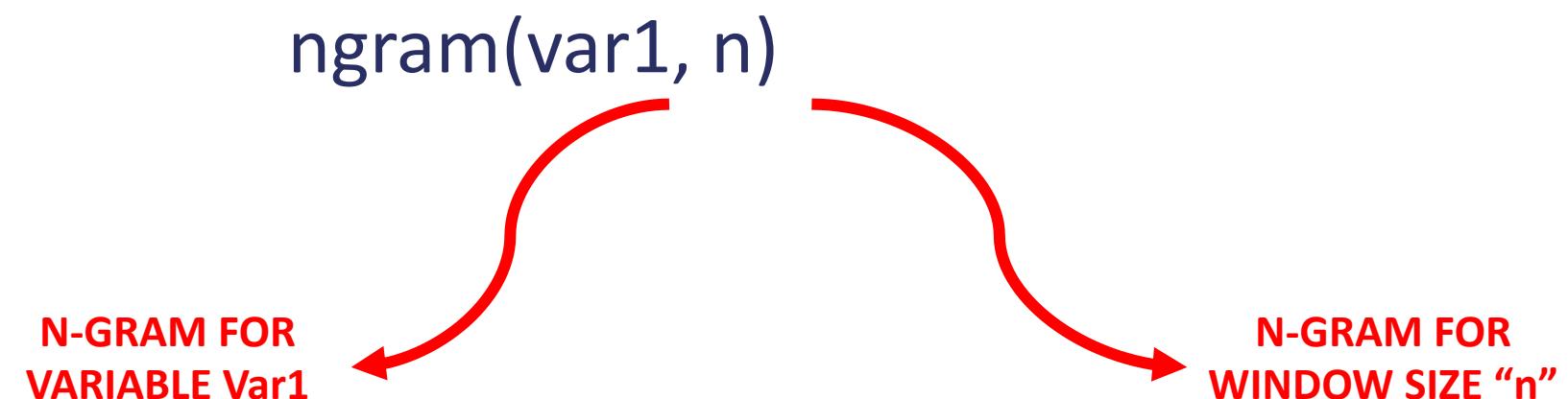
N-GRAM (UNIGRAM Vs. BIGRAM Vs. TRIGRAM) TRANSFORMATION



N-GRAM: INTRODUCTION



- The n-gram transformation converts text data to strings corresponding to sliding a window of n words.
- This is important to perform machine learning; you can now train a model using a single word or group of words.
- You can select the size of “n” as follows:
 - **Unigram** => n=1
 - **Bigram** => n=2
 - **Trigram** => n=3



N-GRAM: UNIGRAM

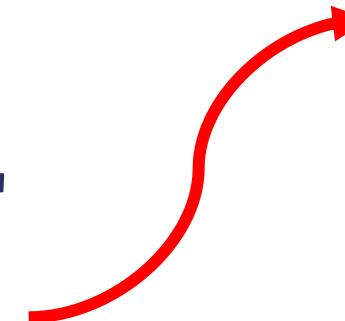


- Let's start with Unigram first:

ngram(var1, 1)

NOTE THAT N-GRAM BREAKS
THEH RAW INPUT DATA BASED
ON WHITESPACE CHARACTERS

"Let's have a happy life"



Unigram: {"Let's", "have", "a" , “happy” , “life”}

N-GRAM: WHAT ABOUT BI-GRAM AND TRI-GRAM?



- Let's assume that now we want to train a machine learning model to understand this but with Bigrams and Trigrams as well:

"Let's have a happy life"

**NOTE THAT BIGRAMS AND
TRIGRAMS INCLUDE
UNIGRAMS AS WELL!**

- We can divide this sentence into unigrams, bigrams and Trigrams
 - Unigram:** {"Let's", "have", "a", "happy", "life"}
 - Bigram:** {"Let's have", "have a", "a happy", "happy life", "Let's", "have", "a", "happy", "life"}
 - Trigram:** {"Let's have a", "have a happy", "a happy life", "Let's have", "have a", "a happy", "happy life", "Let's", "have", "a", "happy", "life"}

UNIGRAM Vs. BIGRAM Vs. TRIGRAM



- Note that n-grams breaks the raw text based on whitespace.
- Any punctuations will be considered a part of the word. (unless you want to remove the punctuation first)

"red, green, blue"

Bigram:

{“red,” , “green,” , “blue” , “red, green” , “green, blue”}



**NOTE THAT THE COMMA IS CONSIDERED
A PART OF THE WORD!**

UNIGRAM Vs. BIGRAM Vs. TRIGRAM



“Let’s have a happy life”
“Let’s have a better job”

ORTHOGONAL SPARSE BIGRAM (OSB)



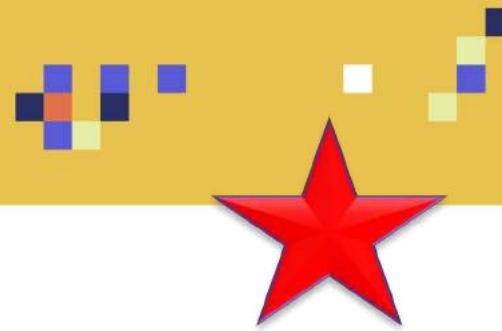
ORTHOGONAL SPARSE BIGRAM (OSB)



- OSB is used for text transformation and it is an alternative to bi-gram transformation.
- OSB encodes the words in a text along with how many words have been skipped as well (distance between words).
- OSB might perform better than n-grams if text data has large text fields (10 or more words)
- OSB works as follows:
 - Slide window of size n over text.
 - Output every pair of words that includes the first word in the window.
 - Join words with underscore “_” and include another “_” for every skipped token.

CONFUSED? LET'S TAKE A LOOK AT AN EXAMPLE!!

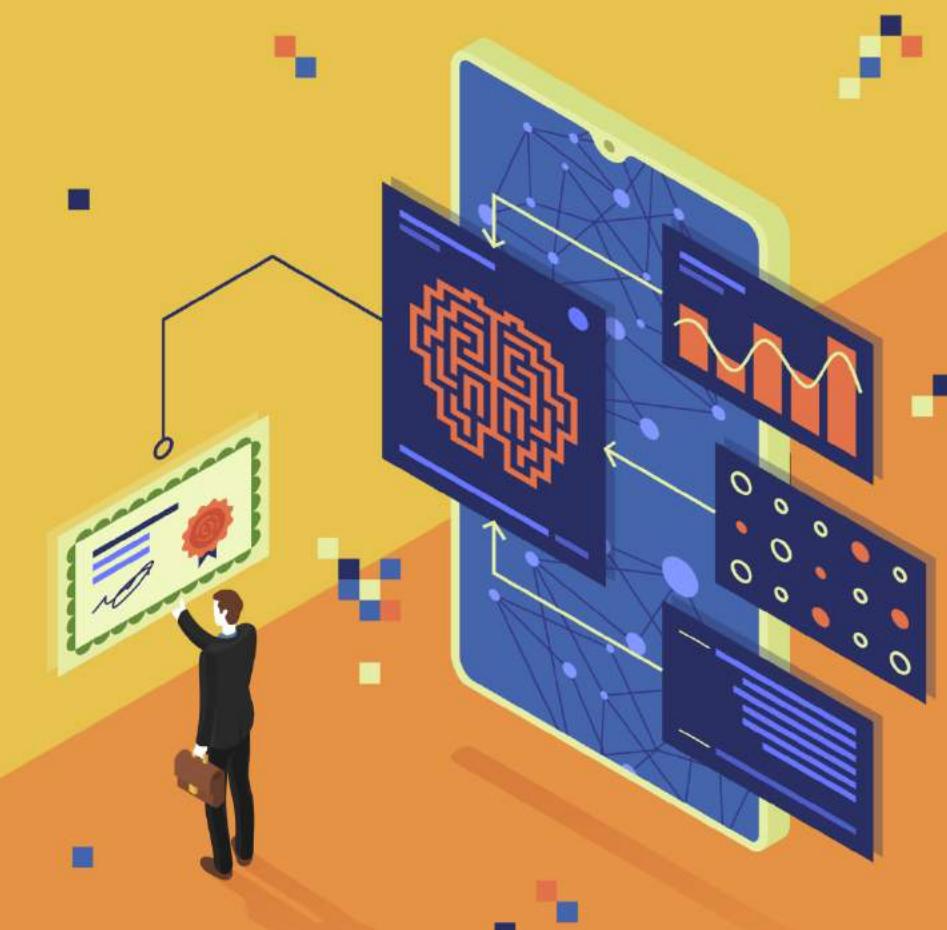
ORTHOGONAL SPARSE BIGRAM (OSB): EXAMPLE



- Example: "The quick brown fox jumps over the lazy dog", and OSBs of size 4.
- The six four-word windows, and the last two shorter windows from the end of the string are shown in the following example, as well OSBs generated from each:

"The quick brown fox", {The_quick, The_brown, The_fox}
"quick brown fox jumps", {quick_brown, quick_fox, quick_jumps}
"brown fox jumps over", {brown_fox, brown_jumps, brown_over}
"fox jumps over the", {fox_jumps, fox_over, fox_the}
"jumps over the lazy", {jumps_over, jumps_the, jumps_lazy}
"over the lazy dog", {over_the, over_lazy, over_dog}
"the lazy dog", {the_lazy, the_dog} "lazy dog", {lazy_dog}

CARTESIAN PRODUCT TRANSFORMATION

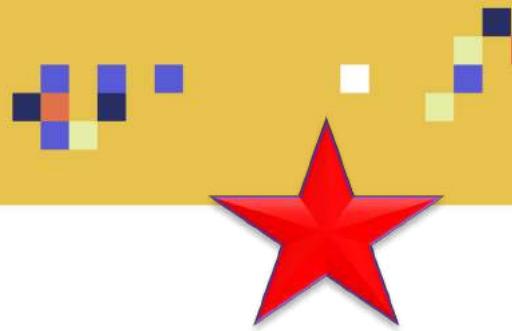


CARTESIAN PRODUCT TRANSFORMATION



- The Cartesian transformation works by creating permutations between variables.
- Cartesian transformation works well if there is an interactions between features.
- Example:
 - purchase_product, education, job
 - 0, university.degree, technician
 - 0, high.school, services
 - 1, university.degree, admin
- There might be interaction between education and job in deciding whether customer would buy a product or not so Cartesian product might work well in this case.
 - purchase_product, education_job_interaction
 - 0, university.degree_technician
 - 0, high.school_services
 - 1, university.degree_admin

CARTESIAN PRODUCT TRANSFORMATION



- The Cartesian transformation works with text as well as follows:

Textbook	Title	Binding	Cartesian product of no_punct>Title) and Binding
0	Deep Learning: basics, applications	Hardcover	{"Deep_Hardcover", "Learning_Hardcover", "basics_Hardcover", "applications_Hardcover"}
1	Machine Learning Practical	Softcover	{"Machine_Softcover", "Learning_Softcover", "Practical_Softcover"}

AWS MACHINE LEARNING CERTIFICATION



DOMAIN #3: MODELING (36% EXAM MARK)



AWS ML CERTIFICATION EXAM DOMAINS



Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #3 OVERVIEW:

SECTION #8: MACHINE AND DEEP LEARNING BASICS – PART #1

- Artificial Neural Networks Basics: Single Neuron Model
- Activation Functions
- Multi-Layer Perceptron Model
- How do Artificial Neural Networks Train?
- ANN Parameters Tuning – Learning rate and batch size
- Tensorflow playground
- Gradient Descent and Backpropagation
- Overfitting and Under fitting
- How to overcome overfitting?
- Bias Variance Trade-off
- L1 Regularization
- L2 Regularization

SECTION #9: MACHINE AND DEEP LEARNING BASICS – PART #2

- Artificial Neural Networks Architectures
- Convolutional Neural Networks
- Recurrent Neural Networks
- Vanishing Gradient Problem
- LSTM Networks
- Model Performance Assessment – Confusion Matrix
- Model Performance Assessment – Precision, recall, F1-score
- Model Performance Assessment – ROC, AUC, Heatmap, and RMSE
- K-Fold Cross validation
- Transfer Learning
- Ensemble Learning – Bagging and Boosting

DOMAIN #3 OVERVIEW:



SECTION #10: MACHINE AND DEEP LEARNING IN AWS – BUILT-IN ALGORITHMS PART #1

- AWS SageMaker
- Deep Learning on AWS
- SageMaker Built-in algorithms
- Object Detection
- Image Classification
- Semantic Segmentation
- SageMaker Linear Learner
- Factorization Machines
- XG-Boost
- SageMaker Seq2Seq
- SageMaker DeepAR
- SageMaker Blazing Text

SECTION #11: MACHINE AND DEEP LEARNING IN AWS – BUILT-IN ALGORITHMS PART #2

- Object2Vec
- Random Cut Forest
- Neural Topic Model
- LDA
- K-Nearest Neighbours (KNN)
- K Means
- Principal Component Analysis (PCA)
- IP insights
- Reinforcement Learning
- Automatic Model Tuning
- SageMaker and Spark

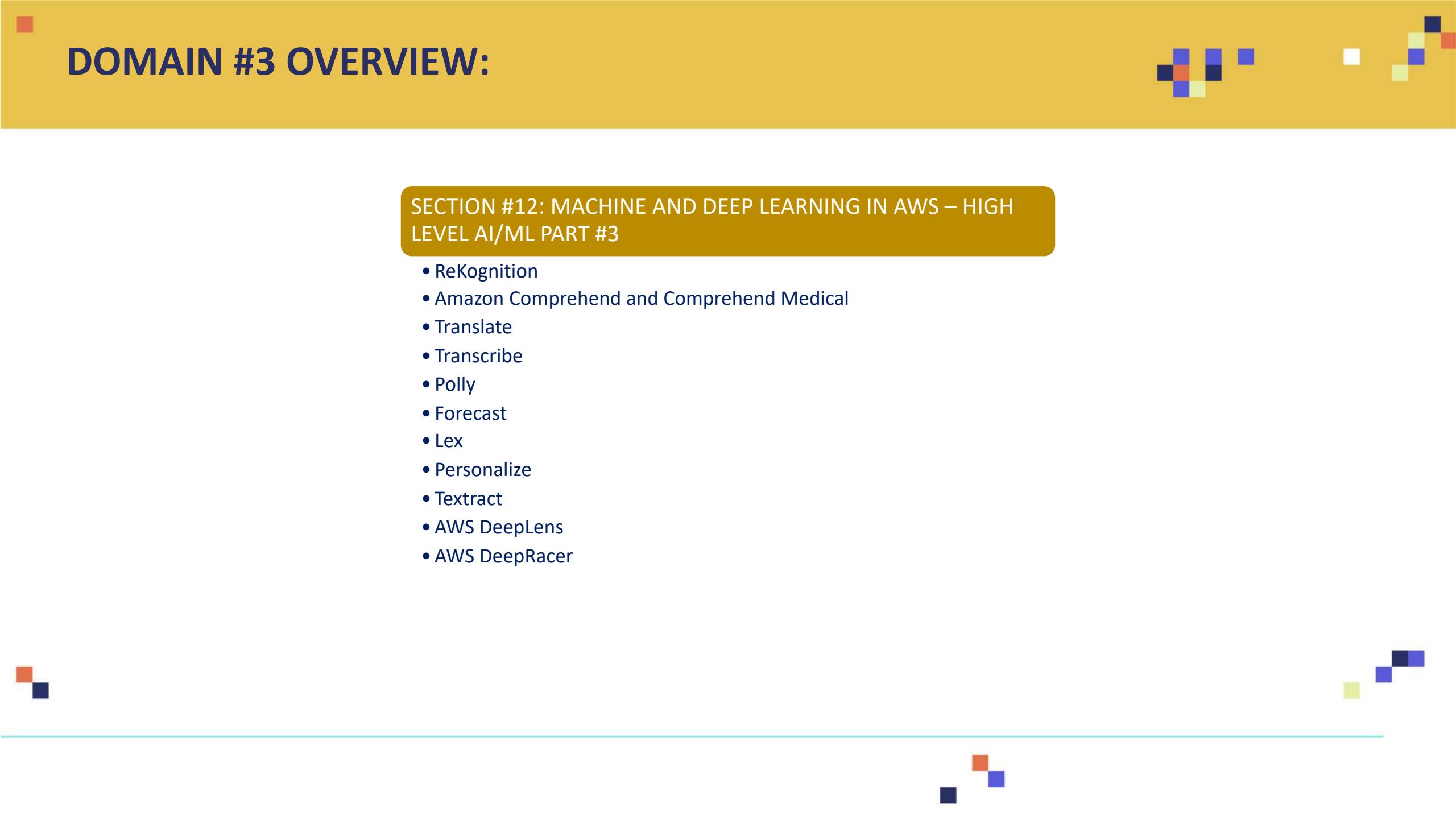


DOMAIN #3 OVERVIEW:



SECTION #12: MACHINE AND DEEP LEARNING IN AWS – HIGH LEVEL AI/ML PART #3

- ReKognition
- Amazon Comprehend and Comprehend Medical
- Translate
- Transcribe
- Polly
- Forecast
- Lex
- Personalize
- Textract
- AWS DeepLens
- AWS DeepRacer



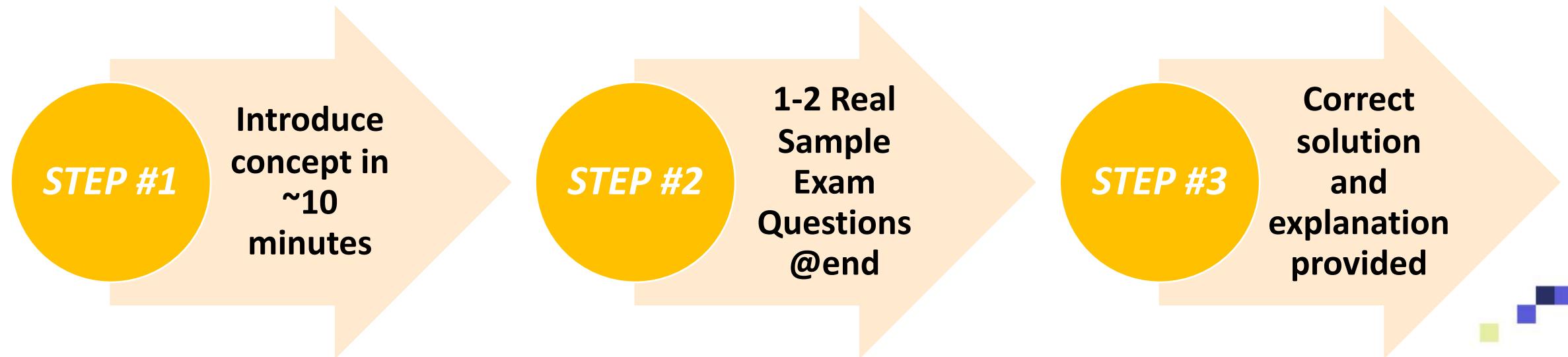
LECTURE DESIGN



- We know how hard it is to study for an exam especially if you have a busy schedule.
- This course is designed to be extremely on point and optimized to pass the exam.

No boring content. Zero unnecessary information.

- Here's the lecture structure that we will follow:

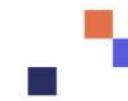


VALUABLE PRIZE!

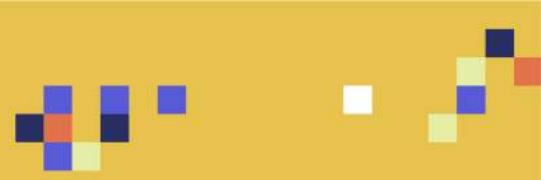


- For those of you who will successfully complete the entire Module and watch the videos till the end, they will receive a valuable prize!

**10 NEW SAMPLE EXAM QUESTIONS + COMPLETE
ANSWER KEY**



GAME AND MINI CHALLENGES!



- Unfortunately, you can't skip the videos.
- You have to collect a code throughout the lectures to unlock the exam.
- Special characters will appear at random moments throughout the video.
- You will need to collect the code and enter it to a website to access the material.
- That's what the final code might look like!

F 2 @ 9 & B



ARTIFICIAL NEURAL NETWORKS BASICS



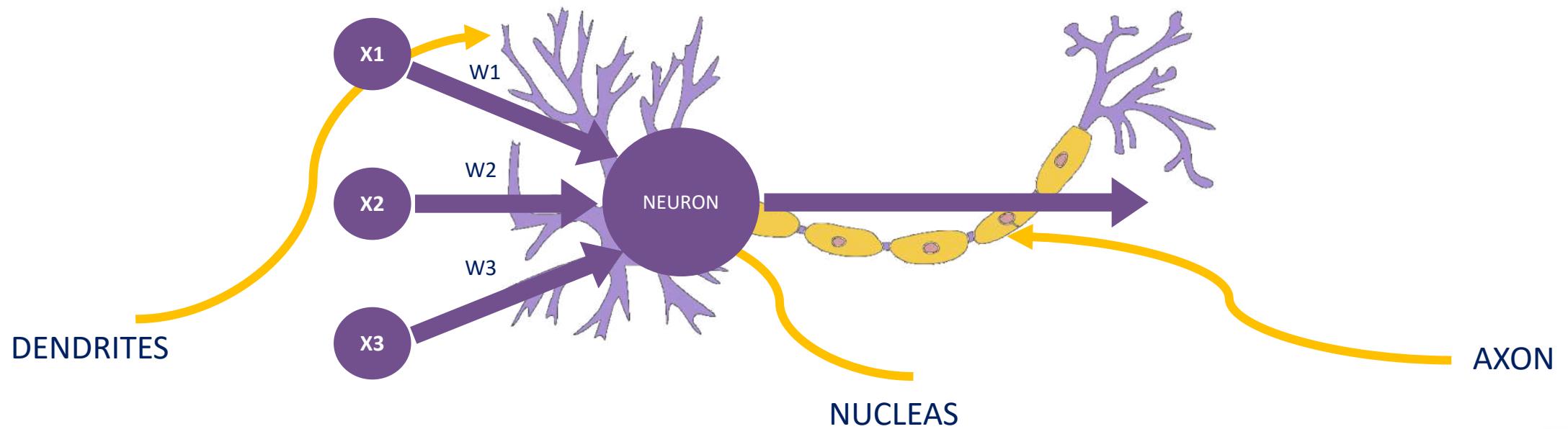
SINGLE NEURON MODEL



NEURON MATHEMATICAL MODEL



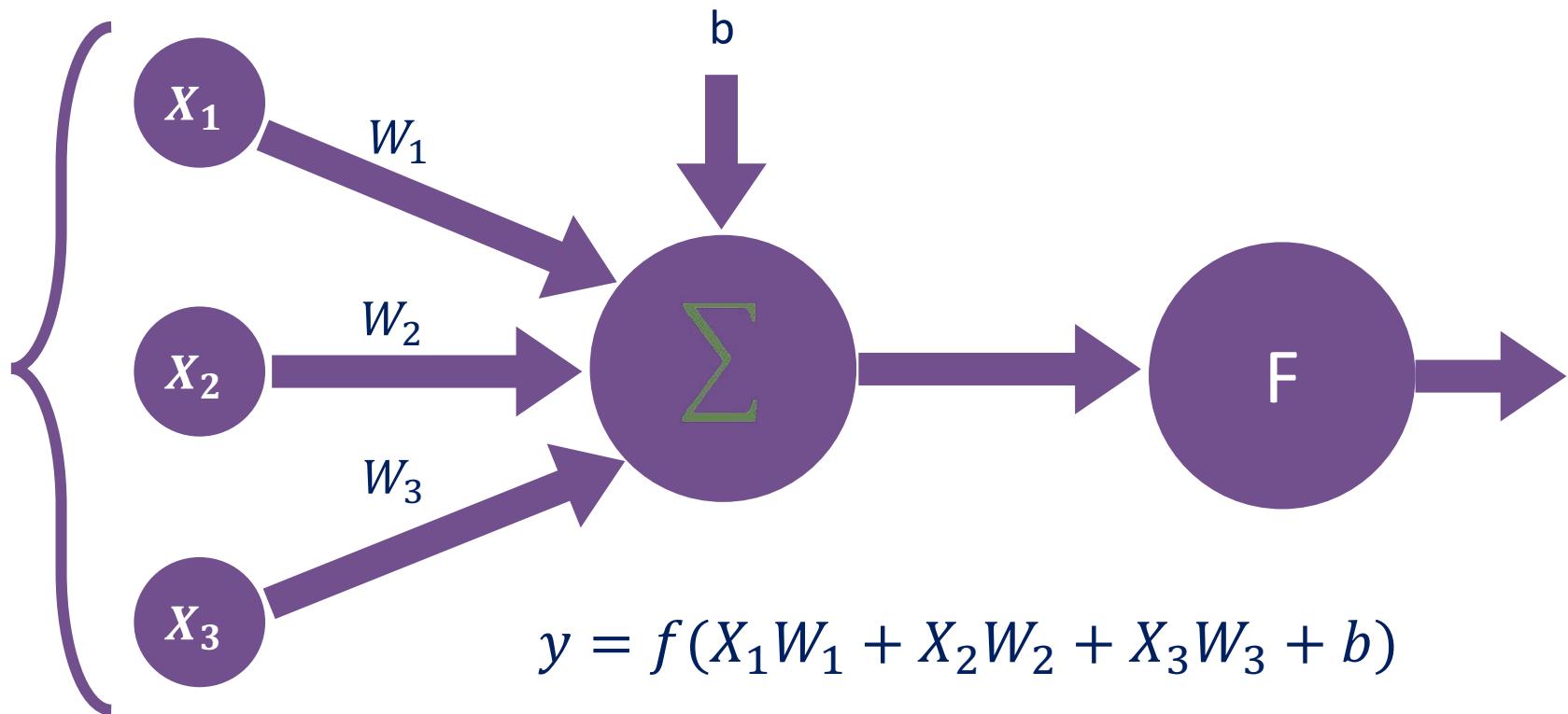
- The neuron collects signals from input channels named dendrites, processes information in its nucleus, and then generates an output in a long thin branch called axon.



NEURON MATHEMATICAL MODEL

- Bias allows to shift the activation function curve up or down.
- Number of adjustable parameters = 4 (3 weights and 1 bias).
- Activation function “F”.

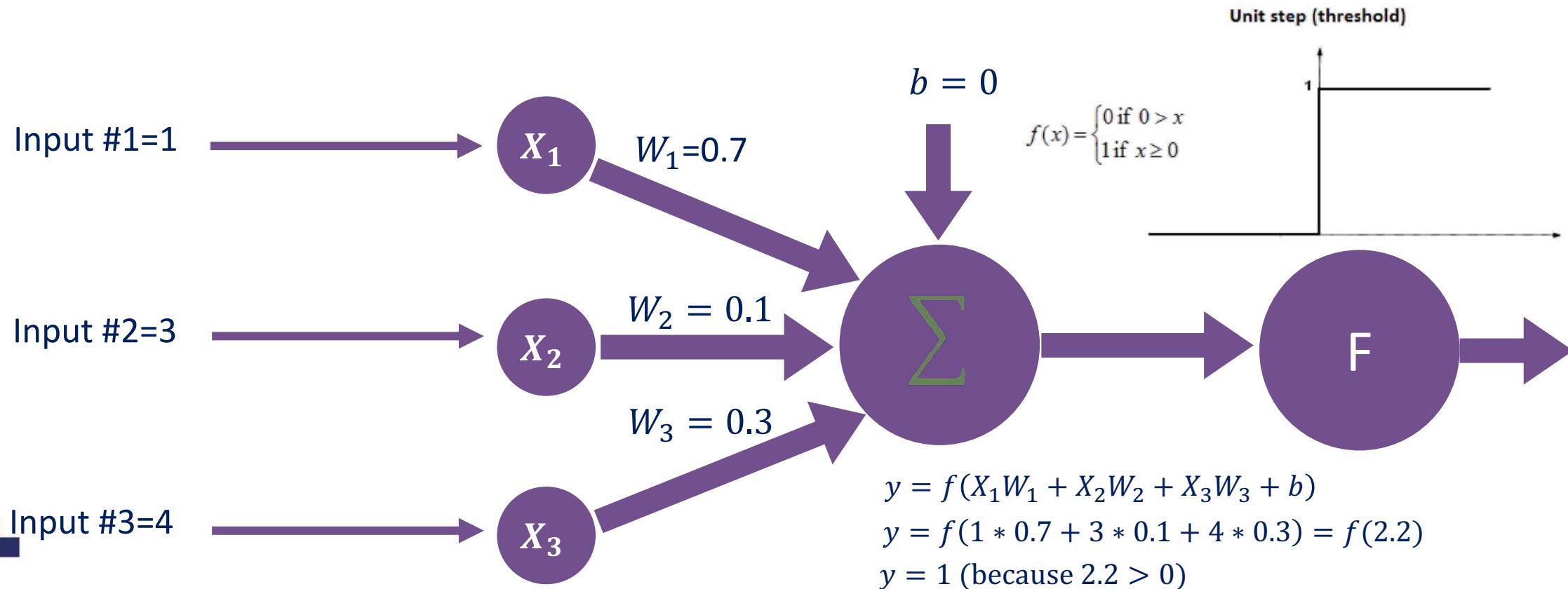
INPUTS/
INDEPENDENT
VARIABLES



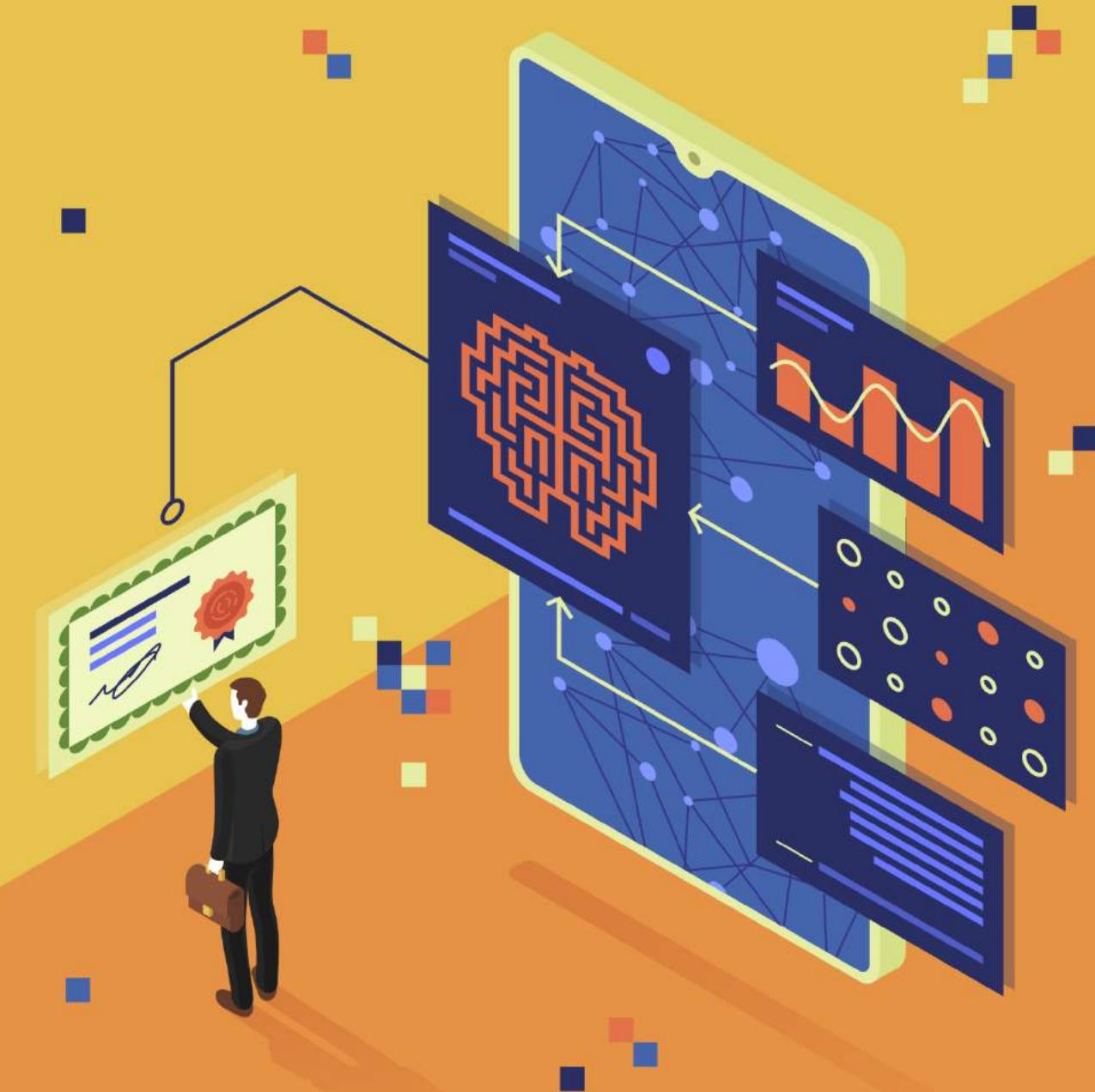
$$y = f(X_1W_1 + X_2W_2 + X_3W_3 + b)$$

NEURON MODEL IN ACTION!

- Let's assume an activation function of Unit Step.
- The activation functions is used to map the input between (0, 1).



ACTIVATION FUNCTIONS

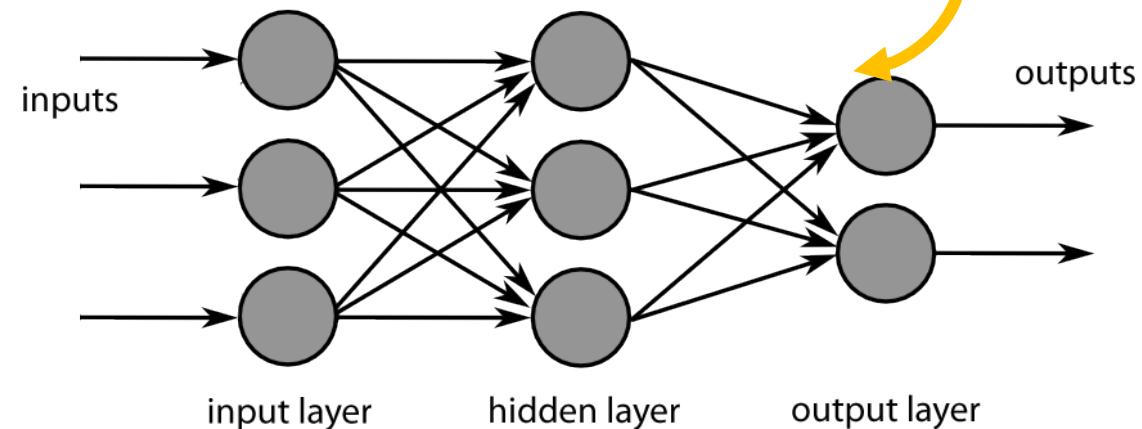
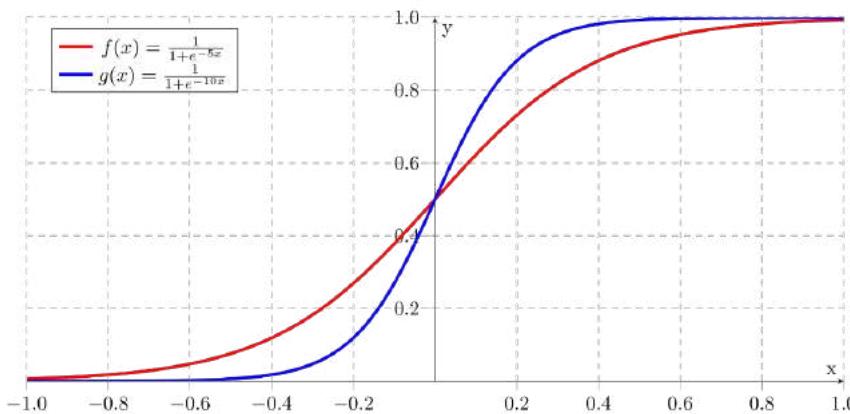
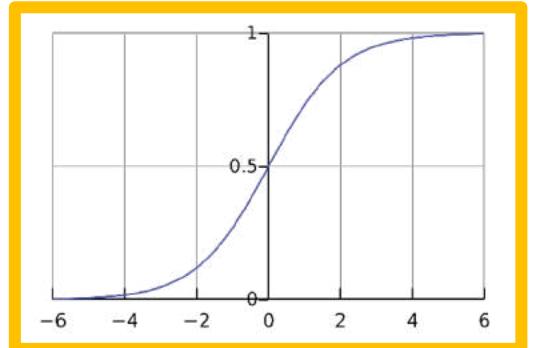


ACTIVATION FUNCTIONS



- **SIGMOID:**

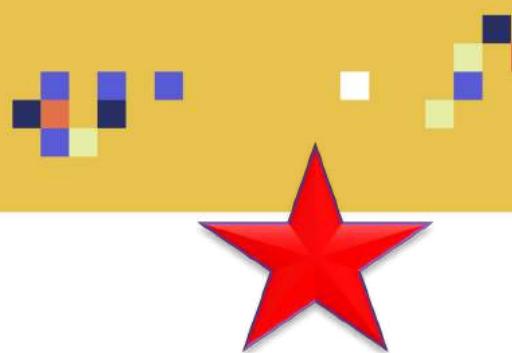
- Takes a number and sets it between 0 and 1
- Converts large negative numbers to 0 and large positive numbers to 1.
- Generally used in output layer.



- Photo credit: <https://commons.wikimedia.org/wiki/File:Sigmoid-function.svg>
- Photo Credit: https://fr.m.wikipedia.org/wiki/Fichier:MultilayerNeuralNetworkBigger_english.png
- Photo Credit: <https://commons.wikimedia.org/wiki/File:Logistic-curve.svg>

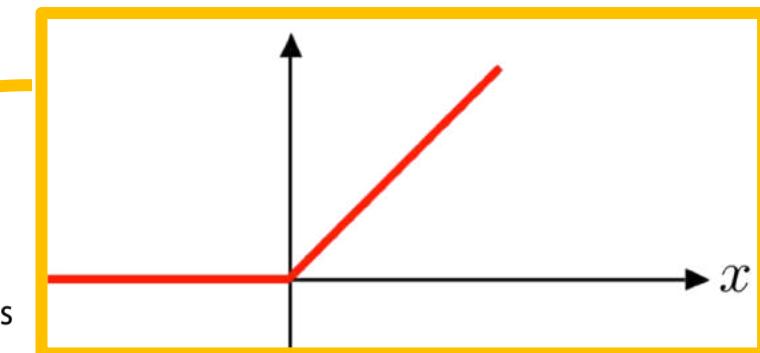
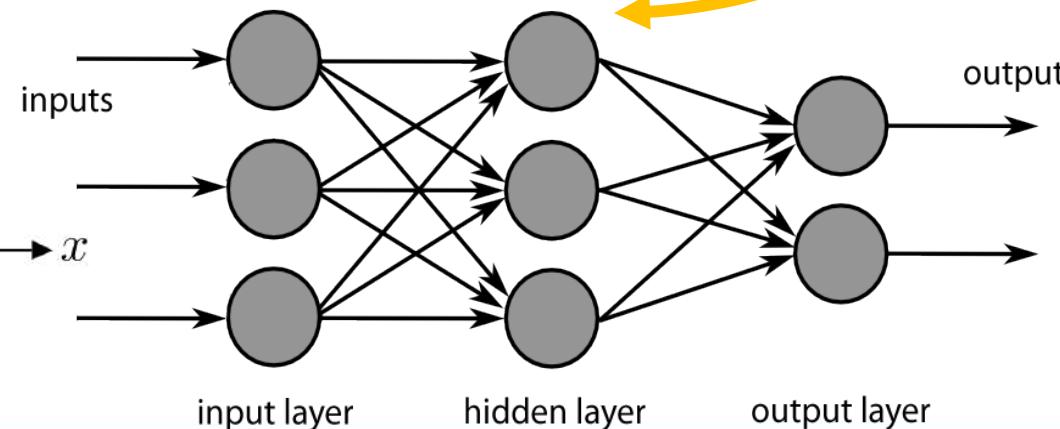
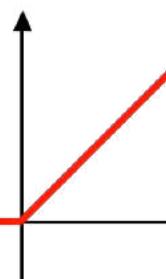


ACTIVATION FUNCTIONS



- **RELU (RECTIFIED LINEAR UNITS):**
 - if input $x < 0$, output is 0 and if $x > 0$ the output is x .
 - RELU does not saturate so it avoids vanishing gradient problem.
 - It uses simple thresholding so it is computationally efficient.
 - Generally used in hidden layers.

$$\text{ReLU}(x) \triangleq \max(0, x)$$

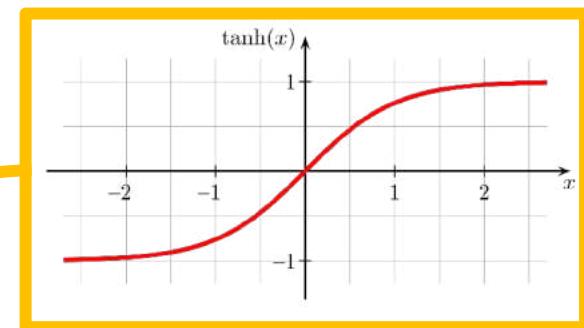
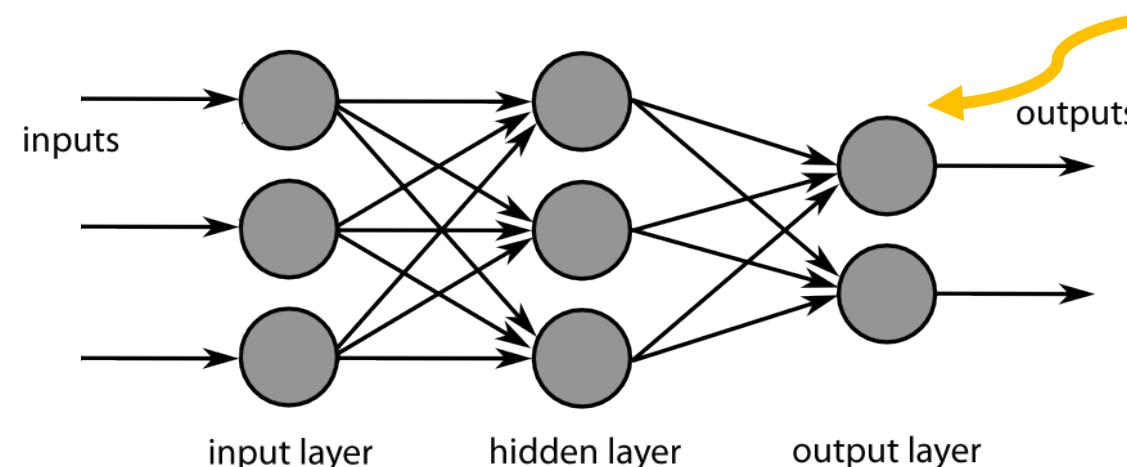
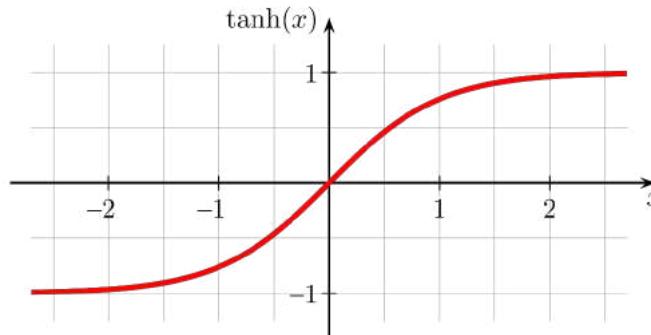


- Photo credit: https://commons.wikimedia.org/wiki/File:ReLU_and_Nonnegative_Soft_Thresholding_Functions.svg
- Photo Credit: https://fr.m.wikipedia.org/wiki/Fichier:MultiLayerNeuralNetworkBigger_english.png

ACTIVATION FUNCTIONS



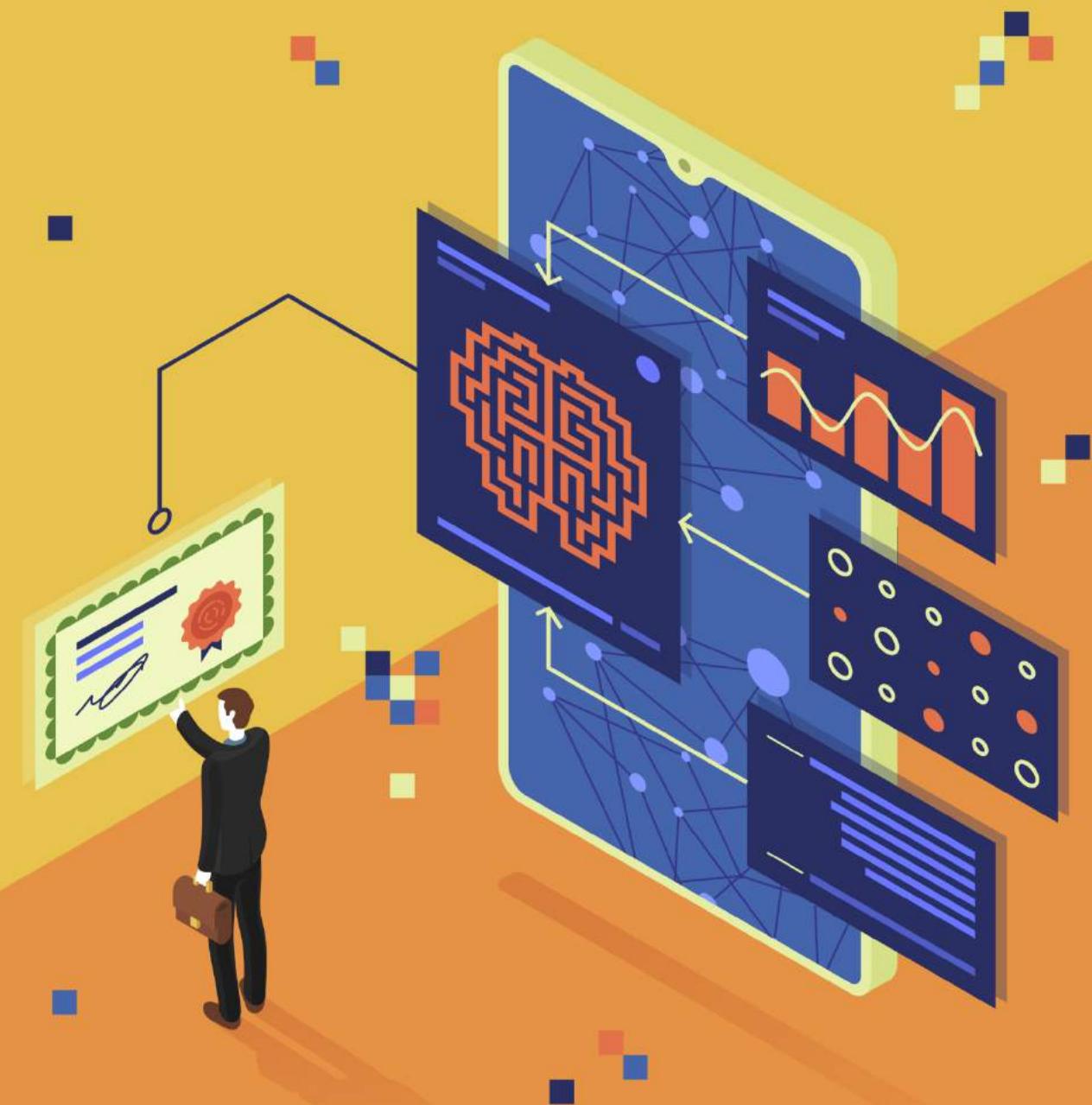
- **HYPERBOLIC TANGENT ACTIVATION FUNCTION:**
 - “Tanh” is similar to sigmoid, converts number between -1 and 1.
 - Unlike sigmoid, tanh outputs are zero-centered (range: -1 and 1).
 - Tanh suffers from vanishing gradient problem so it kills gradients when saturated.
 - In practice, tanh is preferable over sigmoid.



- Photo credit: https://commons.wikimedia.org/wiki/File:Hyperbolic_Tangent.svg
- Photo Credit: https://fr.m.wikipedia.org/wiki/Fichier:MultiLayerNeuralNetworkBigger_english.png



MULTI-NEURON MODEL (MULTI-LAYER PERCEPTRON MODEL)

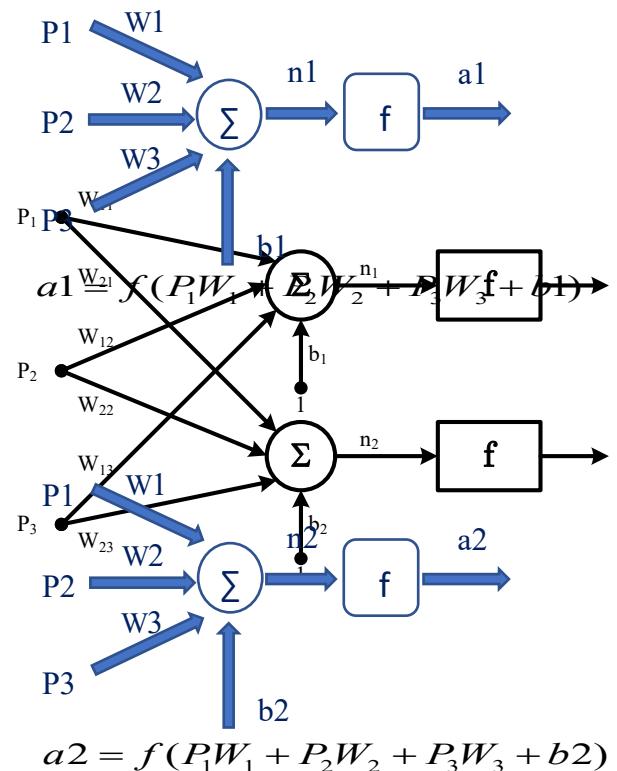


MULTI-LAYER PERCEPTRON NETWORK

- The network is represented by a matrix of weights, inputs and outputs.
- Total Number of adjustable parameters = 8:
- Weights = 6
- Biases = 2

Matrix Representation

$$P = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix}$$
$$W = \begin{bmatrix} W_{11} & W_{12} & W_{13} \\ W_{21} & W_{22} & W_{23} \end{bmatrix}$$
$$b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$
$$a = f(W \times P + b)$$

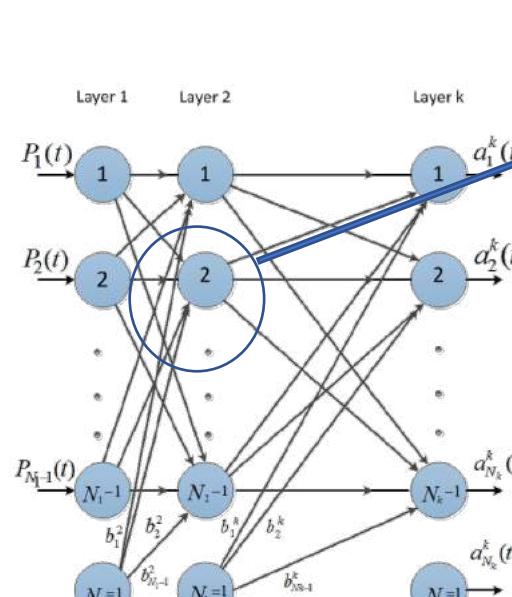


MULTI-LAYER PERCEPTRON NETWORK

- Let's connect multiple of these neurons in a multi-layer fashion.
- The more hidden layers, the more "deep" the network will get.

$$P = \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_{N_1} \end{bmatrix}$$

$$\begin{bmatrix} W_{11} & W_{12} & \dots & W_{1,N_1} \\ W_{21} & W_{22} & \ddots & W_{2,N_1} \\ \vdots & \ddots & \ddots & \vdots \\ W_{m-1,1} & W_{m-1,2} & \dots & W_{m-1,N_1} \\ W_{m,1} & W_{m,2} & \dots & W_{m,N_1} \end{bmatrix}$$



Layer n+1

$x_i^{n+1}(t) = \varphi(\sum_{j=1}^{N_n} w_{i,j}^n x_j^n(t))$

Node $(n+1, i)$ representation

Non-Linear Sigmoid Activation function

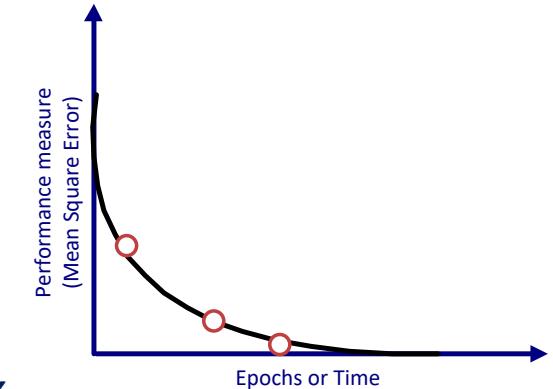
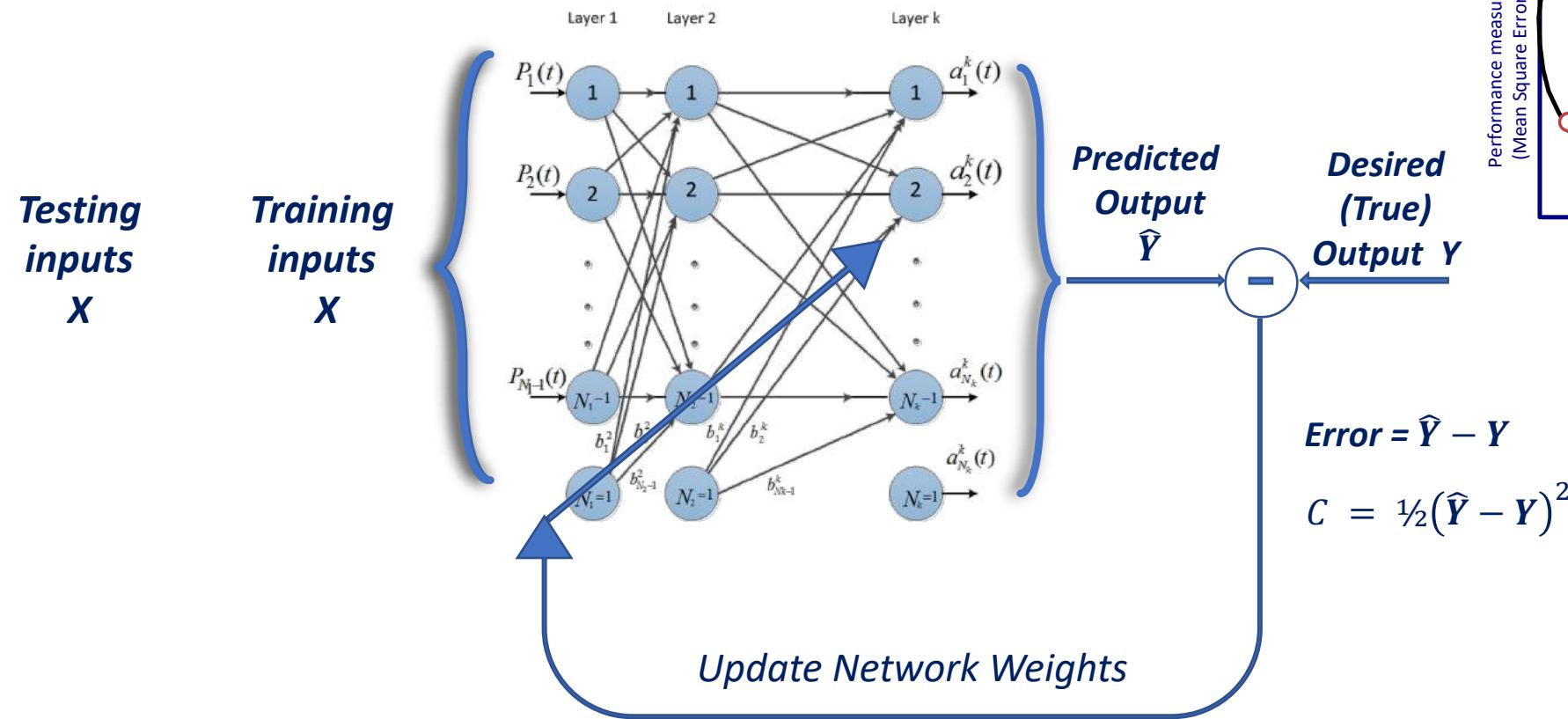
$$\varphi(w) = \frac{1}{1 + e^{-w}}$$

m : number of neurons in the hidden layer
 N_1 : number of inputs

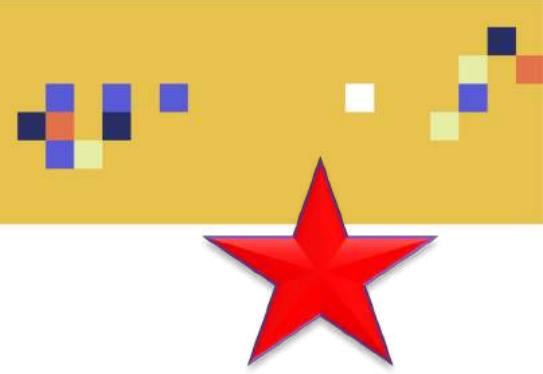
HOW DO ANNS TRAIN?



SUPERVISED ANN TRAINING



ANN TRAINING STRATEGIES



Supervised learning

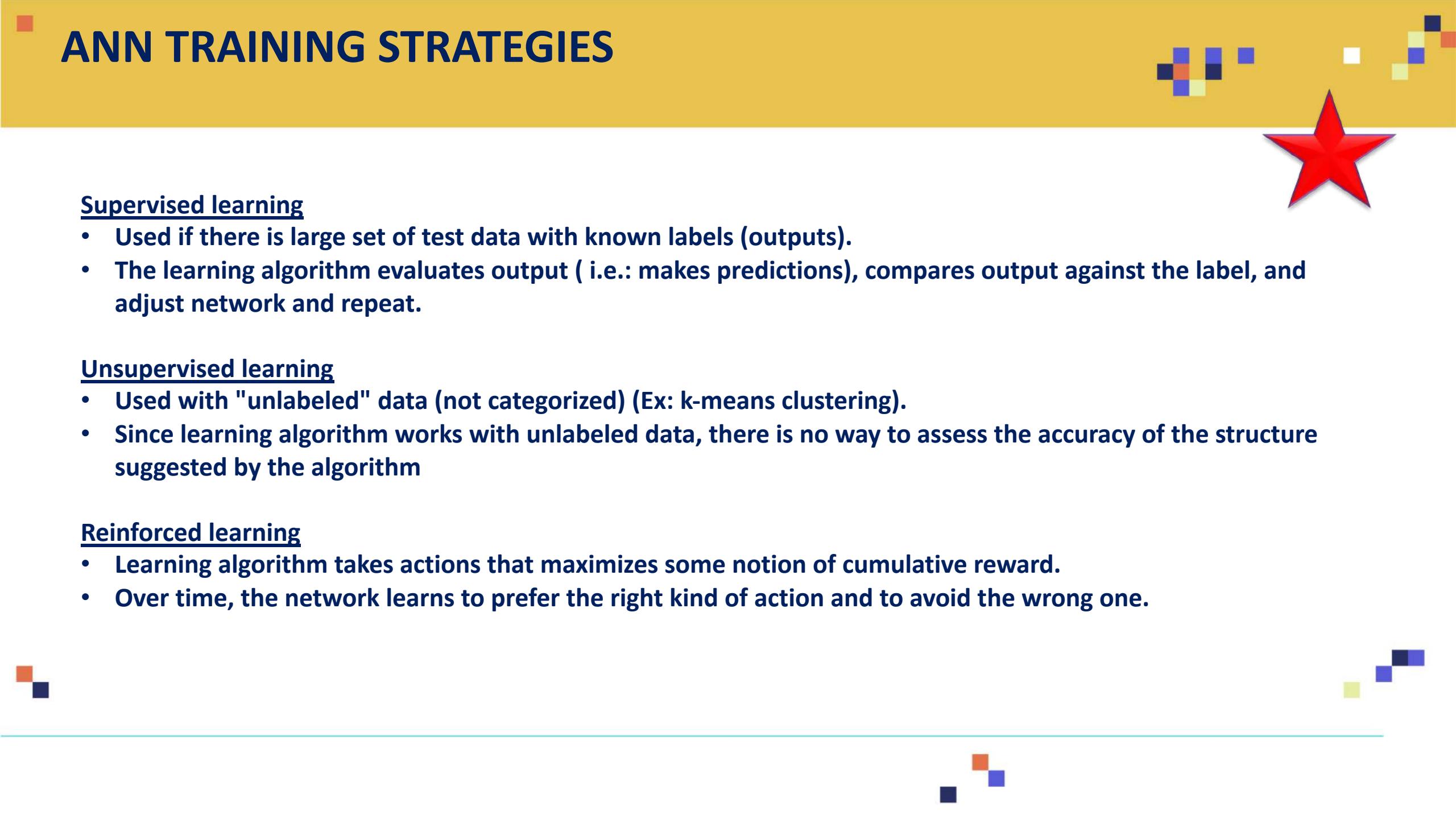
- Used if there is large set of test data with known labels (outputs).
- The learning algorithm evaluates output (i.e.: makes predictions), compares output against the label, and adjust network and repeat.

Unsupervised learning

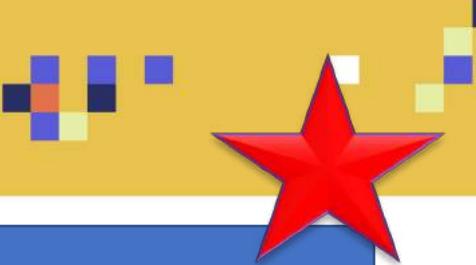
- Used with "unlabeled" data (not categorized) (Ex: k-means clustering).
- Since learning algorithm works with unlabeled data, there is no way to assess the accuracy of the structure suggested by the algorithm

Reinforced learning

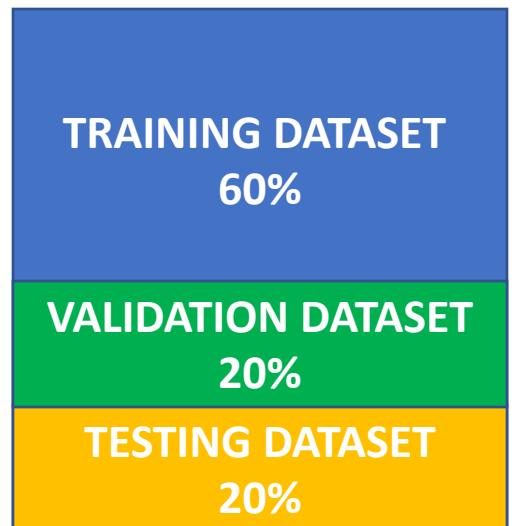
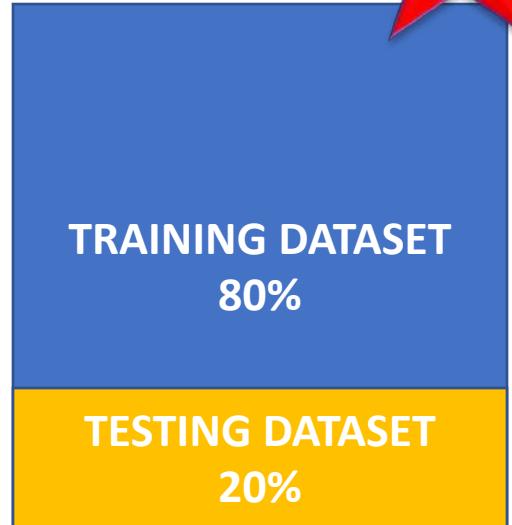
- Learning algorithm takes actions that maximizes some notion of cumulative reward.
- Over time, the network learns to prefer the right kind of action and to avoid the wrong one.



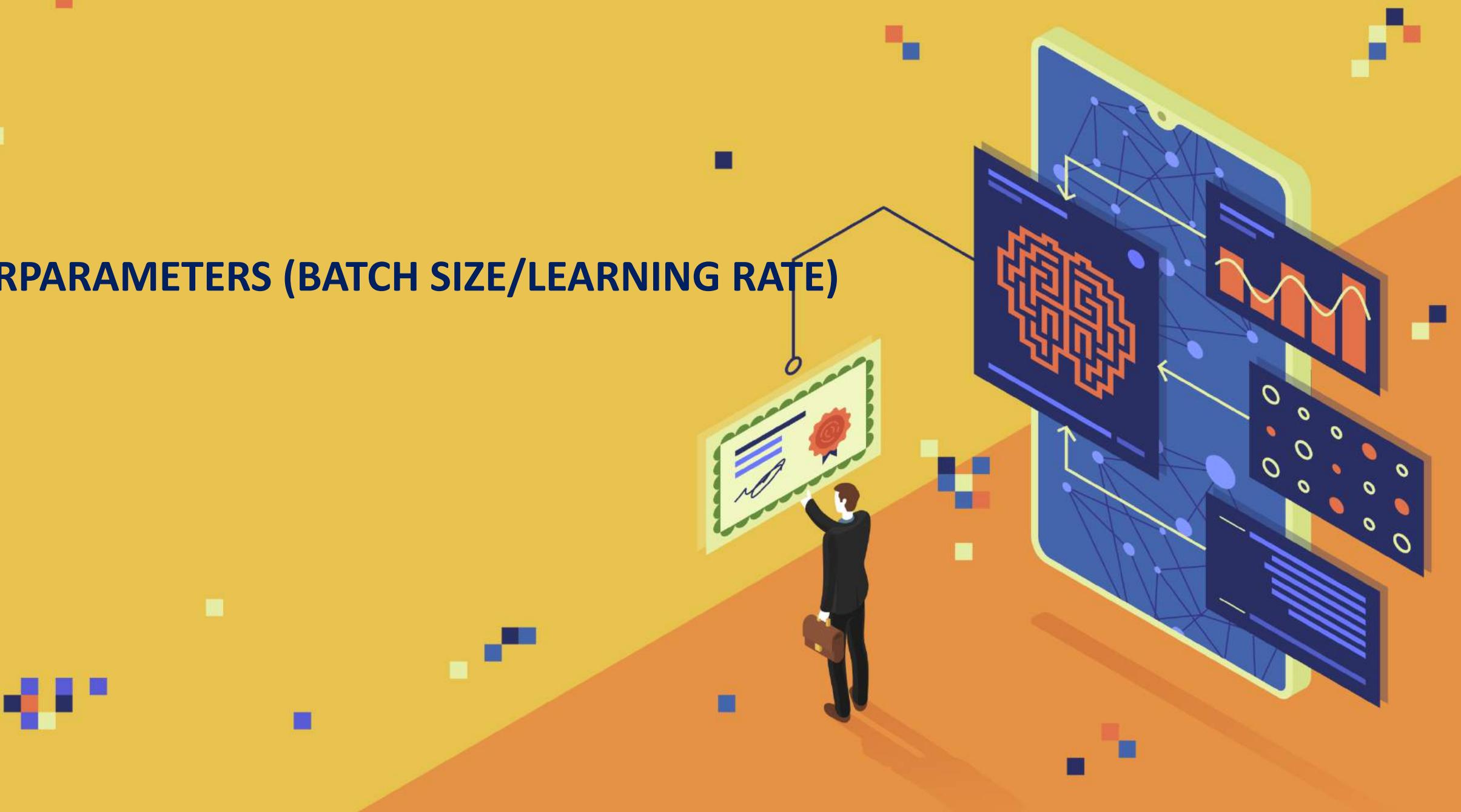
DIVIDE DATA INTO TRAINING AND TESTING



- Data set is generally divided into 80% for training and 20% for testing.
- Sometimes, we might include cross validation dataset as well and then we divide it into 60%, 20%, 20% segments for training, validation, and testing, respectively (numbers may vary).
 1. Training set: used for gradient calculation and weight update.
 2. Validation set:
 - used for cross-validation which is performed to assess training quality as training proceeds.
 - Cross-validation is implemented to overcome over-fitting (over-training). Over-fitting occurs when algorithm focuses on training set details at cost of losing generalization ability.
 - Trained network MSE might be small during training but during testing, the network may exhibit poor generalization performance.
 3. Testing set: used for testing trained network.



HYPERPARAMETERS (BATCH SIZE/LEARNING RATE)



ANN HYPERPARAMETERS TUNING

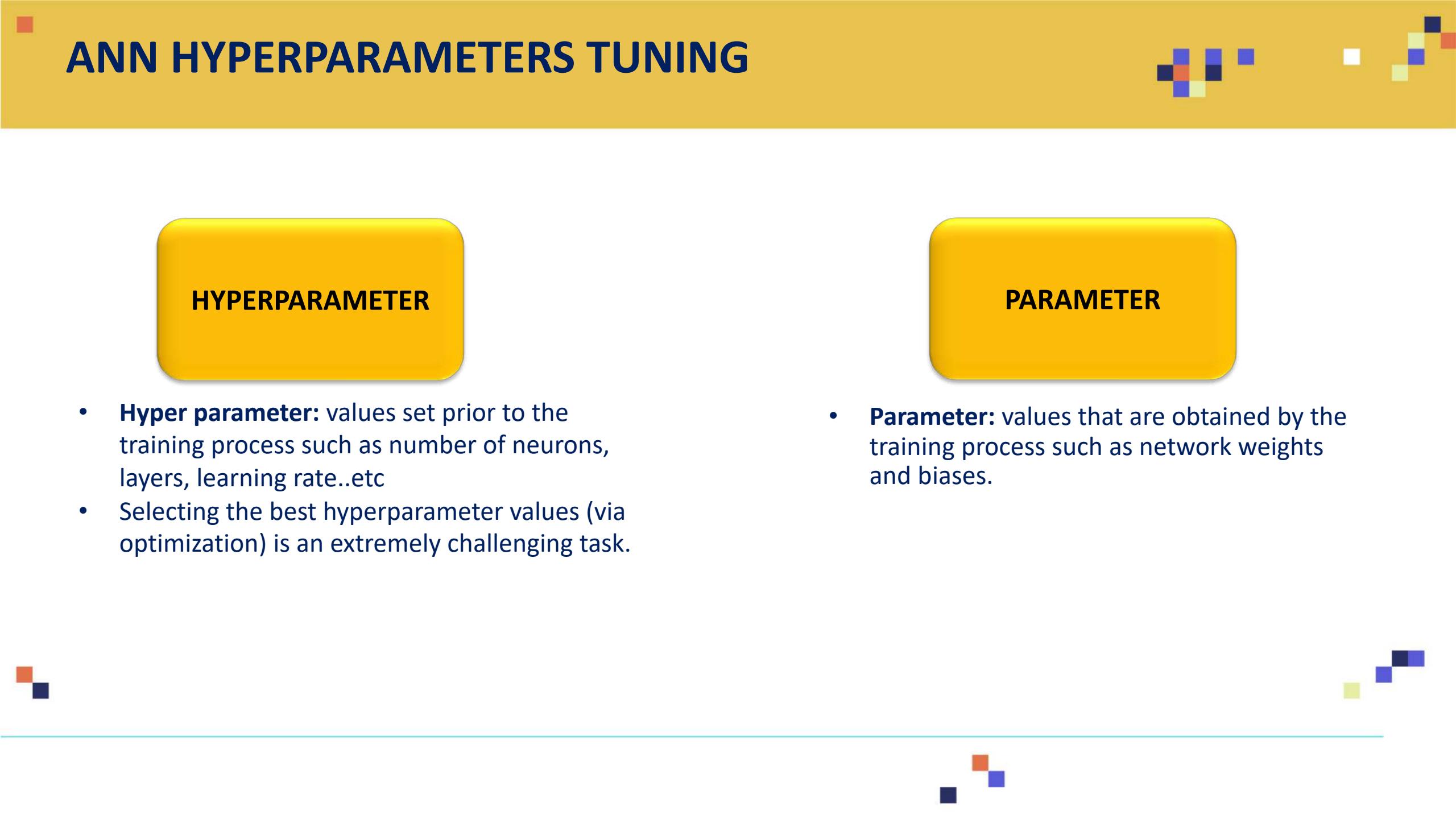


HYPERPARAMETER

- **Hyper parameter:** values set prior to the training process such as number of neurons, layers, learning rate..etc
- Selecting the best hyperparameter values (via optimization) is an extremely challenging task.

PARAMETER

- **Parameter:** values that are obtained by the training process such as network weights and biases.



LEARNING RATE

- ANNs are trained using gradient descent algorithm.
- An important parameter used during training is known as the “learning rate”.
- Learning rate is a hyperparameter that represents the size of the steps taken which indicates how aggressive you’d like to update the parameters.
- **If learning rate increases, the area covered in the search space will increase so we might reach global minimum faster.**
- **However, we can overshoot the target.**
- **For small learning rates, training will take much longer to reach optimized weight values**

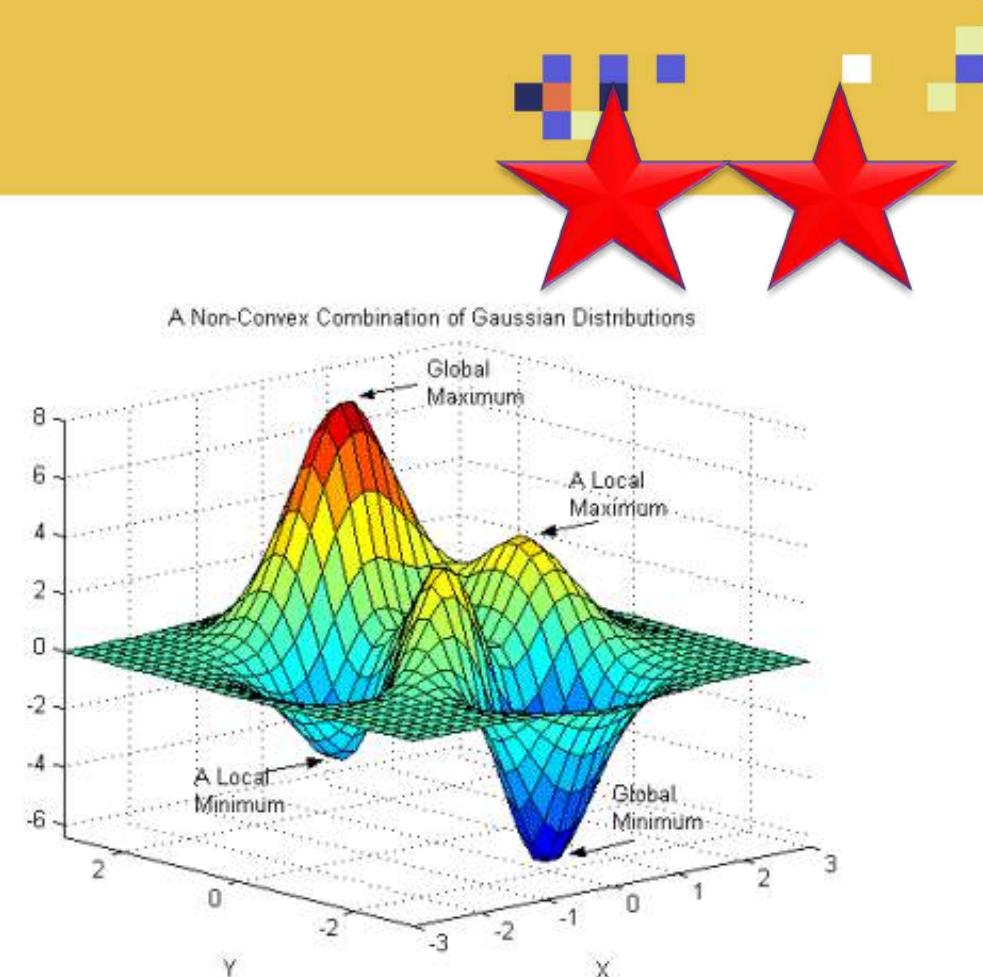
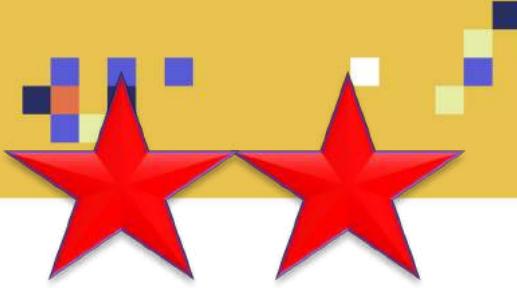
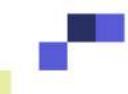


Photo Credit: https://commons.wikimedia.org/wiki/File:Non-Convex_Objective_Function.gif

BATCH SIZE



- Batch size indicates the number of samples that will propagate through the network.
- Example:
 - Let's assume that we have 1000 images for training.
 - For batch size = 50, the first 50 images (from index 1 to index 50) will be propagated to network and used for training.
 - Then the next 50 images are propagated (index 51 to index 100).
 - Procedure is repeated until we use all the training data.
- You can use large or small batch size, in the previous example, you can use batch size = 50 or 1000.
- Note that we can shuffle the training dataset between samples and this will lead to very different results every time we train the network.
- **If the batch size is small, the ANN can easily escape local minimum areas!**
- **If the batch size is large, the ANN can get stuck in a local minimum.**

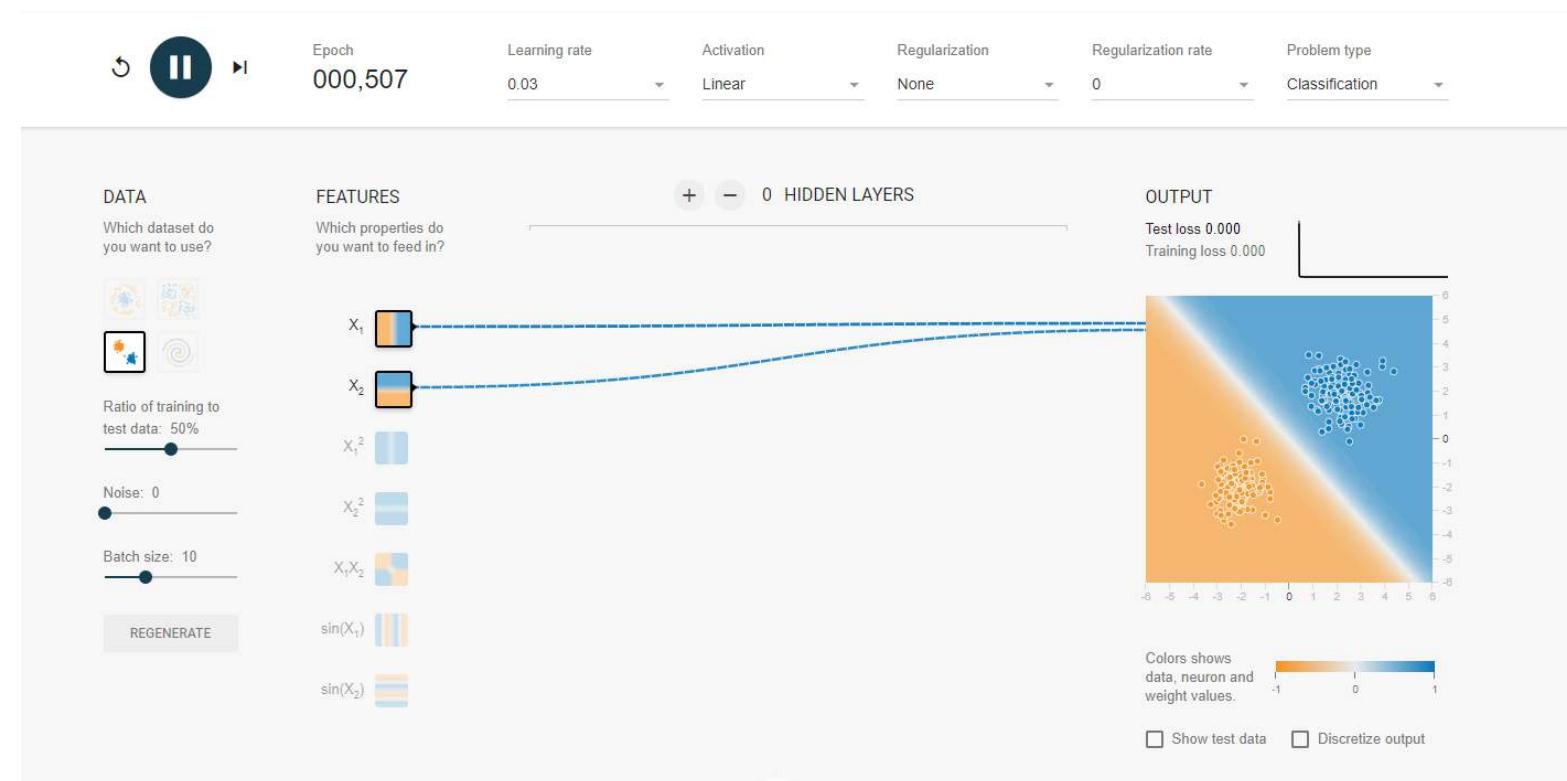


TENSORFLOW PLAYGROUND



DEEP DIVE INTO TF PLAYGROUND

- Check this out: <https://playground.tensorflow.org>



GRADIENT DESCENT AND BACK PROPAGATION



GRADIENT DESCENT

- Gradient descent is an optimization algorithm used to obtain the optimized network weight and bias values
- It works by iteratively trying to minimize the cost function
- It works by calculating the gradient of the cost function and moving in the negative direction until the local/global minimum is achieved
- If the positive of the gradient is taken, local/global maximum is achieved

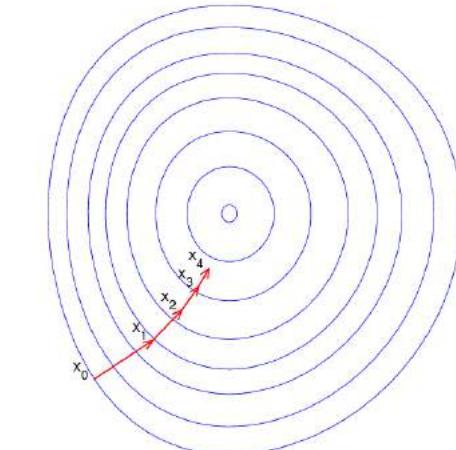
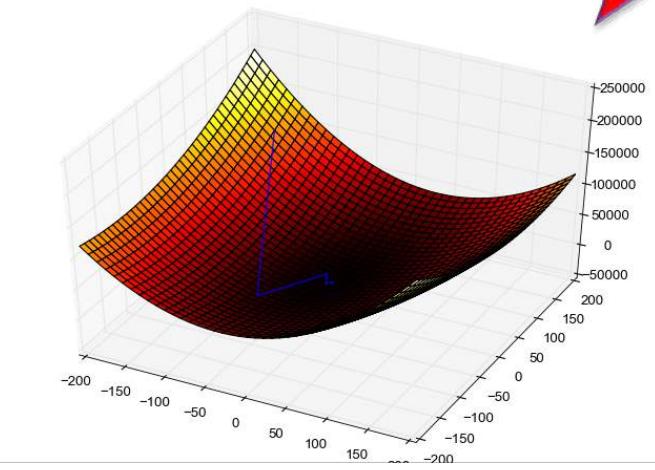


Photo Credit: https://commons.wikimedia.org/wiki/File:Gradient_descent_method.png

Photo Credit: https://commons.wikimedia.org/wiki/File:Gradient_descent.png

GRADIENT DESCENT

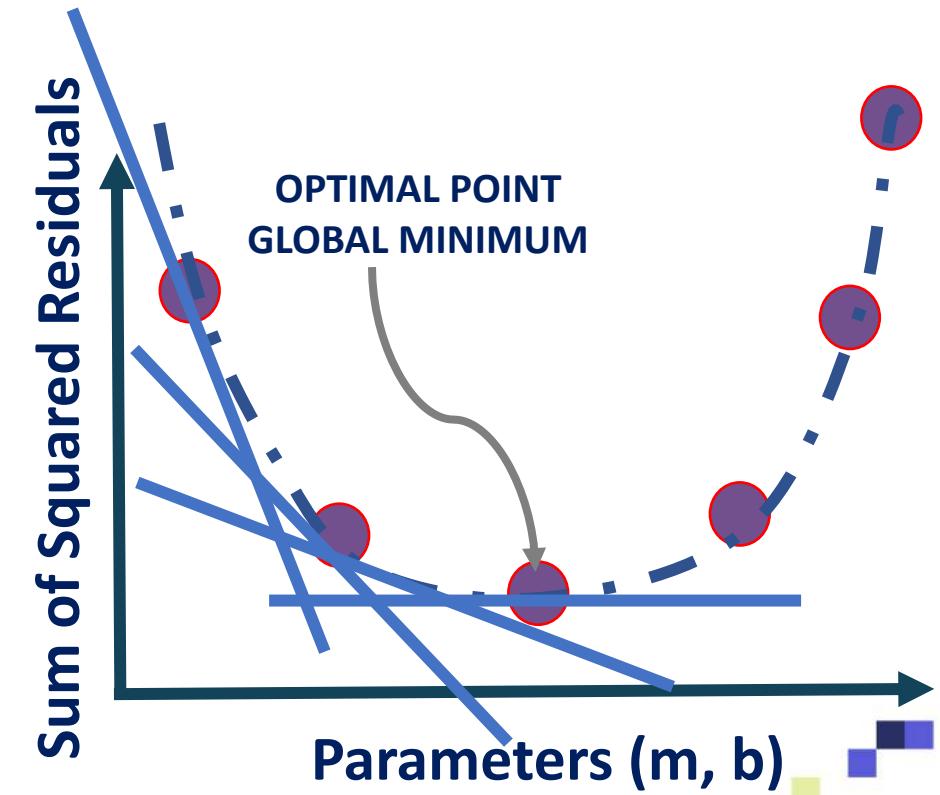
Gradient descent works as follows:

1. Calculate the derivative (gradient) of the Loss function
2. Pick random values for parameters m, b and substitute
3. Calculate the step size (how much are we going to update the parameters?)
$$\text{Step size} = \text{Slope} * \text{learning rate}$$
4. Update the parameters and repeat

$$y = b + m * x$$

GOAL IS TO FIND
BEST PARAMETERS

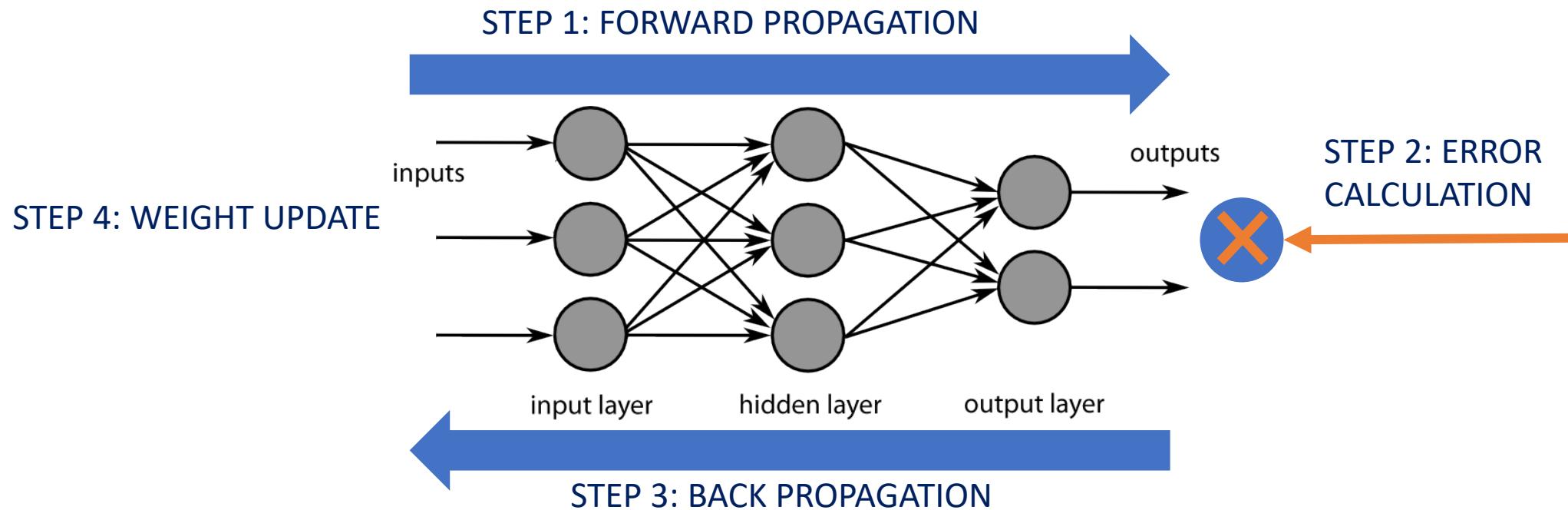
*Note: in reality, this graph is 3D and has three axes, one for m, b and sum of squared residuals



BACK PROPAGATION



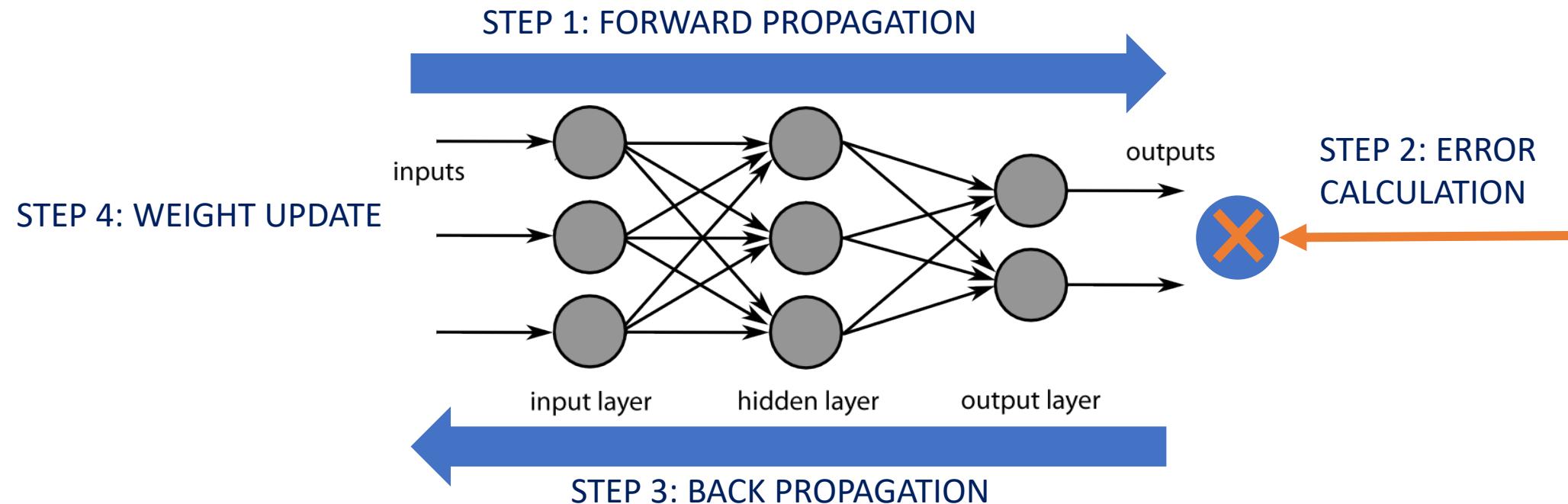
- Backpropagation is a method used to train ANNs by calculating gradient needed to update network weights.
- It is commonly used by the gradient descent optimization algorithm to adjust the weight of neurons by calculating the gradient of the loss function.



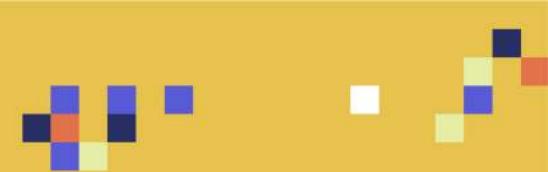
BACK PROPAGATION



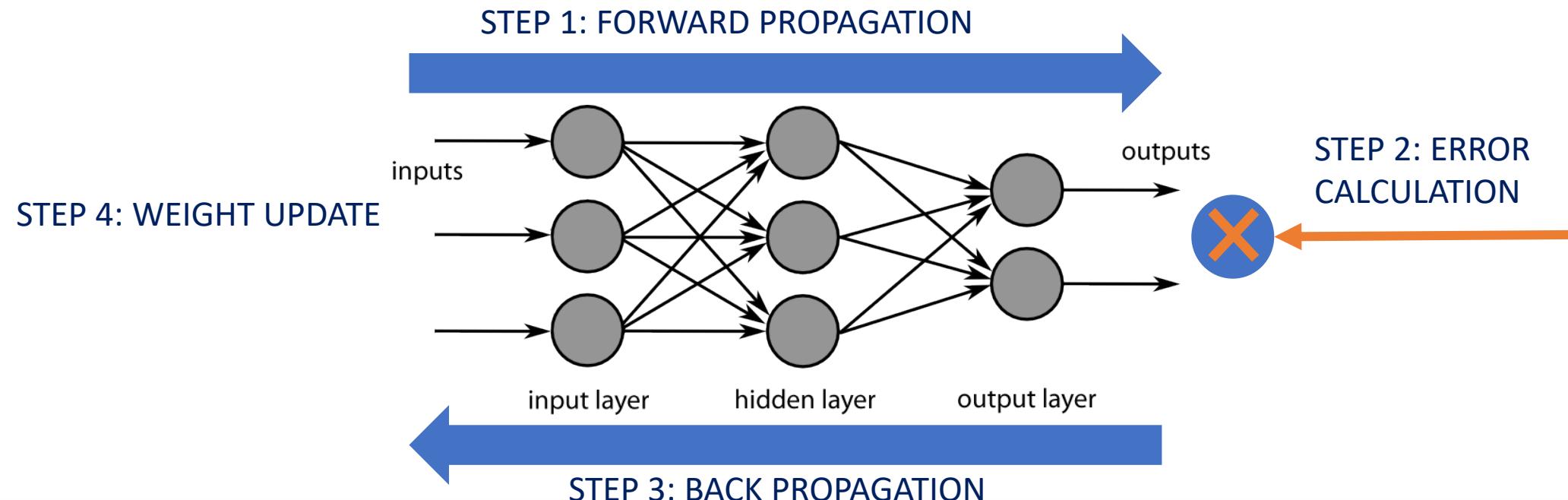
- Backpropagation Phase 1: propagation
 - Propagation forward through the network to generate the output value(s)
 - Calculation of the cost (error term)
 - Propagation of output activations back through network using training pattern target in order to generate the deltas (difference between targeted and actual output values)



BACK PROPAGATION



- Phase 2: weight update
 - Calculate weight gradient.
 - A ratio (percentage) of the weight's gradient is subtracted from the weight.
 - This ratio influences the speed and quality of learning and called learning rate. The greater the ratio, the faster neuron train, but lower ratio, more accurate the training is.



OVERFITTING Vs. UNDERFITTING MODELS

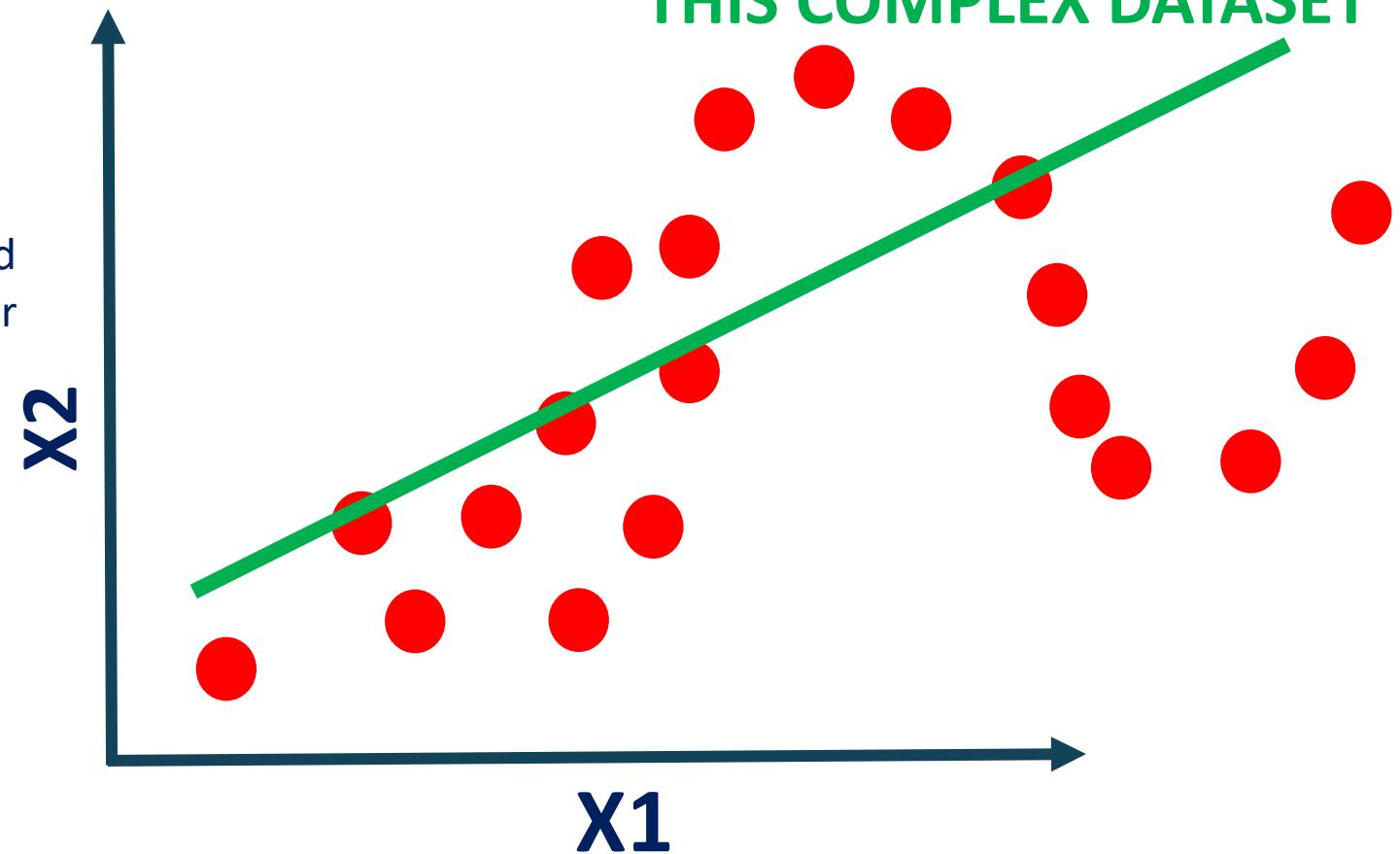


UNDERFITTING MODEL



- Model is under fitting if it's too simple that it cannot reflect the complexity of the training dataset.
- We can overcome under fitting by:
 - increasing the complexity of the model.
 - Training the model for a longer period of time (more epochs) to reduce error

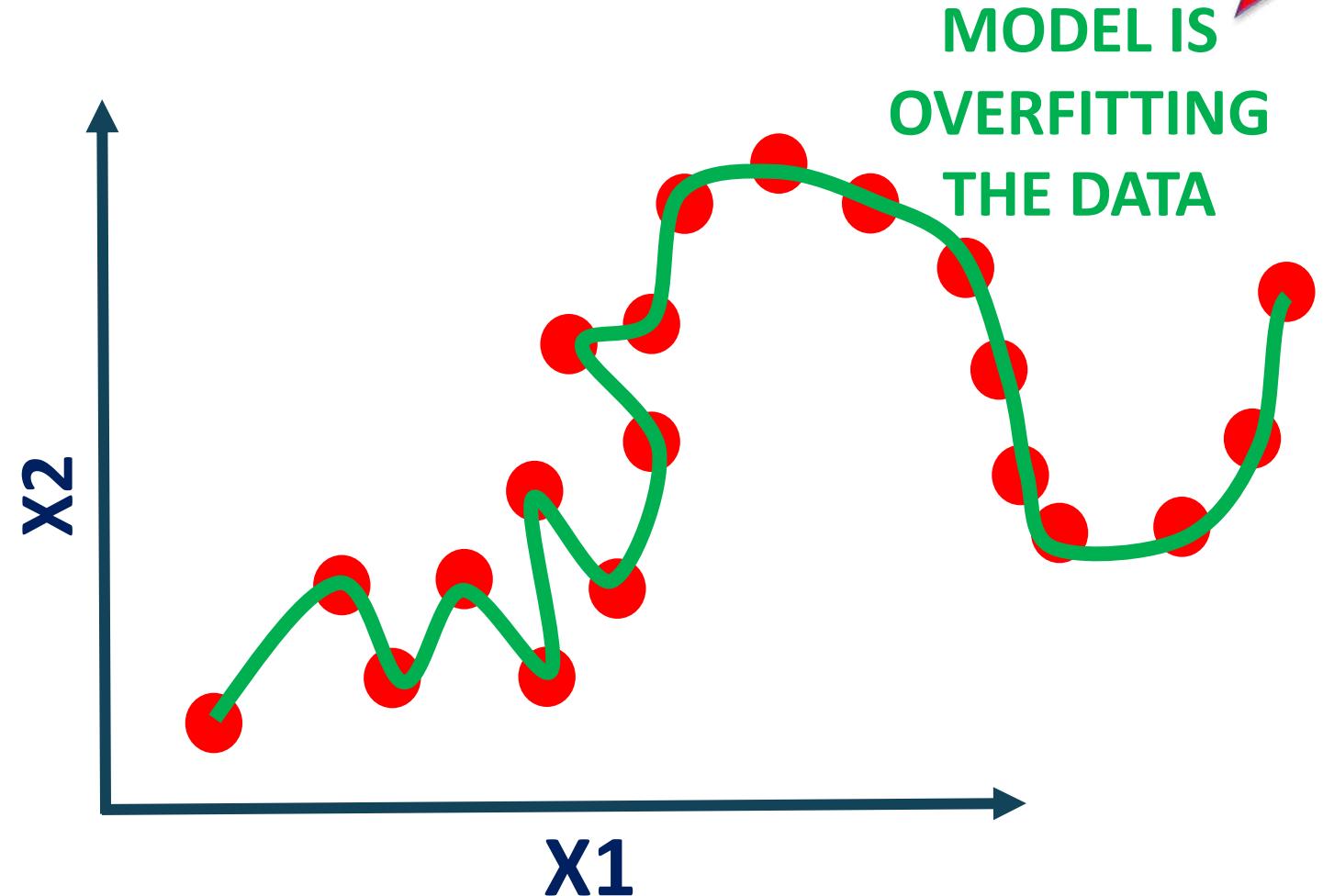
**MODEL IS TOO SIMPLE FOR
THIS COMPLEX DATASET**



OVERFITTING MODEL



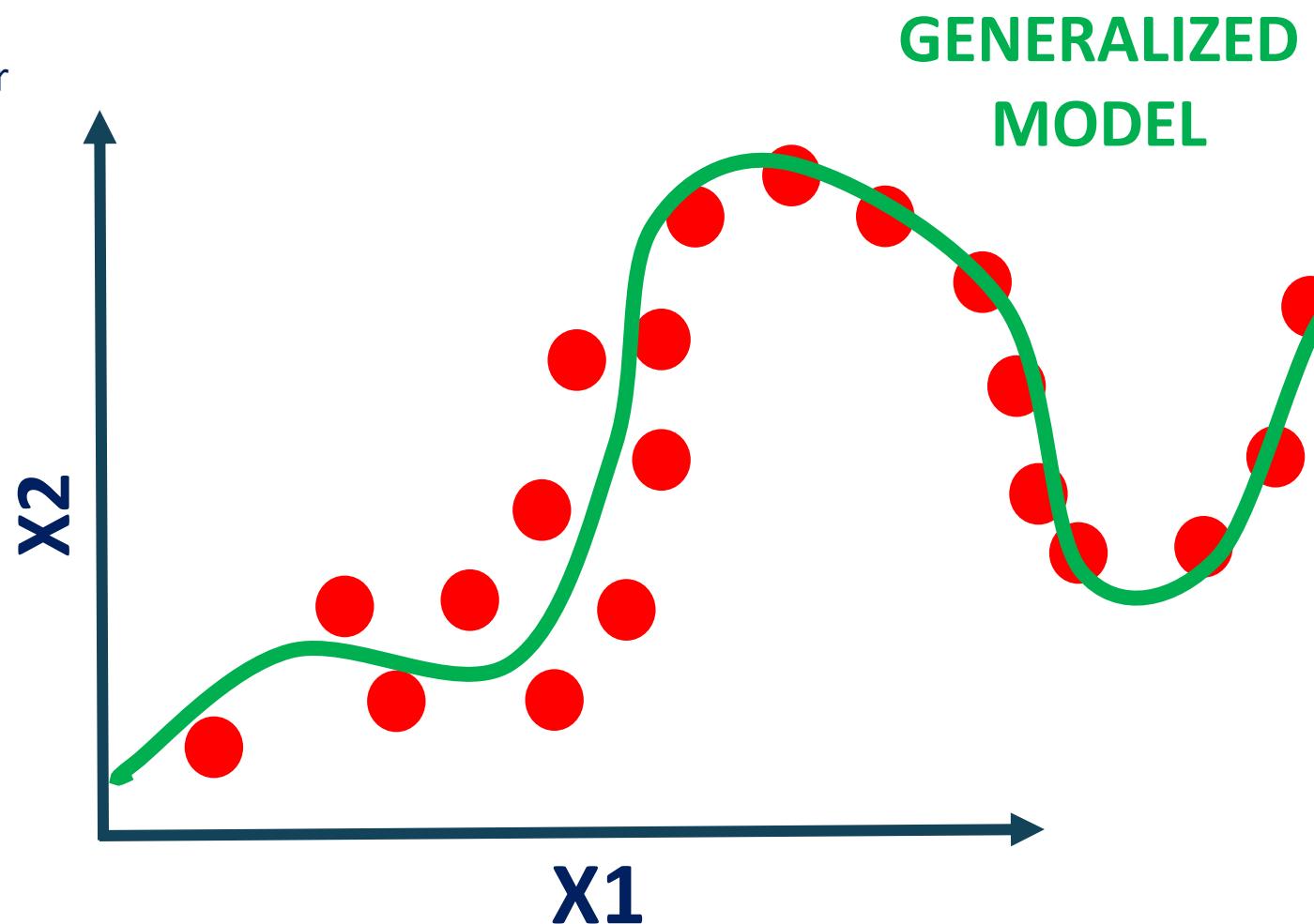
- Model is overfitting data when it memorizes all the specific details of the training data and fails to generalize.
- Overfitting models tend to perform very well on the training dataset but poorly on any new dataset (testing dataset)
- Machine learning is the art of creating models that are able to generalize and avoid memorization.



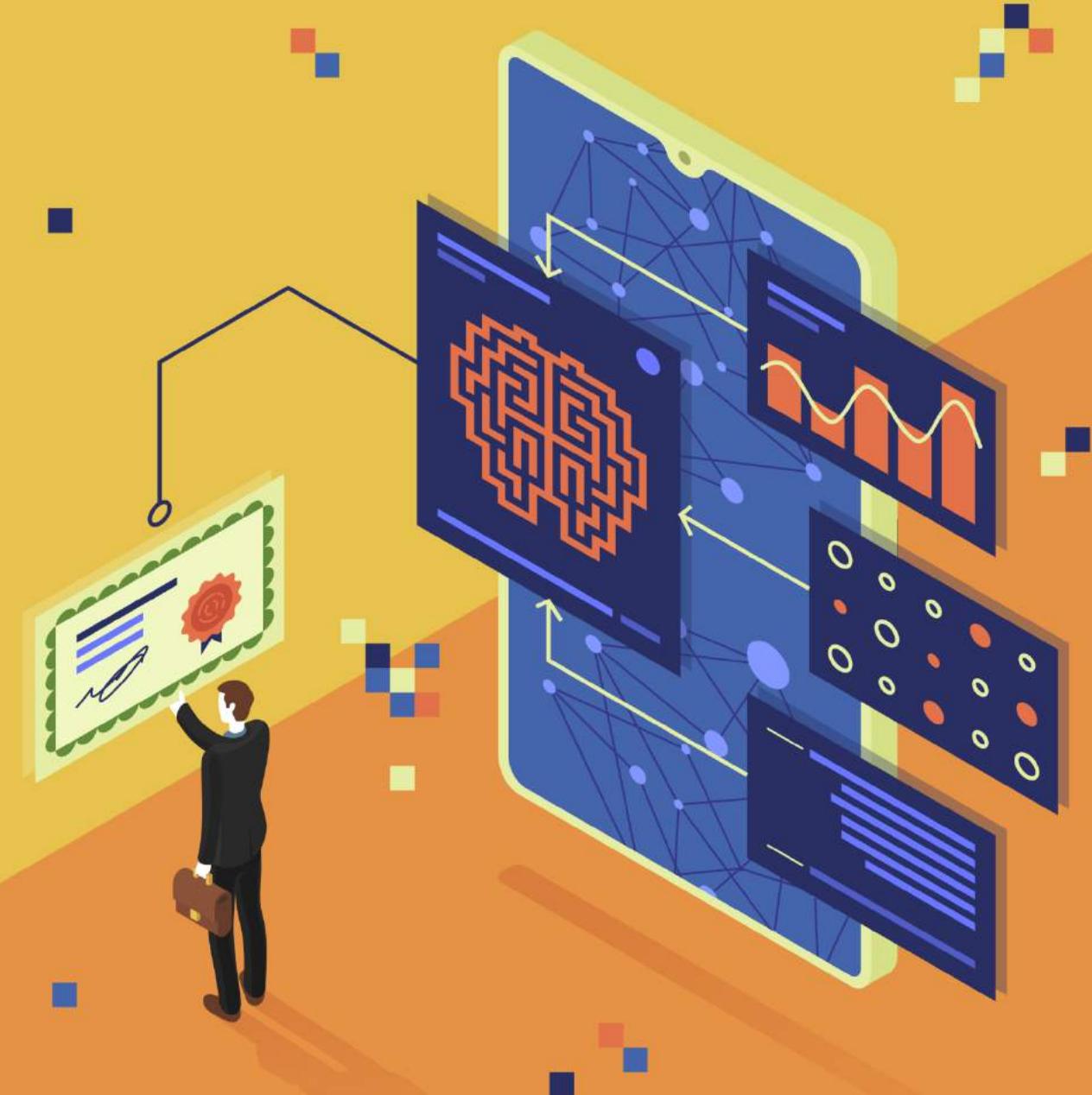
BEST MODEL (GENERALIZED)



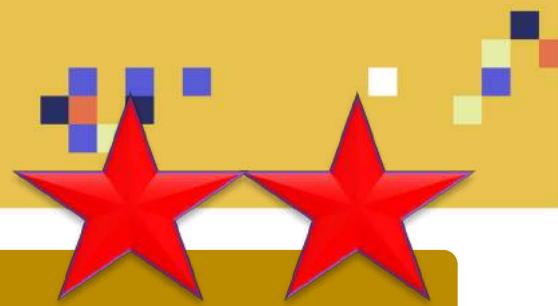
- Model that performs well during training and testing (on new dataset that has never seen before) is considered the best model (goal).



HOW TO OVERCOME OVERFITTING?



HOW TO OVERCOME OVERFITTING?



PERFORM EARLY STOPPING

- Stop training when you notice that the validation loss increases while training loss decreases.

DO REGULARIZATION

- Regularization improves the model generalization capability.

ADD MORE DATASET

- By increasing the size of the dataset, the model might generalize more.

PERFORM FEATURE SELECTION

- Dropping useless features could improve the model generalization capability.

USE BOOSTING AND BAGGING (ENSEMBLE LEARNING)

- By combining voting from many different models via bagging and boosting, this will improve model generalization

ADD MORE NOISE

- Adding noise might enable model to become more general

USE DROPOUT TECHNIQUE

- In Artificial Neural Network training, dropping some neurons using Dropout technique improves networks generalization ability.

HOW TO AVOID OVERFITTING?



1. Early Stopping:

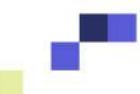
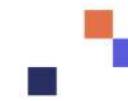
- In order to avoid overfitting during ANN training, early stopping technique could be used.
- This is used to when the accuracy over the training dataset is increased but the accuracy over the validation dataset starts to decrease.
- Early stopping ensures that the network is able to generalize. Meaning it will perform great over training and testing dataset.

2. Dropout Regularization

- Another technique that could be used to avoid overfitting is known as dropout regularization. Dropout regularization forces the learning to be spread out amongst the artificial neurons, further preventing overfitting.

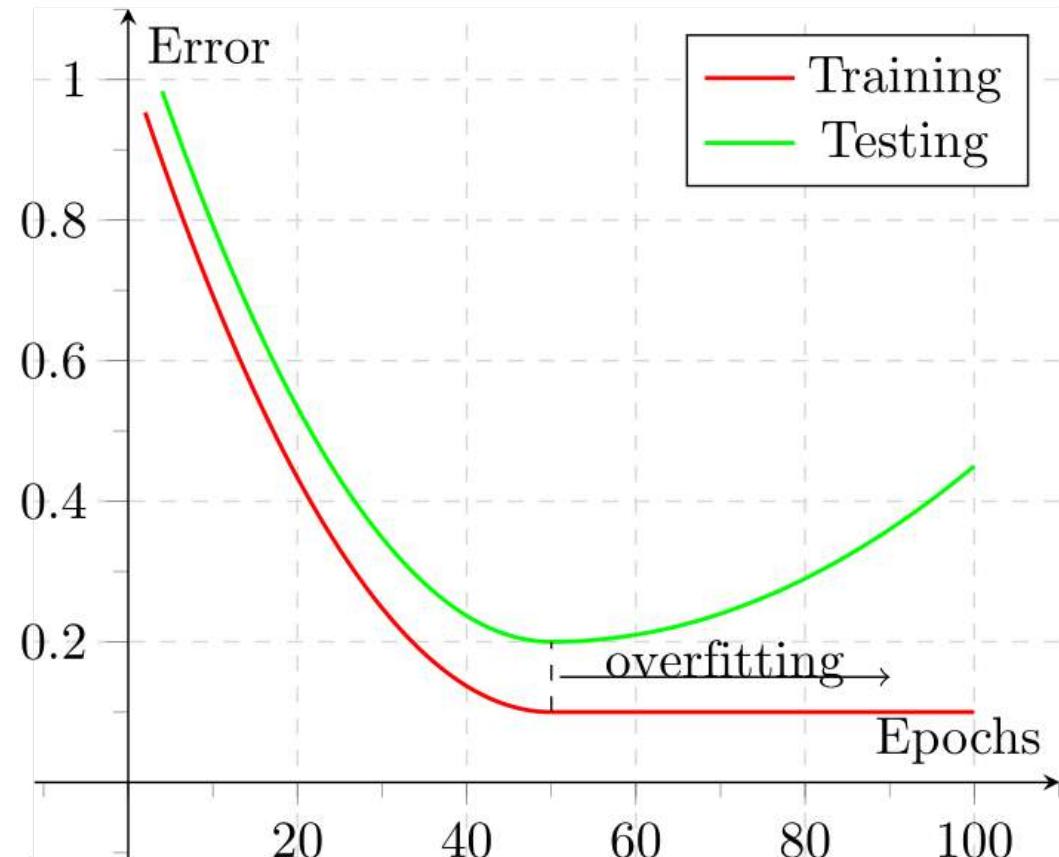
3. Remove ANNs Layers:

- Another technique is to reduce the complexity of the ANN by removing layers, rather than adding them. This might also help prevent an overly complex model from being created.



1. EARLY STOPPING

- Early stopping can be used when the accuracy over the training dataset is increased but the accuracy over the validation (evaluation) dataset starts to decrease.

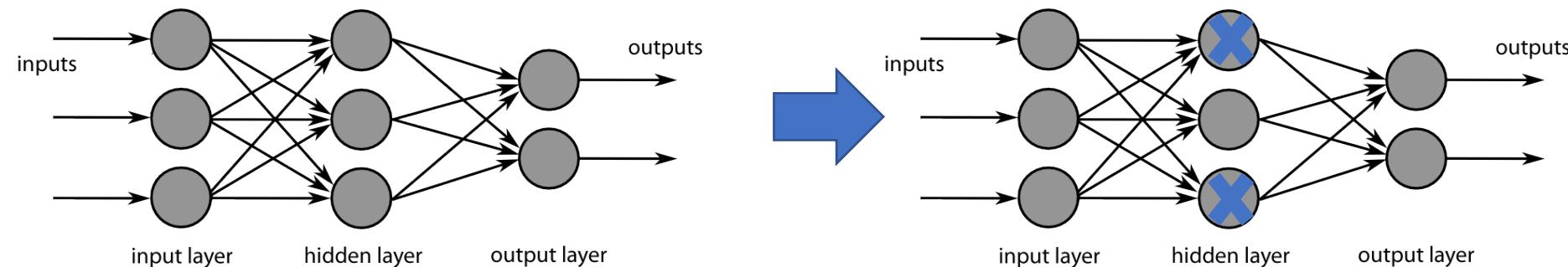


• Photo Credit: <https://commons.wikimedia.org/wiki/File:2d-epochs-overfitting.svg>

2. DROPOUT



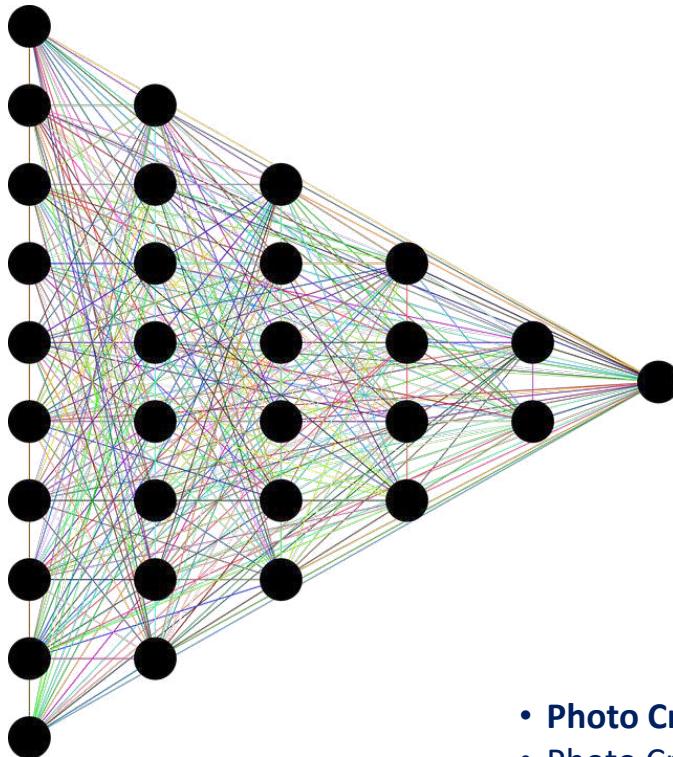
- Improve accuracy by adding a dropout.
- Dropout refers to dropping out units in a neural network.
- Neurons develop co-dependency amongst each other during training
- Dropout is a regularization technique for reducing overfitting in neural networks.
- It enables training to occur on several architectures of the neural network



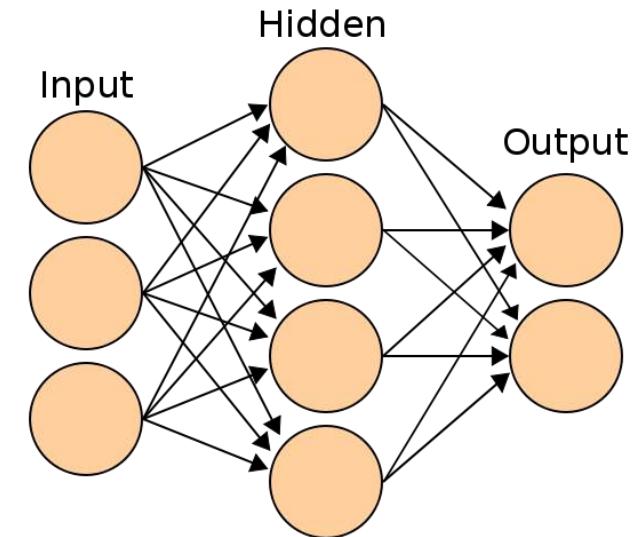
3. REMOVE ANNS LAYERS OR NEURONS (REDUCE MODEL COMPLEXITY)



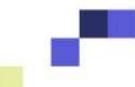
COMPLEX ANN WITH MANY LAYERS AND NEURONS



SIMPLE ANN WITH FEW LAYERS AND NEURONS



- Photo Credit: <https://pixabay.com/vectors/neural-network-thought-mind-mental-3816319/>
- Photo Credit: https://commons.wikimedia.org/wiki/File:Artificial_neural_network.svg



BIAS VARIANCE TRADE-OFF



BIAS VARIANCE INTUITION

- Let's assume that we want to get the relationship between the Temperature and Bike rental usage.
- As temperature experience increase, the bike rental usage tend to increase as well.
- As temperature goes beyond a certain limit, usage tend to plateau and it does not increase anymore.

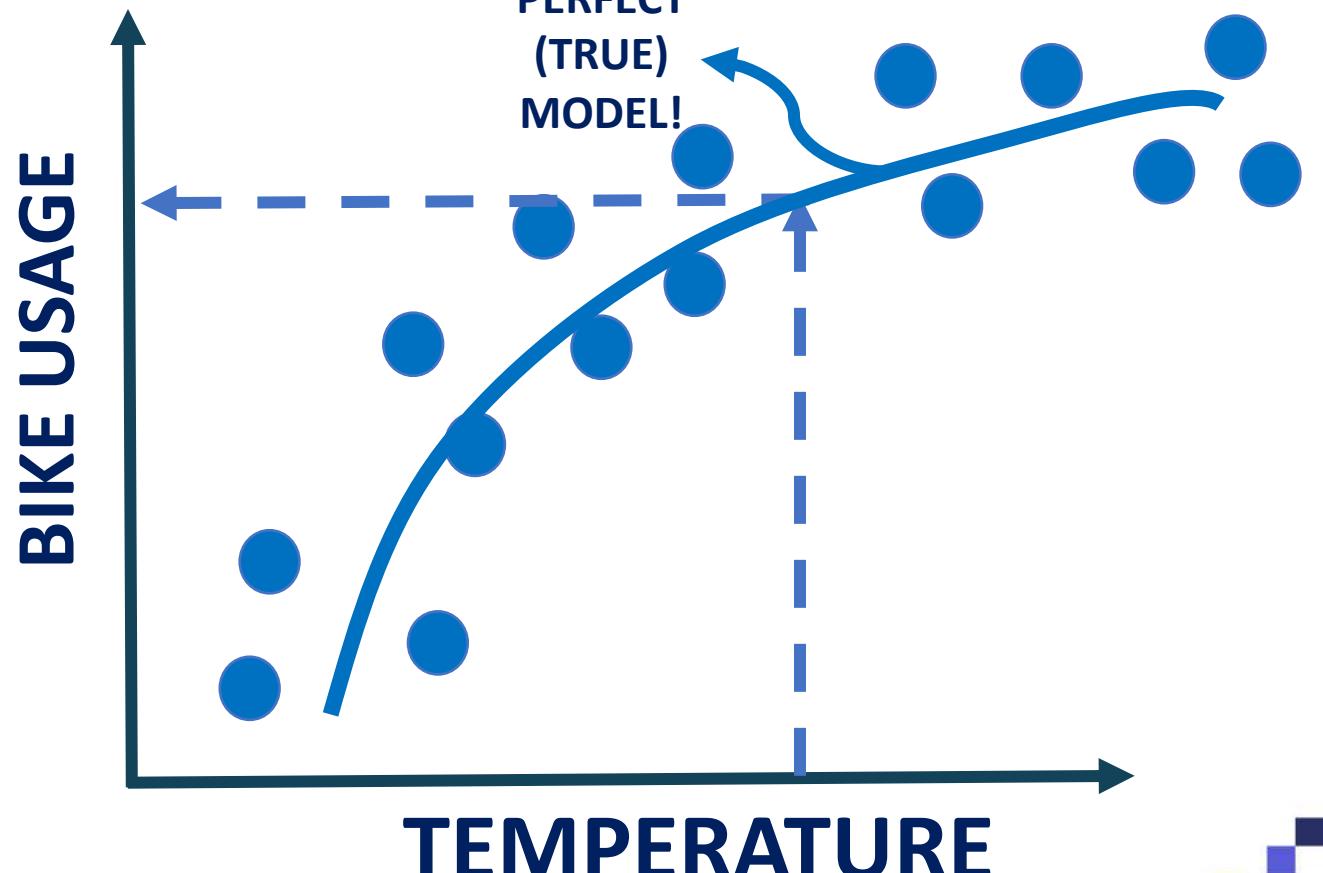
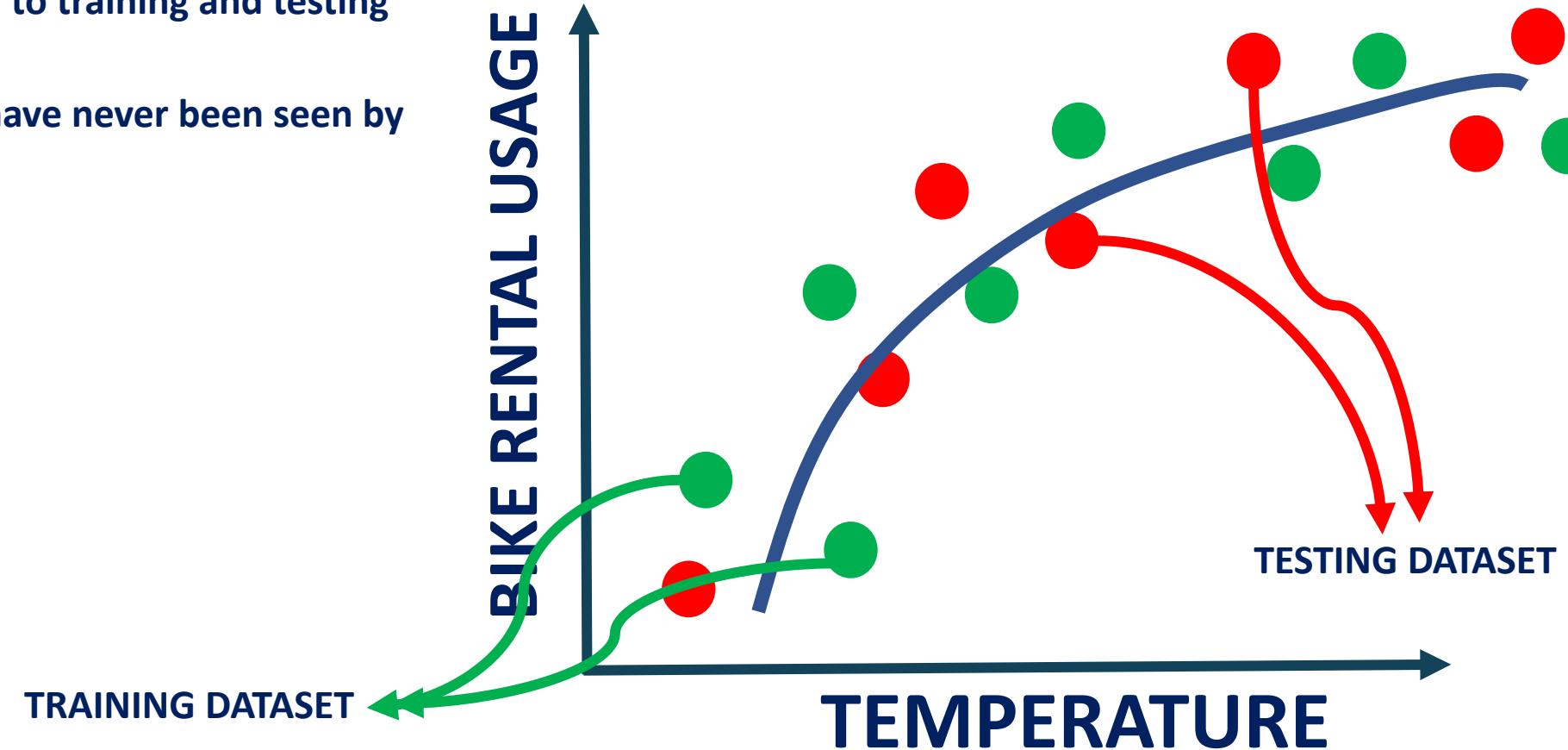


Image Source: <https://pixabay.com/photos/bike-rental-bikes-rent-pay-2284380/>

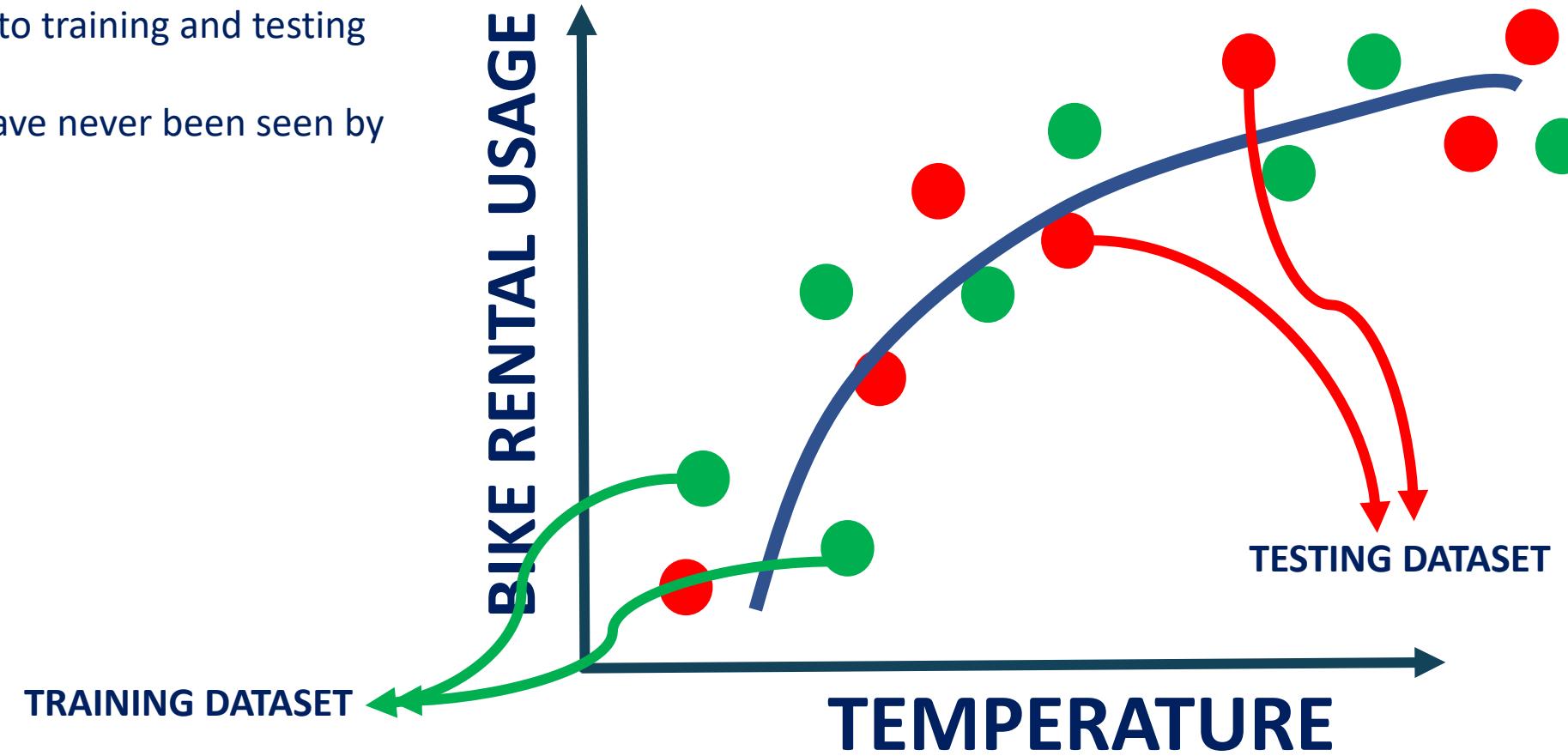
BIAS VARIANCE TRAINING VS. TESTING DATASETS

- Dataset is divided to training and testing datasets
- Testing datasets have never been seen by the model before



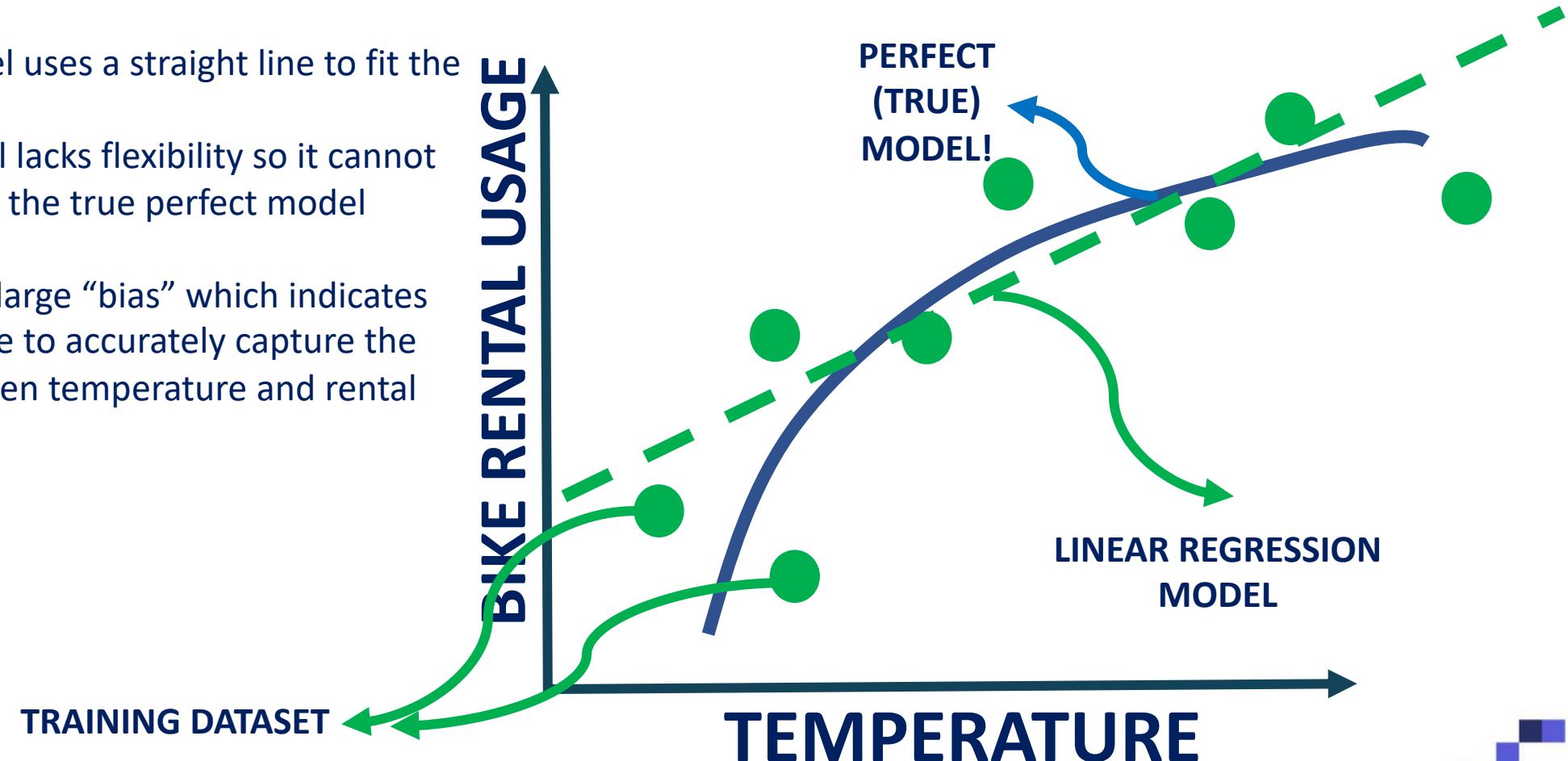
BIAS AND VARIANCE: TRAINING VS. TESTING DATASETS

- Dataset is divided to training and testing datasets
- Testing datasets have never been seen by the model before



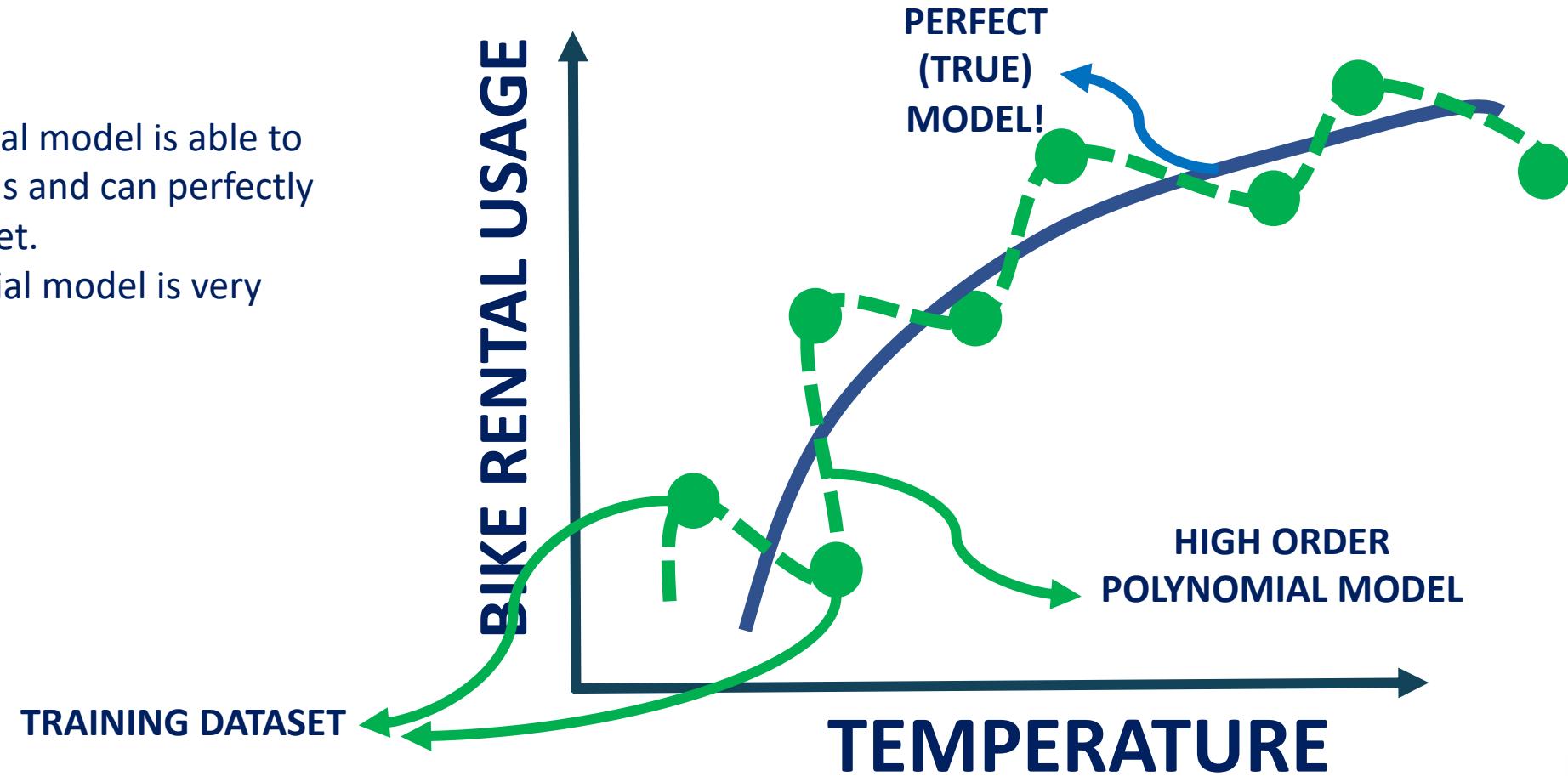
BIAS AND VARIANCE: MODEL #1– LINEAR REGRESSION (SIMPLE)

- Linear Regression model uses a straight line to fit the training dataset
- Linear regression model lacks flexibility so it cannot properly fit the data (as the true perfect model does!)
- The linear model has a large “bias” which indicates that the model is unable to accurately capture the true relationship between temperature and rental usage.

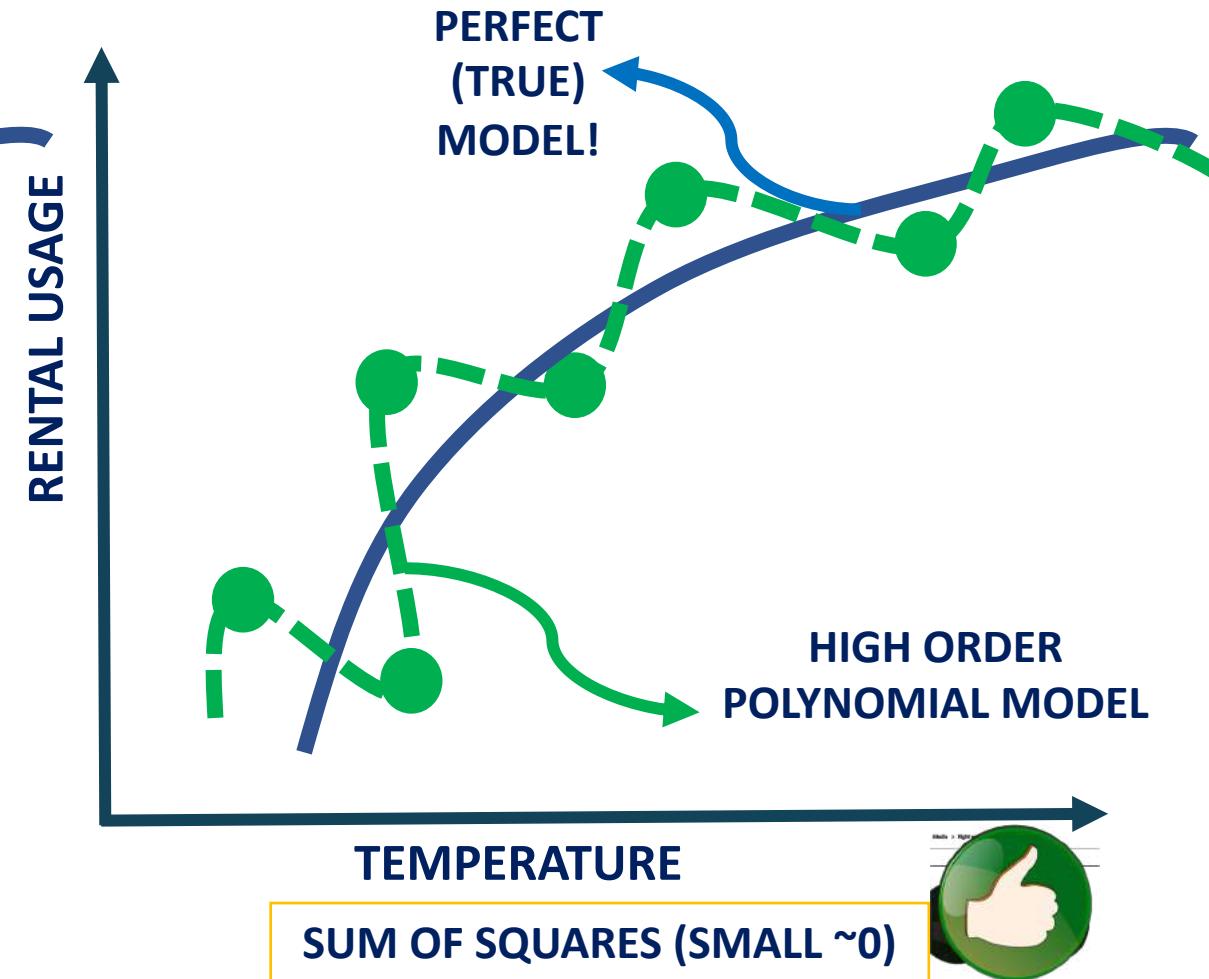
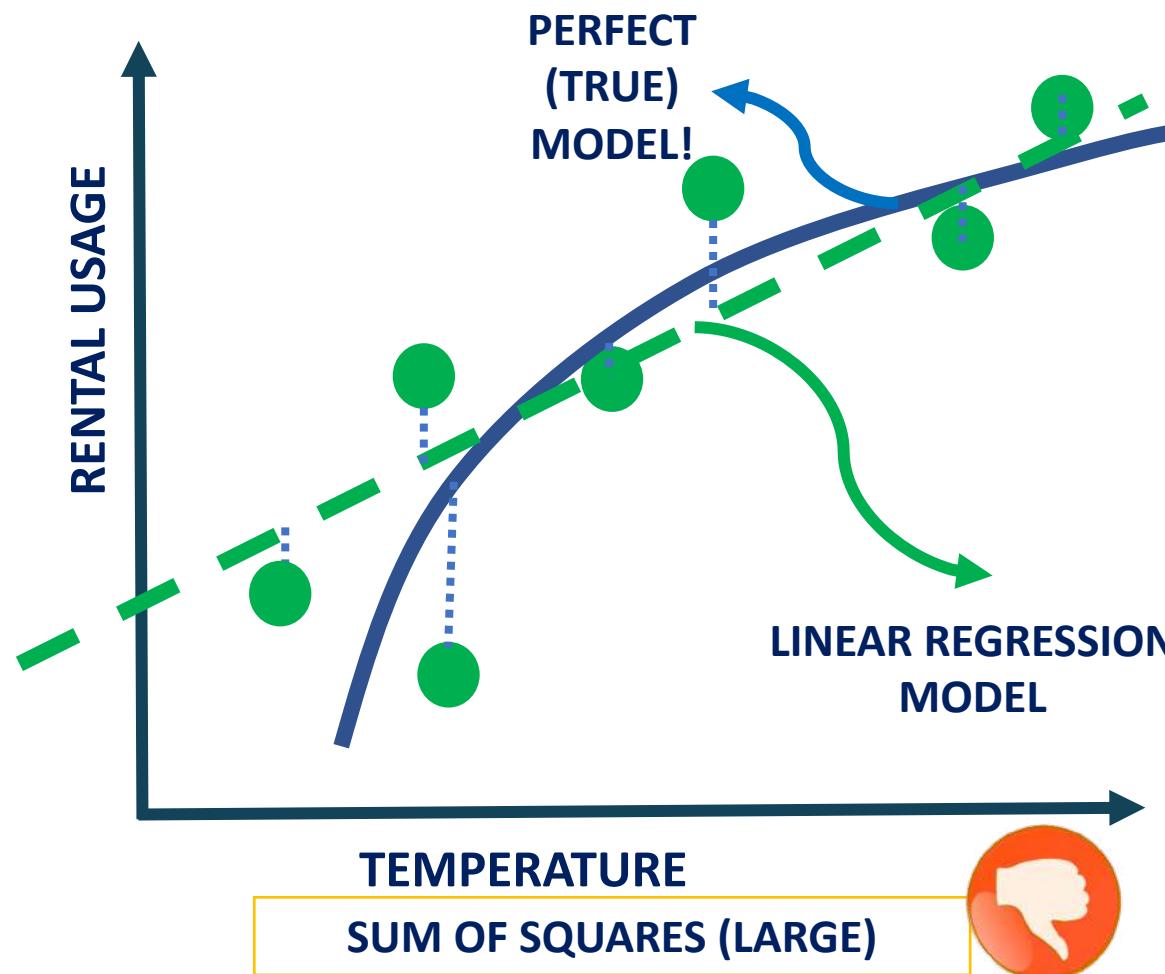


BIAS AND VARIANCE: MODEL #2 – HIGH ORDER POLYNOMIAL REGRESSION (COMPLEX)

- High order polynomial model is able to have a very small bias and can perfectly fit the training dataset.
- High-order polynomial model is very flexible

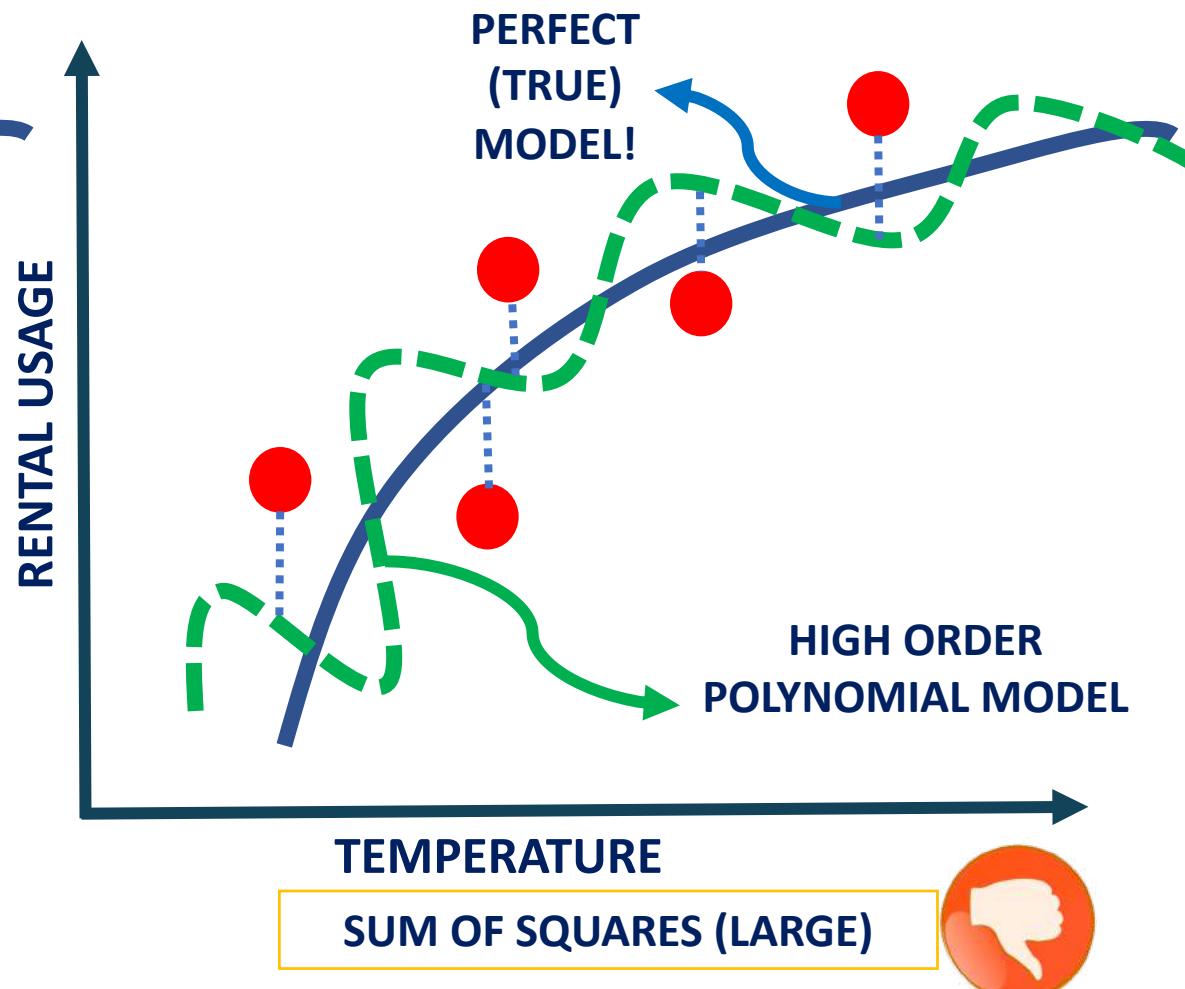
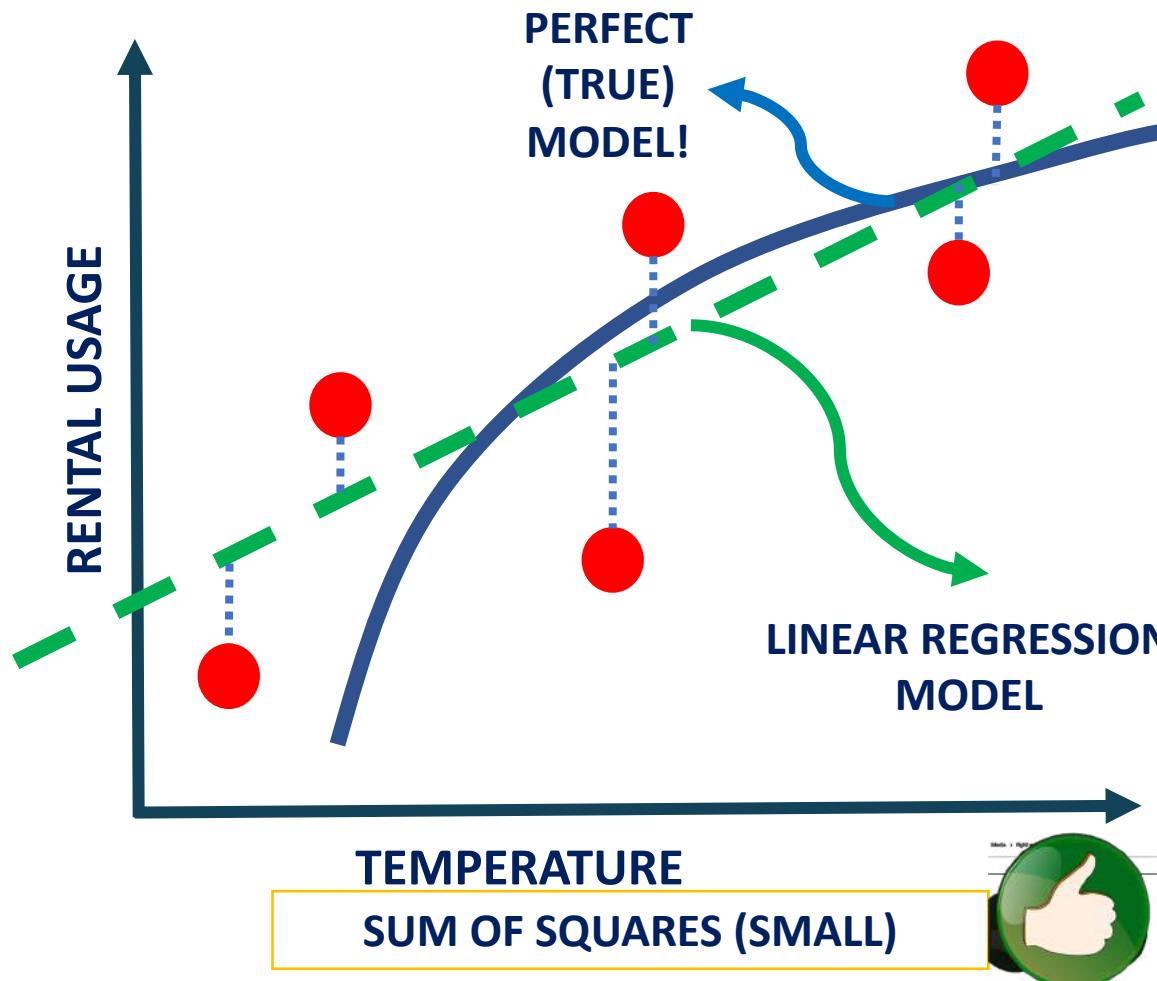


BIAS AND VARIANCE: MODEL #1 Vs. MODEL #2 DURING TRAINING



THIS IS NOT THE WHOLE STORY!!

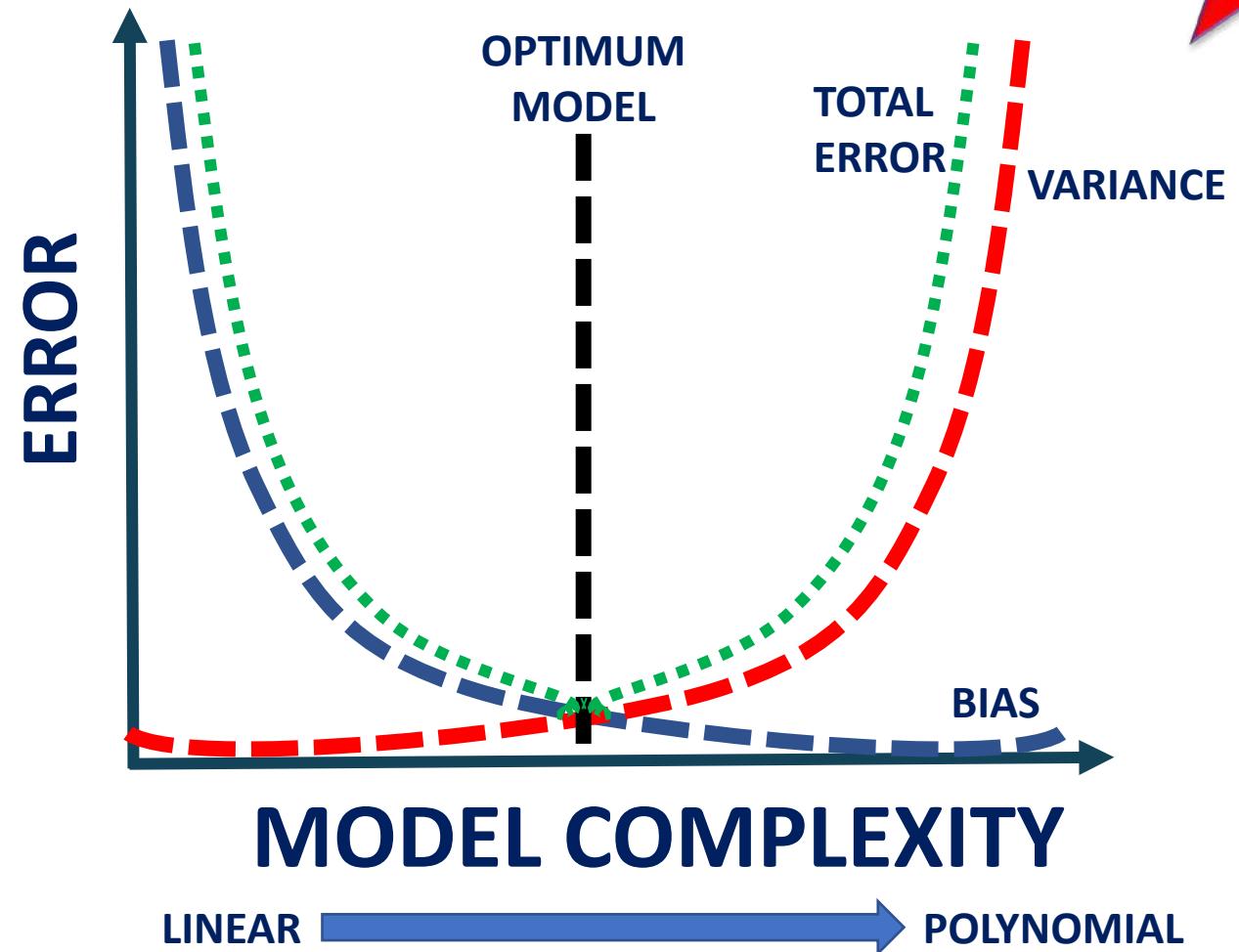
BIAS AND VARIANCE: MODEL #1 Vs. MODEL #2 DURING TESTING



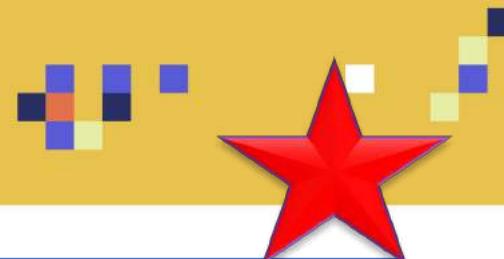
The polynomial model performs poorly on the testing dataset and therefore it has large variance

MODEL COMPLEXITY VS. ERROR

- Regularization works by reducing the variance at the cost of adding some bias to the model.
- A trade-off between variance and bias occurs



MODEL COMPLEXITY VS. ERROR



MODEL #1 (LINEAR REGRESSION) (SIMPLE)	MODEL #2 (HIGH ORDER POLYNOMIAL) (COMPLEX)
Model has High bias because it is very rigid (not flexible) and cannot fit the training dataset well	Model has small bias because it is flexible and can fit the training dataset very well.
Has small variance (variability) because it can fit the training data and the testing data with similar level (the model is able to generalize better) and avoids overfitting	Has large variance (variability) because the model over fitted the training dataset and it performs poorly on the testing dataset
Performance is consistent between the training dataset and the testing dataset	Performance varies greatly between the training dataset and the testing dataset (high variability)
Good generalization	Over fitted

- *Variance measures the difference in fits between the training dataset and the testing dataset*
- *If the model generalizes better, the model has small variance which means the model performance is consistent among the training and testing datasets*
- *If the model over fits the training dataset, the model has large variance*

PERFECT REGRESSION MODEL SHALL HAVE SMALL BIAS AND SMALL VARIABILITY!
A TRADEOFF BETWEEN THE BIAS AND VARIANCE SHALL BE PERFORMED FOR ULTIMATE RESULTS

L2 REGULARIZATION (RIDGE REGRESSION)



REGULARIZATION: INTUITION

- Regularization techniques are used to avoid networks overfitting
- ANN overfitting occurs when the network provide great results on the training data but performs poorly on testing dataset.
- Overfitting occurs when the ANN learns all the patterns of the training dataset but fails to generalize.
- Overfitted ANNs generally provide high accuracy on training dataset but low accuracy on testing and validation (evaluation) datasets

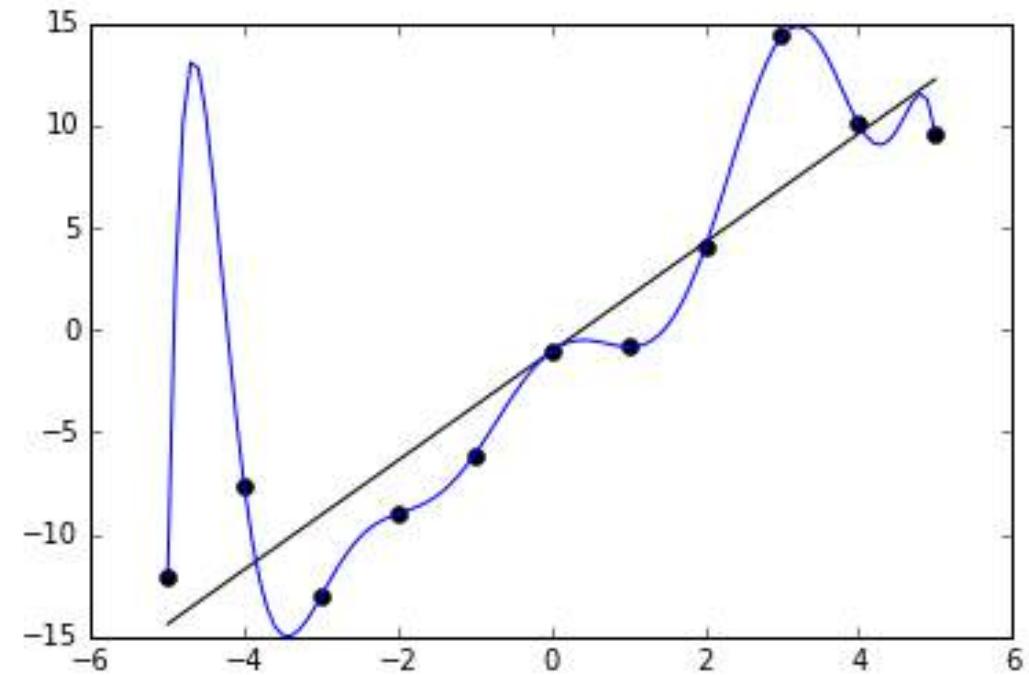
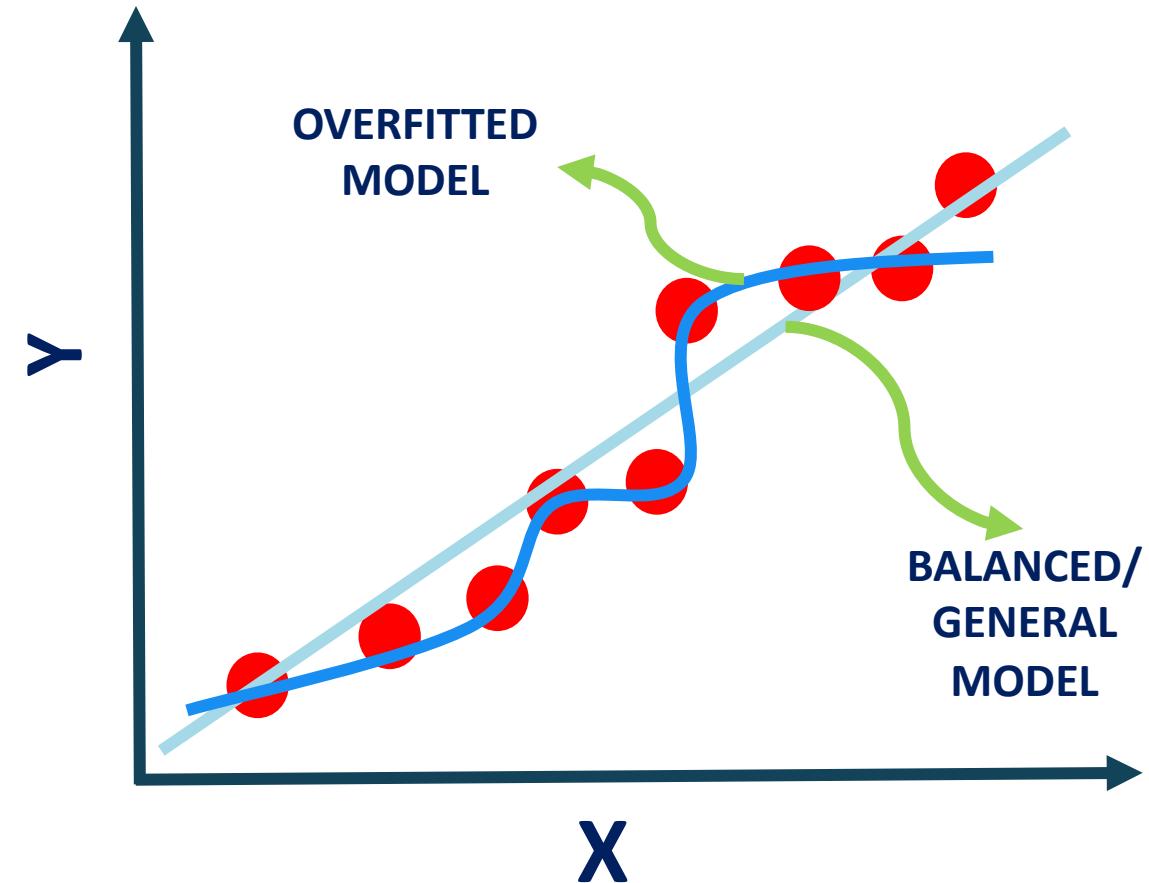


Photo Credit: https://commons.wikimedia.org/wiki/File:Overfitted_Data.png

RIDGE REGRESSION (L2 REGULARIZATION): INTUITION

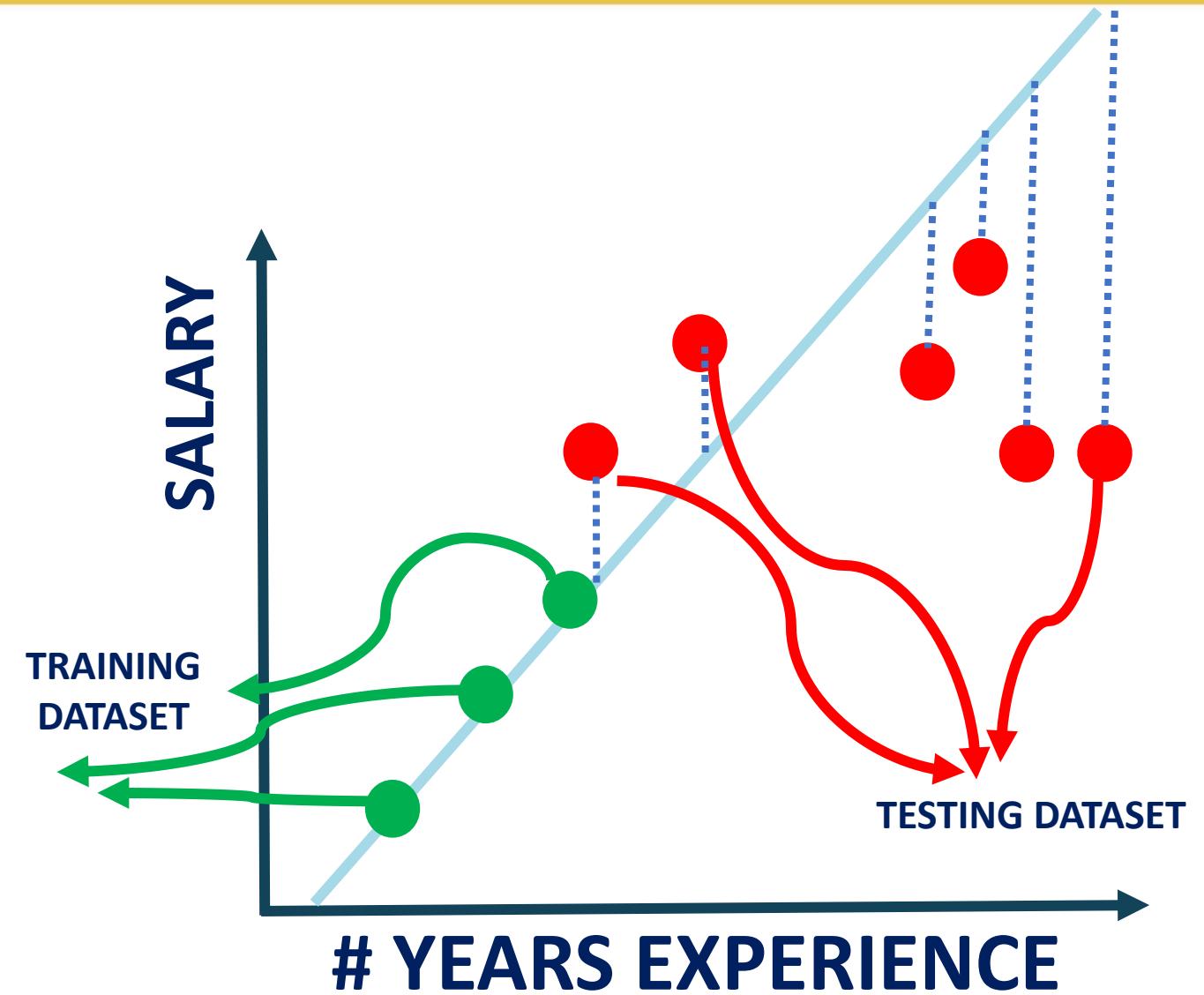
- Ridge regression advantage is to avoid overfitting.
- Our ultimate model is the one that could generalize patterns; i.e.: works best on the training and testing dataset
- Overfitting occurs when the trained model performs well on the training data and performs poorly on the testing datasets
- Ridge regression works by applying a penalizing term (reducing the weights and biases) to overcome overfitting.



RIDGE REGRESSION (L2 REGULARIZATION): INTUITION

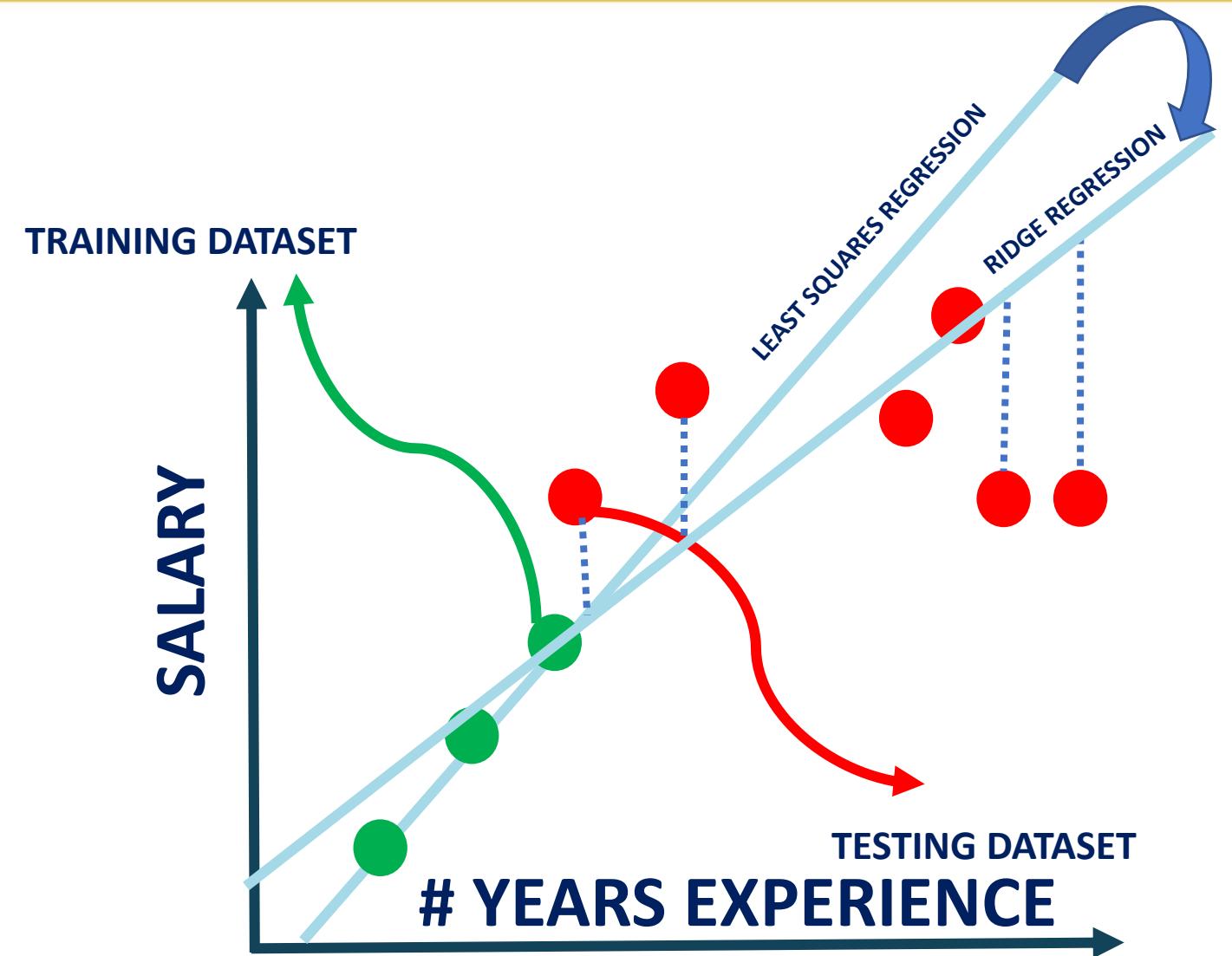


- Least sum of squares is applied to obtain the best fit line
- Since the line passes through the 3 training dataset points, the sum of squared residuals = 0
- However, for the testing dataset, the sum of residuals is large so the line has a high variance.
- Variance means that there is a difference in fit (or variability) between the training dataset and the testing dataset.
- This regression model is overfitting the training dataset



RIDGE REGRESSION (L2 REGULARIZATION): INTUITION

- Ridge regression works by attempting at increasing the bias to improve variance (generalization capability)
- This works by changing the slope of the line
- The model performance might be little poor on the training set but it will perform consistently well on both the training and testing datasets.



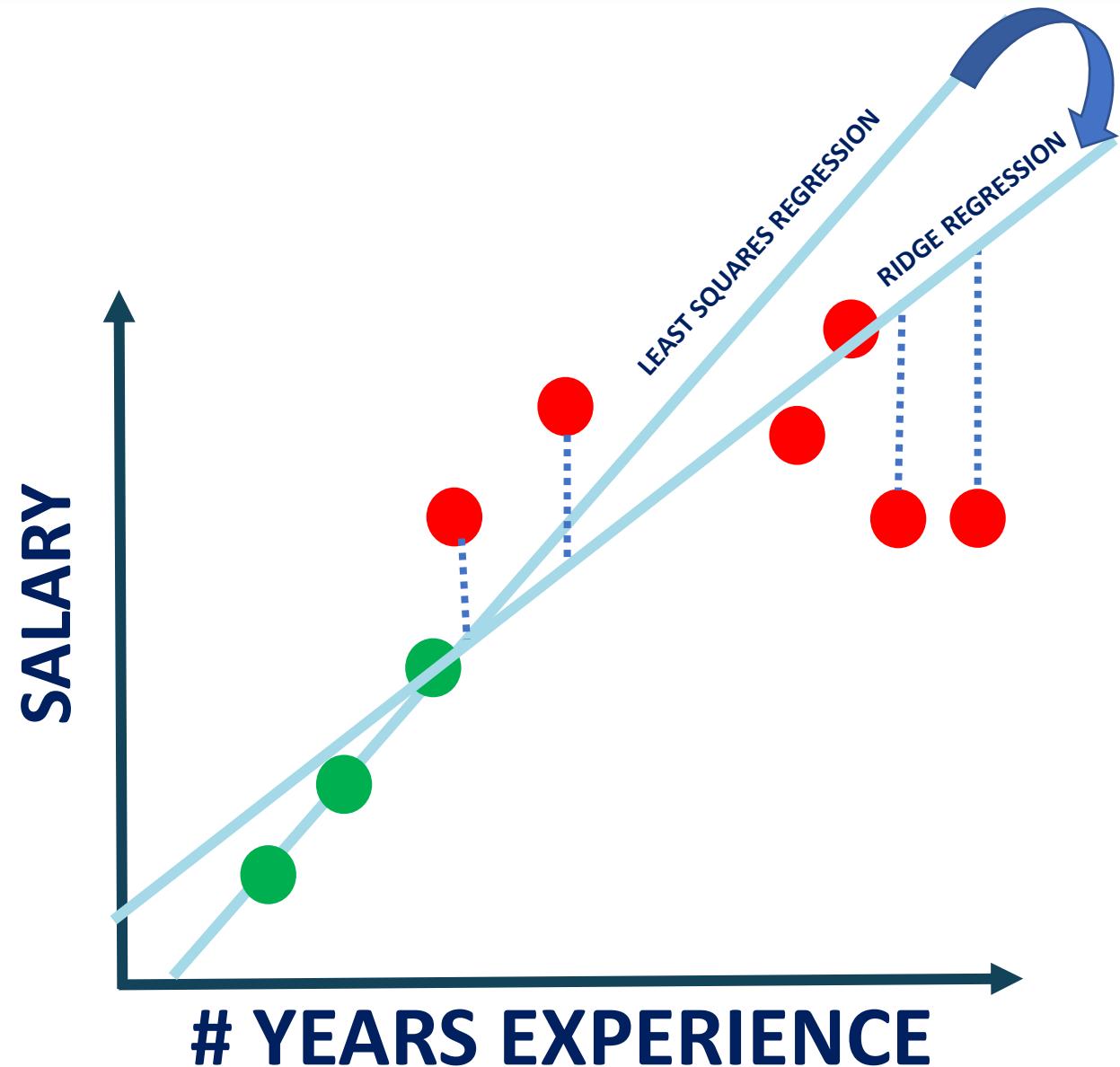
RIDGE REGRESSION (L2 REGULARIZATION): MATH

- Slope has been reduced with ridge regression penalty and therefore the model becomes less sensitive to changes in the independent variable (#Years of experience)

PENALTY TERM

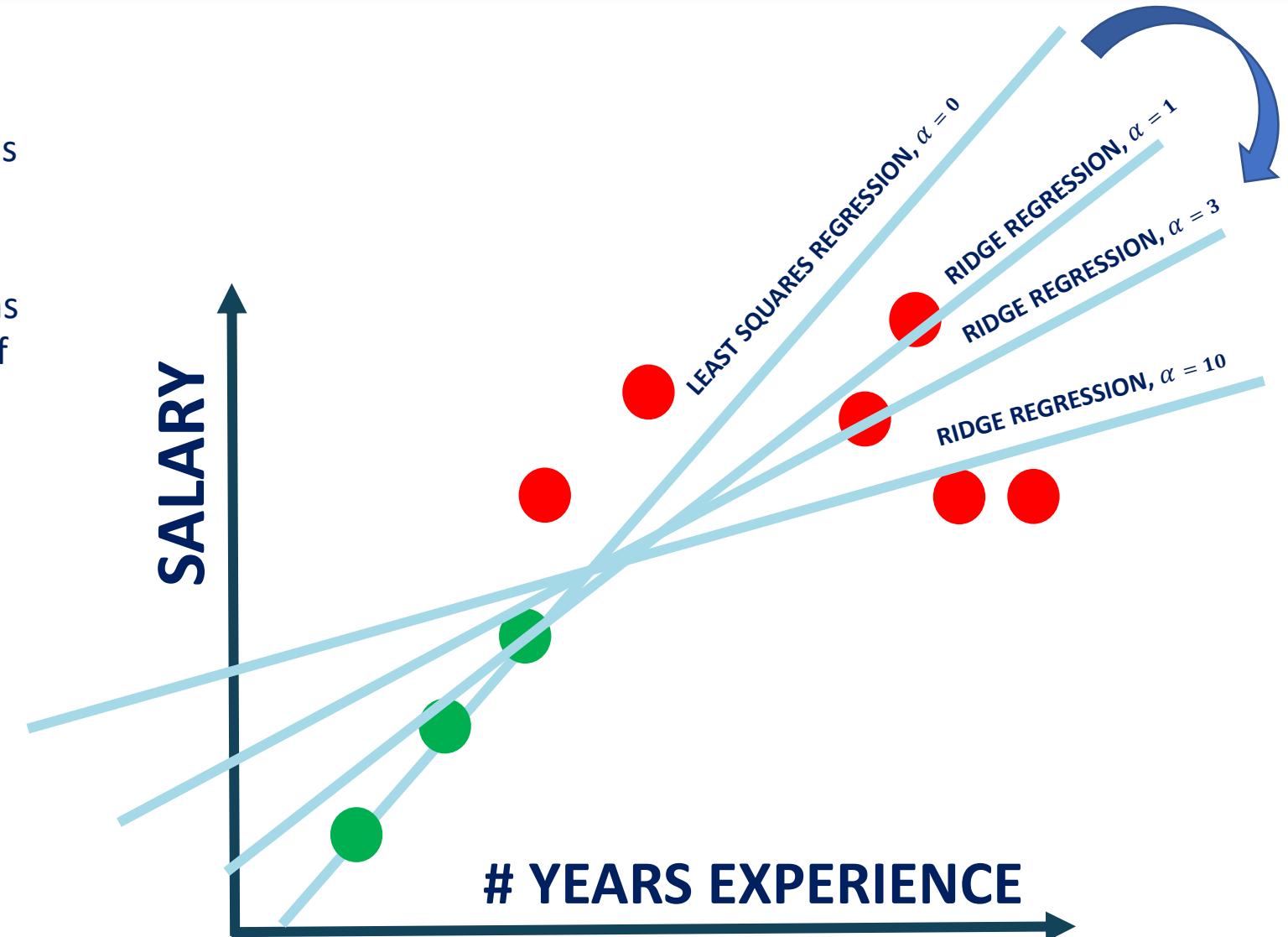
Least Squares Regression:
 $\text{Min}(\text{sum of the squared residuals})$

Ridge Regression:
 $\text{Min}(\text{sum of squared residuals} + \alpha * \text{slope}^2)$

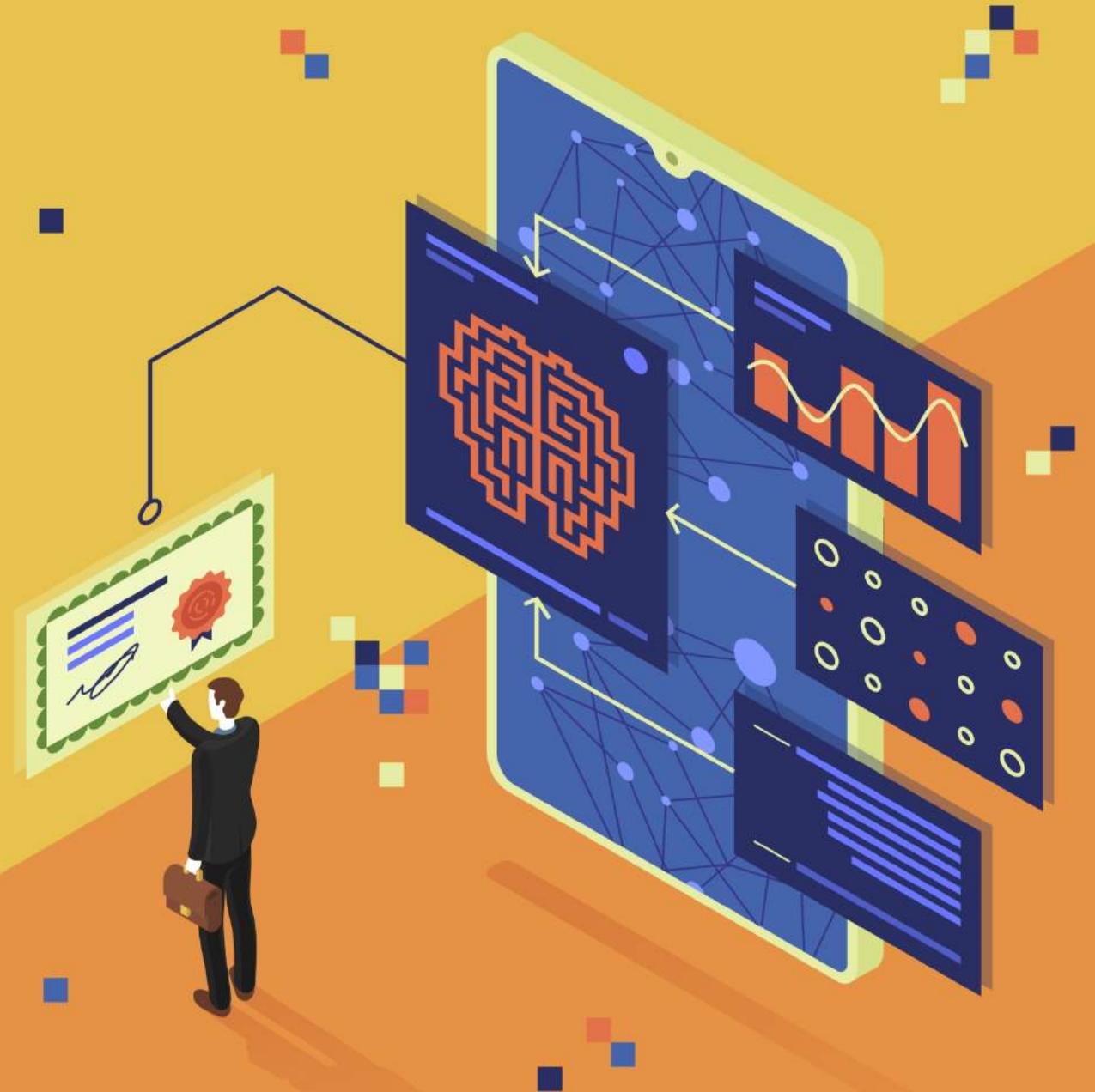


RIDGE REGRESSION (L2 REGULARIZATION): ALPHA EFFECT

- As Alpha increases, the slope of the regression line is reduced and becomes more horizontal.
- As Alpha increases, the model becomes less sensitive to the variations of the independent variable (# Years of experience)



L1 REGULARIZATION (LASSO REGRESSION)



LASSO REGRESSION (L1 REGULARIZATION): MATH

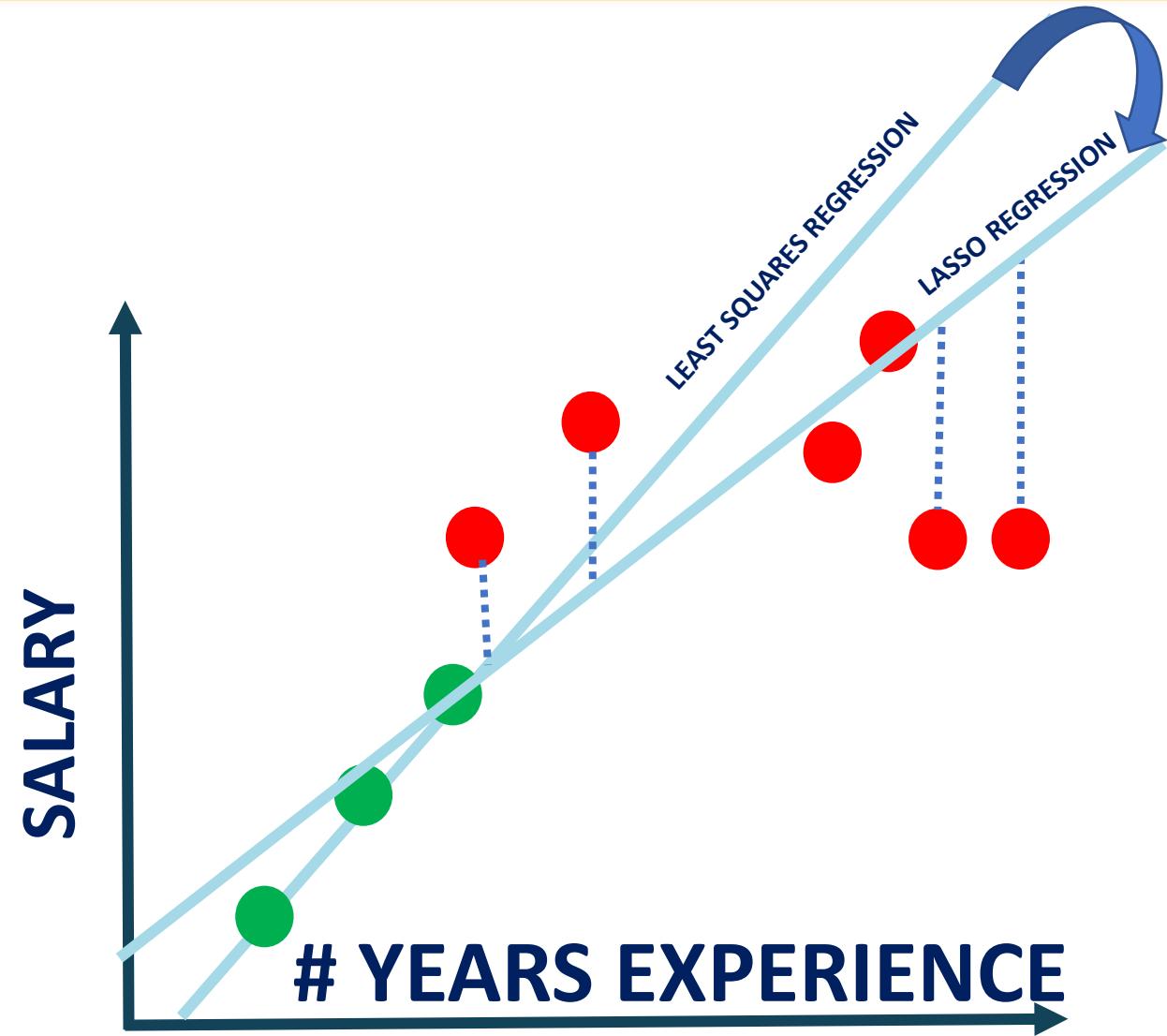


- Lasso Regression is similar to Ridge regression
- It works by introducing a bias term but instead of squaring the slope, the absolute value of the slope is added as a penalty term

Least Squares Regression:
 $\text{Min}(\text{sum of the squared residuals})$

Lasso Regression:
 $\text{Min}(\text{sum of squared residuals} + \alpha * |\text{slope}|)$

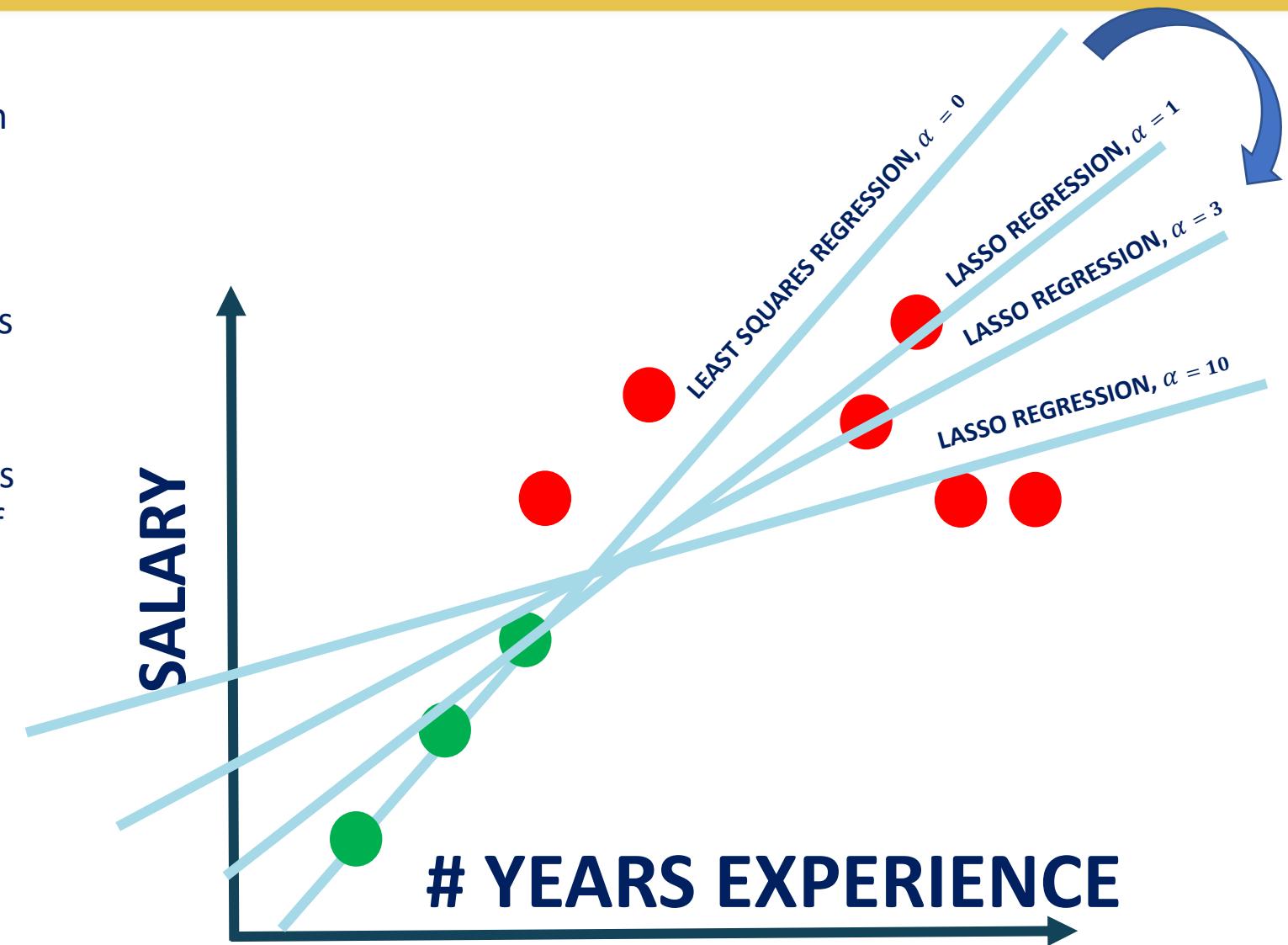
PENALTY TERM



LASSO REGRESSION (L1 REGULARIZATION)



- The effect of Alpha on Lasso regression is similar to its effect on ridge regression
- As Alpha increases, the slope of the regression line is reduced and becomes more horizontal.
- As Alpha increases, the model becomes less sensitive to the variations of the independent variable (# Years of experience)



LASSO REGRESSION: MATH

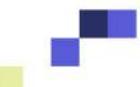


- Lasso regression (L1 regularization) helps reduce overfitting and it is particularly useful for feature selection
- Lasso regression (L1 regularization) can be useful if we have several independent variables that are useless
- Ridge regression can reduce the slope close to zero (but not exactly zero) but Lasso regression can reduce the slope to be exactly equal to zero.

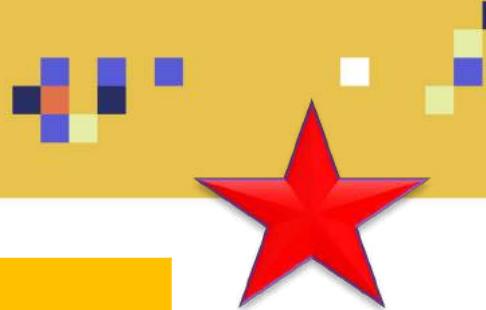
Least Squares Regression:
 $\text{Min}(\text{sum of the squared residuals})$

Ridge Regression (L2 regularization):
 $\text{Min}(\text{sum of squared residuals} + \alpha * \text{slope}^2)$

Lasso Regression (L1 regularization):
 $\text{Min}(\text{sum of squared residuals} + \alpha * |\text{slope}|)$



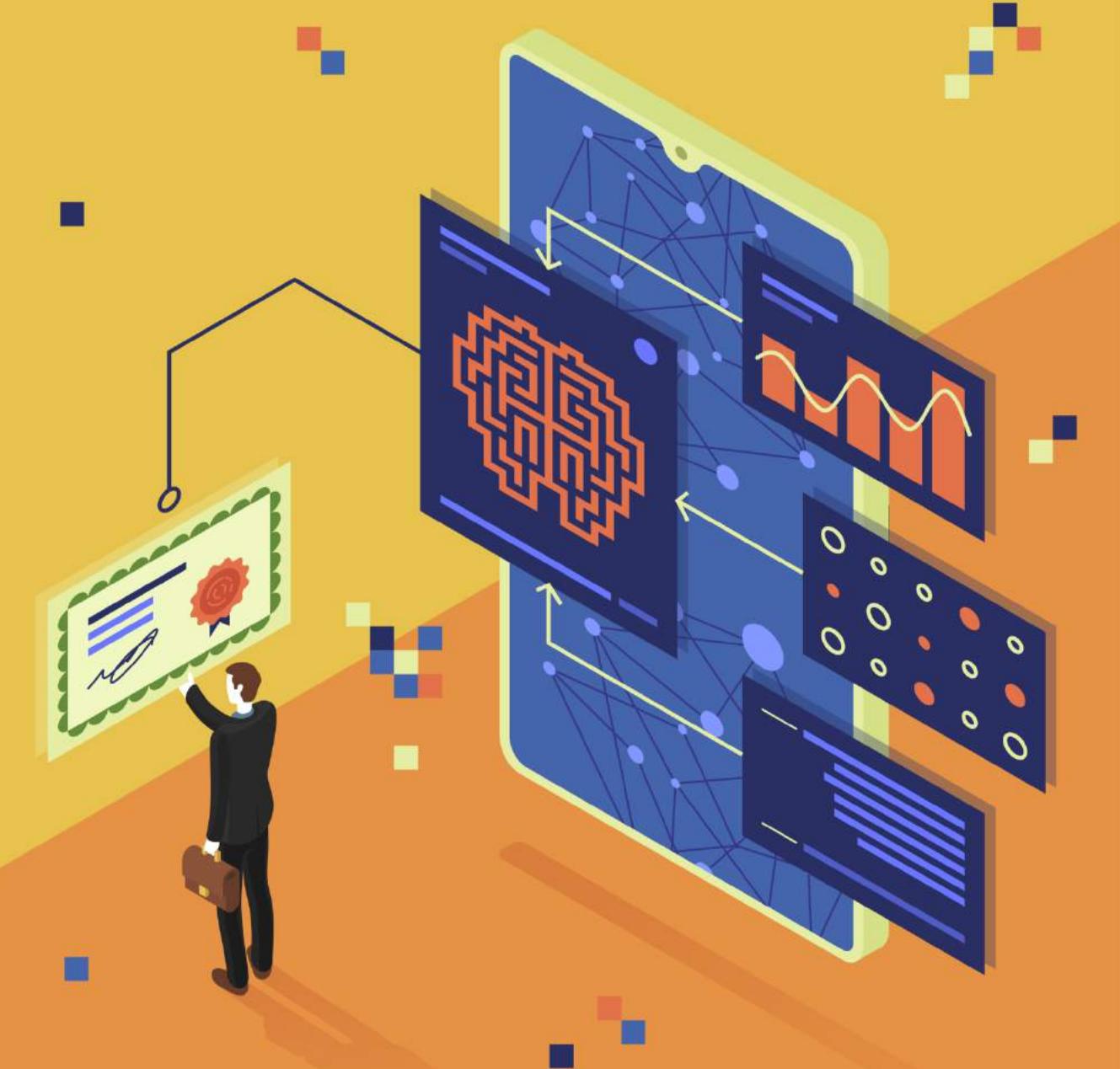
IN SUMMARY



L1 Regularization	L2 regularization
Used to perform feature selection so some features are allowed to go to zero	All features are maintained, but weighted accordingly. No features are allowed to go to zero
Computationally inefficient	Computationally efficient
Sparse output	Dense output

- **When to choose L1?**
 - *If you believe that some features are not important and you can afford to lose them, then L1 regularization is a good choice.*
 - *The output might become sparse since some features might have been removed.*
- **When to choose L2?**
 - *If you believe that all features are important and you'd like to keep them but weigh them accordingly.*

AWS MACHINE LEARNING CERTIFICATION



DOMAIN #3: MODELING

MACHINE AND DEEP LEARNING BASICS – PART #2



AWS ML CERTIFICATION EXAM DOMAINS



Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #3 OVERVIEW:

SECTION #8: MACHINE AND DEEP LEARNING BASICS – PART #1

- Artificial Neural Networks Basics: Single Neuron Model
- Activation Functions
- Multi-Layer Perceptron Model
- How do Artificial Neural Networks Train?
- ANN Parameters Tuning – Learning rate and batch size
- Tensorflow playground
- Gradient Descent and Backpropagation
- Overfitting and Under fitting
- How to overcome overfitting?
- Bias Variance Trade-off
- L1 Regularization
- L2 Regularization

SECTION #9: MACHINE AND DEEP LEARNING BASICS – PART #2

- Artificial Neural Networks Architectures
- Convolutional Neural Networks
- Recurrent Neural Networks
- Vanishing Gradient Problem
- LSTM Networks
- Model Performance Assessment – Confusion Matrix
- Model Performance Assessment – Precision, recall, F1-score
- Model Performance Assessment – ROC, AUC, Heatmap, and RMSE
- K-Fold Cross validation
- Transfer Learning
- Ensemble Learning – Bagging and Boosting

VALUABLE PRIZE!



- For those of you who will successfully complete the entire Module and watch the videos till the end, they will receive a valuable prize!

**10 NEW SAMPLE EXAM QUESTIONS + COMPLETE
ANSWER KEY**



GAME AND MINI CHALLENGES!



- Unfortunately, you can't skip the videos.
- You have to collect a code throughout the lectures to unlock the exam.
- Special characters will appear at random moments throughout the video.
- You will need to collect the code and enter it to a website to access the material.
- That's what the final code might look like!

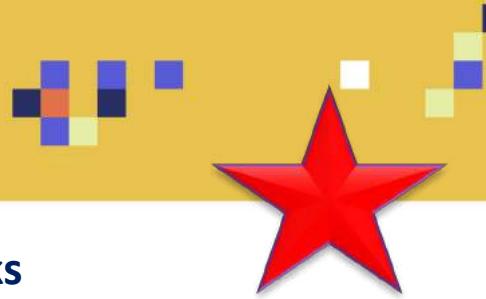
F 2 @ 9 & B



ARTIFICIAL NEURAL NETWORKS ARCHITECTURES

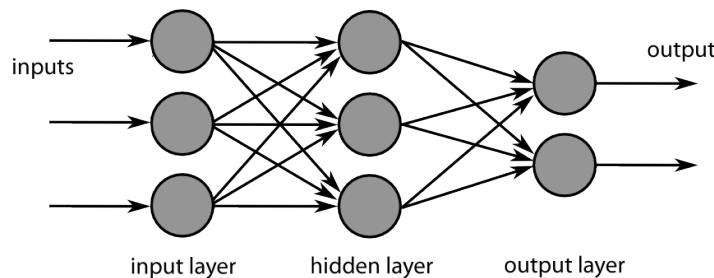


TYPES OF ARTIFICIAL NEURAL NETWORKS



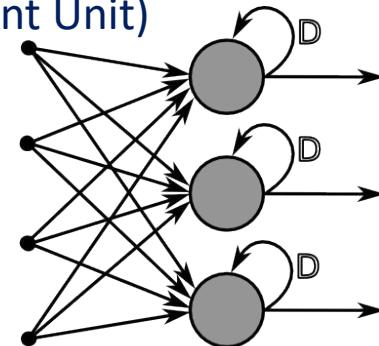
- **Feedforward Neural Networks**

- One to one mapping
- Classification or regression
- information flows from input layer directly through hidden layers then to the output layer without cycles/loops



- **Recurrent Neural Networks**

- Considers temporal (time) information
- Used for sequences prediction (future stock prices, translation, text).
- Long short term memory (LSTM) and GRU (Gated Recurrent Unit)



- **Convolutional Neural Networks**

- Image classification

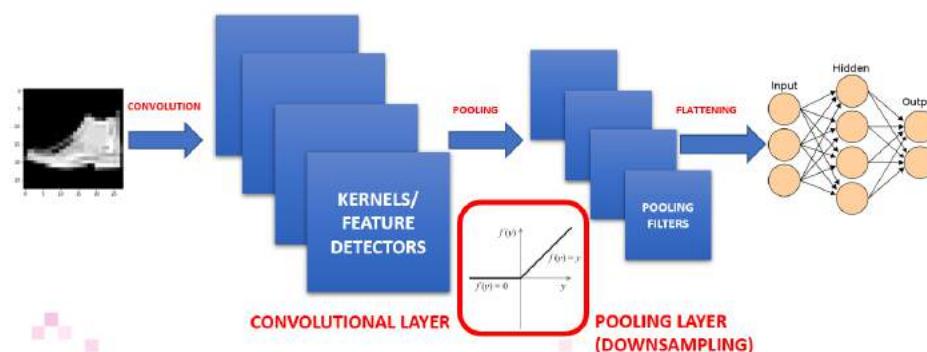


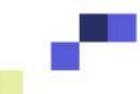
Photo Credit: https://commons.wikimedia.org/wiki/File:RecurrentLayerNeuralNetwork_english.png

Photo Credit: https://commons.wikimedia.org/wiki/File:Artificial_neural_network.svg

DEEP LEARNING FRAMEWORKS



- **TensorFlow:**
 - Google's end-to-end open source machine learning platform
 - <https://www.tensorflow.org/>
- **Keras:**
 - A high-level neural networks API that run on top of TensorFlow and Theano. Keras is super easy to use and is designed for fast implementation.
 - <https://keras.io/>
- **MXNet:**
 - Apache MXNet is an open-source, scalable deep learning framework to train and deploy deep neural networks.
 - <https://mxnet.apache.org/>



CONVOLUTIONAL NEURAL NETWORKS



CONVOLUTIONAL NEURAL NETWORKS BASICS



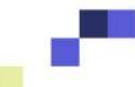
- CNNs are used when data are not represented in column format.
- Here're some applications of CNNs:
 - Image classification
 - Sentiment analysis
 - Machine translation
- CNNs are “feature-location invariant” meaning that CNNs are used to detect and classify objects/features that are not in a specific location.



Photo Credit: <https://www.pexels.com/photo/grey-and-white-short-fur-cat-104827/>

Photo Credit: <https://pixabay.com/photos/cat-kitty-pet-feline-animal-4269479/>

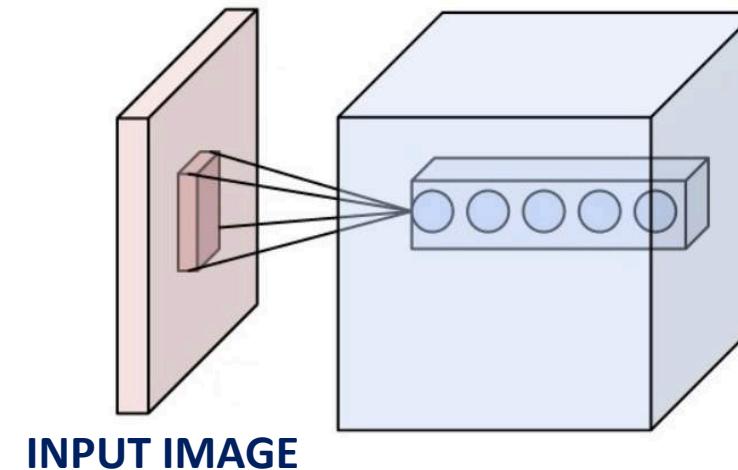
Photo Credit: <https://www.maxpixel.net/Cat-Fur-The-Cat-And-The-Dangerous-Tiger-Animal-4594189>



CONVOLUTIONAL NEURAL NETWORKS BASICS



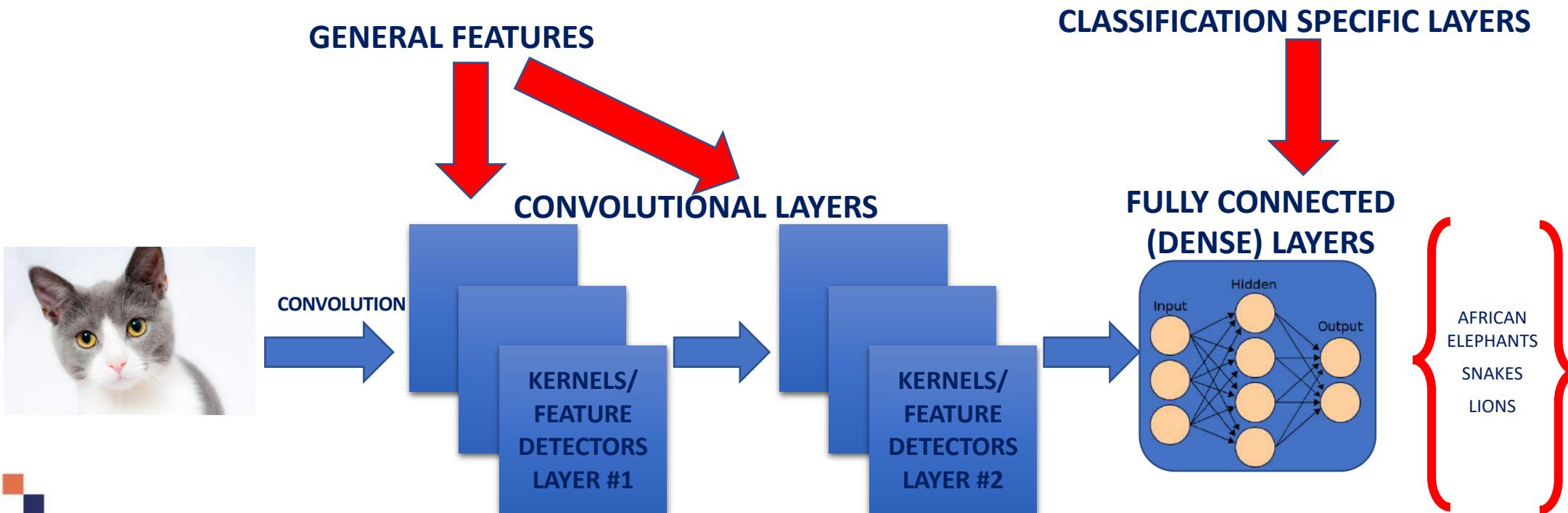
- CNNs are inspired by the visual cortex in the human brain
- CNNs work best with high-dimensional inputs such as images.
- It is generally not practical to connect all the neurons in a certain layer to all the neurons in the subsequent layer.
- Instead, each neuron is connected to only a local region of the input volume. The spatial extent of this connectivity is a hyperparameter named neuron receptive field (filter size).
- These receptive fields overlap to form a complete visual field.



CNN LAYERS



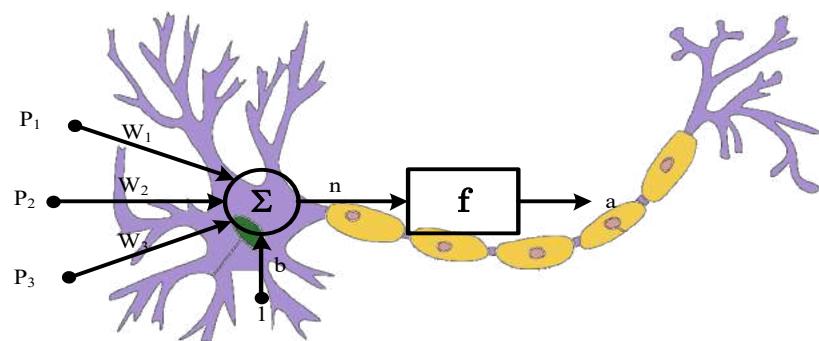
- The first CNN layers are used to **extract high level general features**.
- The last couple of layers are used to perform classification (on a specific task).
- Local receptive fields scan the image first searching for simple shapes such as edges/lines
- These edges are then picked up by the subsequent layer to form more complex features of the cat



CONVOLUTIONAL NEURAL NETWORKS BASICS



- The neuron collects signals from input channels named dendrites, processes information in its nucleus, and then generates an output in a long thin branch called the axon.
- Human learning occurs adaptively by varying the bond strength between these neurons.



$$n = P_1 W_1 + P_2 W_2 + P_3 W_3 + b$$
$$a = f(n)$$

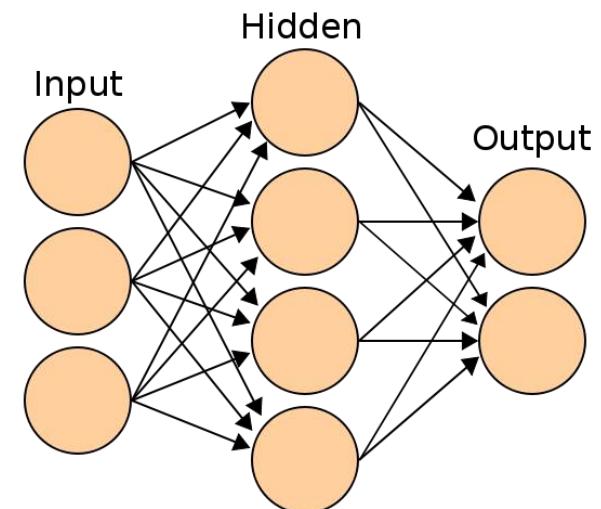
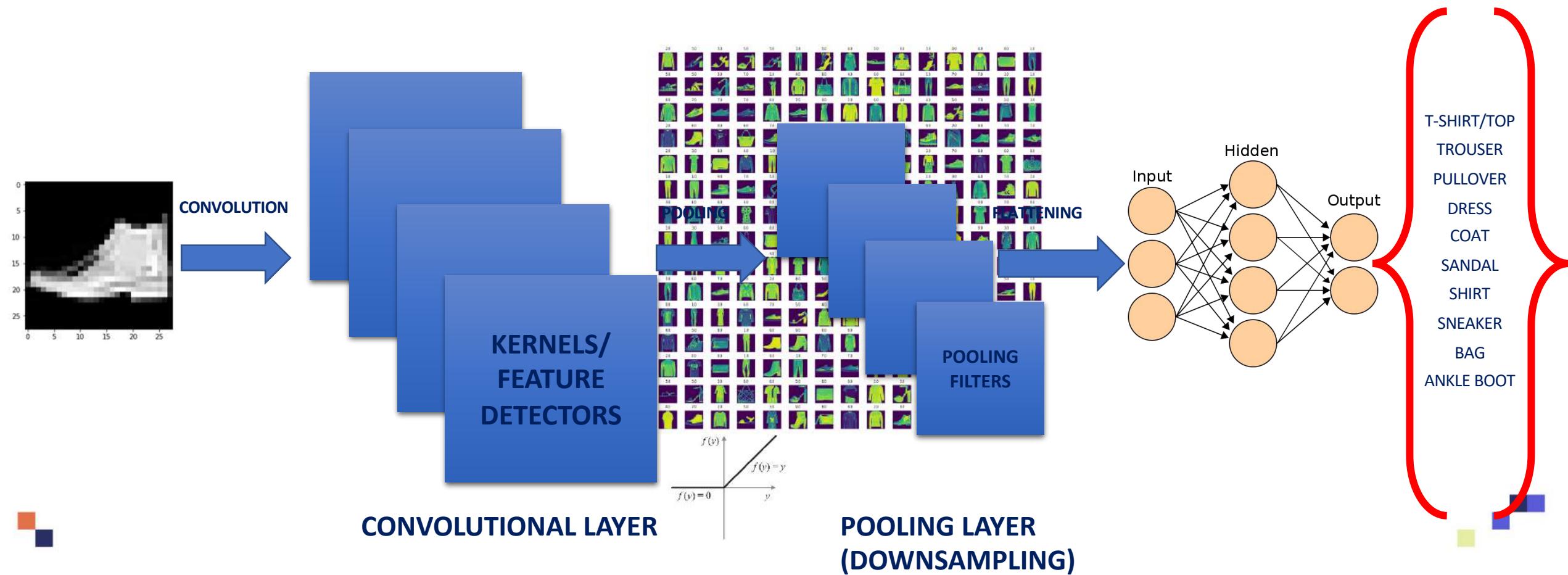


Photo Credit: https://commons.wikimedia.org/wiki/File:Artificial_neural_network.svg
Photo Credit: https://commons.wikimedia.org/wiki/File:Neuron_Hand-tuned.svg



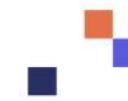
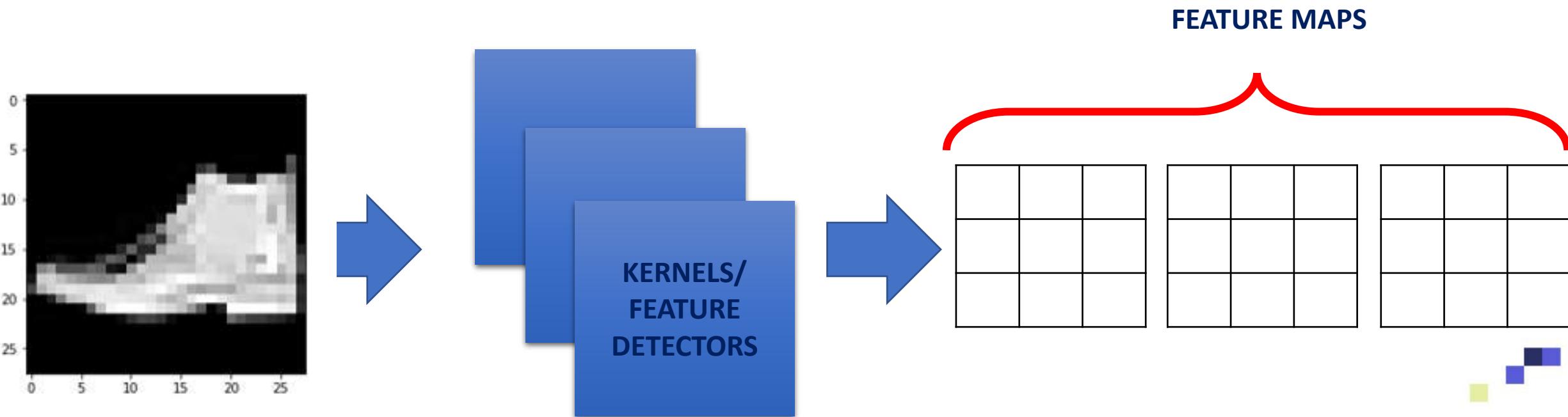
CONVOLUTIONAL NEURAL NETWORKS: ENTIRE NETWORK OVERVIEW



FEATURE DETECTORS



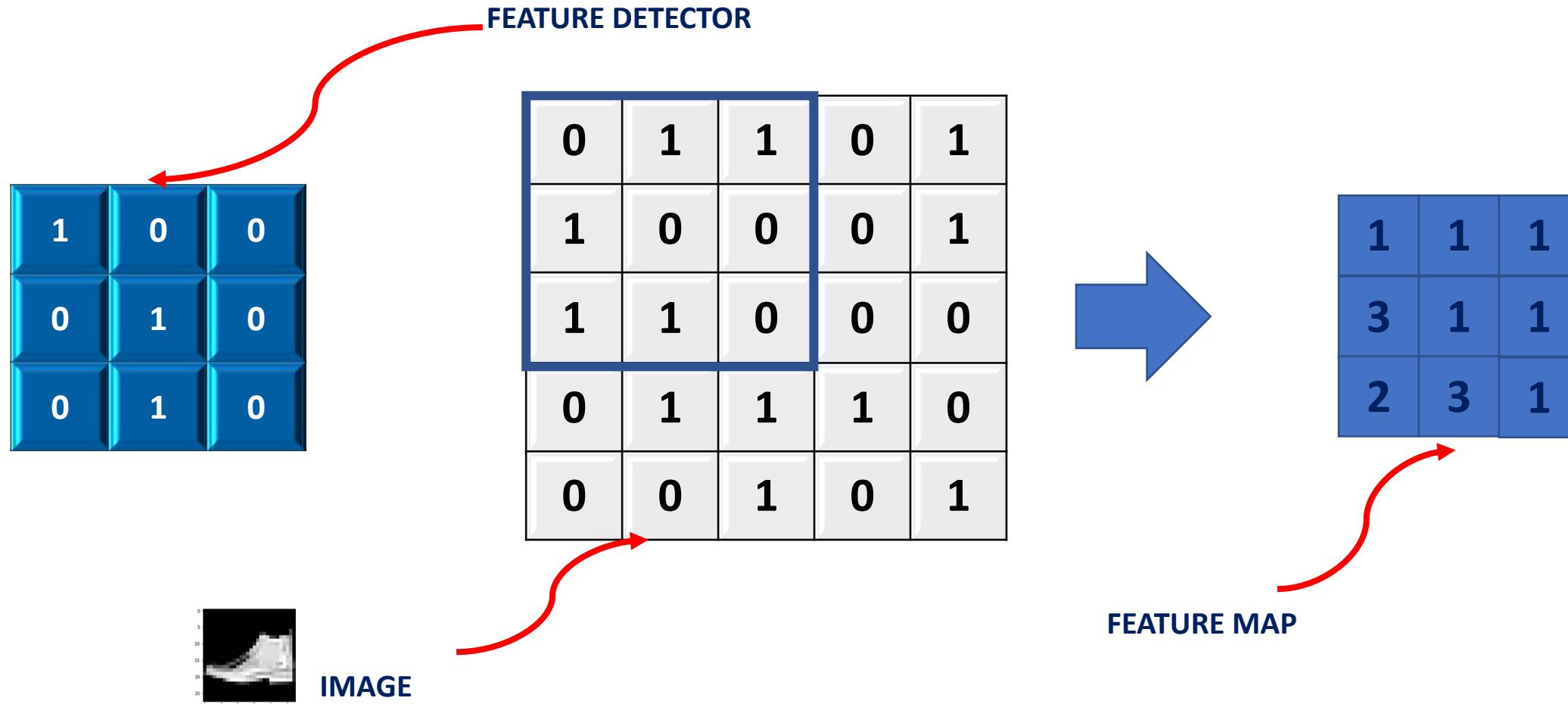
- Convolutions use a kernel matrix to scan a given image and apply a filter to obtain a certain effect.
- An image Kernel is a matrix used to apply effects such as blurring and sharpening.
- Kernels are used in machine learning for feature extraction to select most important pixels of an image.
- Convolution preserves the spatial relationship between pixels.



FEATURE DETECTORS

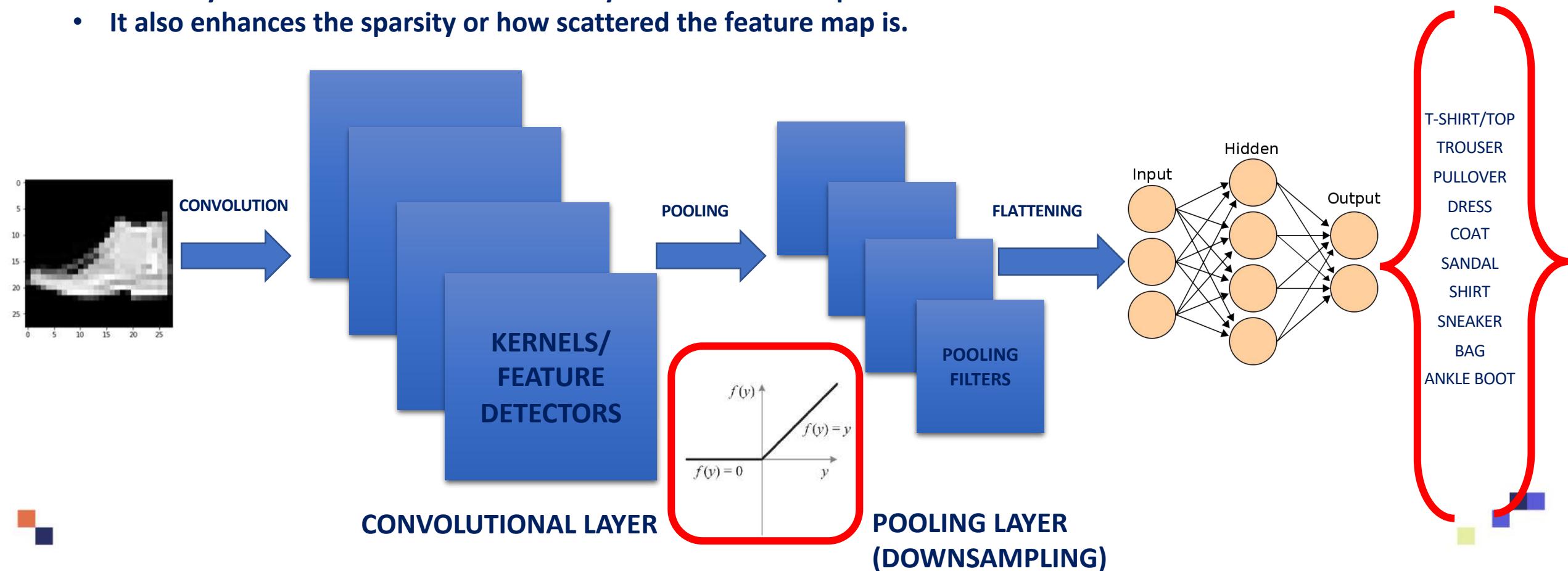


- Live Convolution: <http://setosa.io/ev/image-kernels/>



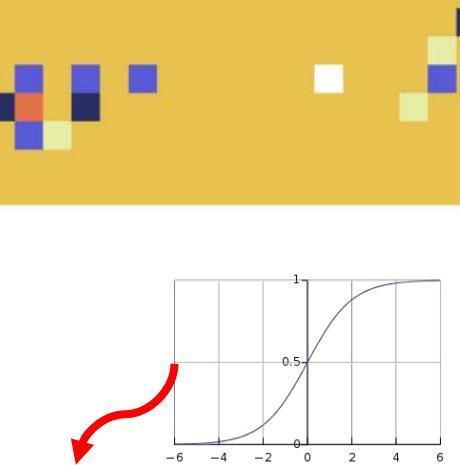
RELU (RECTIFIED LINEAR UNITS)

- RELU Layers are used to add non-linearity in the feature map.
- It also enhances the sparsity or how scattered the feature map is.

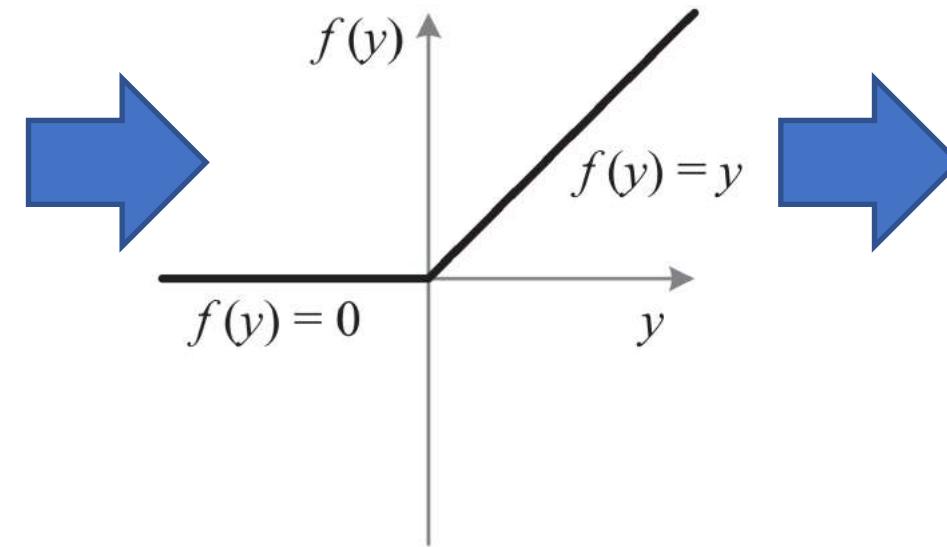


RELU (RECTIFIED LINEAR UNITS)

- RELU Layers are used to add non-linearity in the feature map.
- It also enhances the sparsity or how scattered the feature map is.
- The gradient of the RELU does not vanish as we increase x compared to the sigmoid function



7	10	-5	2	1
1	0	2	3	-6
1	17	-5	0	0
0	1	1	1	0
0	0	-8	12	1

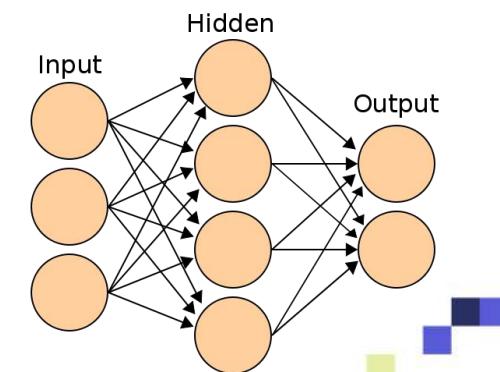
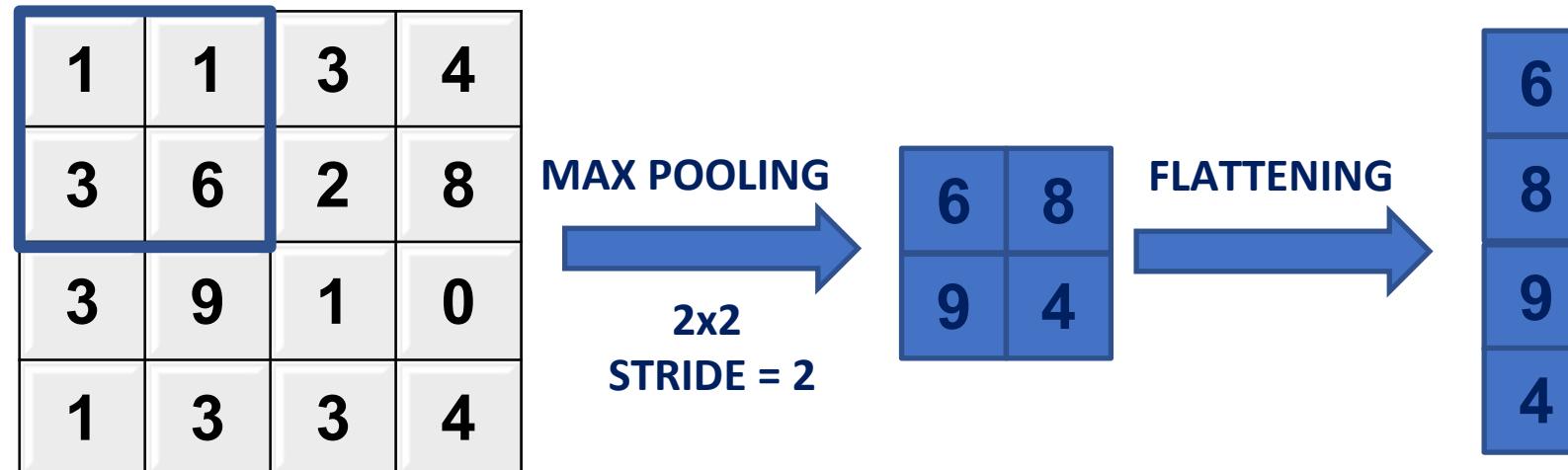


7	10	0	2	1
1	0	2	3	0
1	17	0	0	0
0	1	1	1	0
0	0	0	12	1

POOLING (DOWNSAMPLING)



- Pooling or down sampling layers are placed after convolutional layers to reduce feature map dimensionality.
- This improves the computational efficiency while preserving the features.
- Pooling helps the model to generalize by avoiding overfitting.
- If one of the pixel is shifted, the pooled feature map will still be the same.
- Max pooling works by retaining the maximum feature response within a given sample size in a feature map.
- Live illustration : <http://scs.ryerson.ca/~aharley/vis/conv/flat.html>



CNN IN KERAS (TENSORFLOW 2.0)



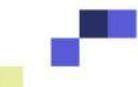
- Let's build a simple CNN in Keras

```
fashion_mnist = tf.keras.datasets.fashion_mnist  
  
(train_images, train_labels), (test_images, test_labels) = fashion_mnist.load_data()  
  
# Reshape training data to be = (60000, 28, 28, 1) instead of (60000, 28,28)  
train_images = train_images.reshape(60000, 28, 28, 1)  
test_images = test_images.reshape(10000, 28, 28, 1)  
  
from tensorflow.keras import datasets, layers, models  
model = models.Sequential()  
  
model.add(layers.Conv2D(6, (5,5), activation = 'relu', input_shape = (28,28,1)))  
model.add(layers.AveragePooling2D())  
  
model.add(layers.Conv2D(16, (5,5), activation = 'relu'))  
model.add(layers.AveragePooling2D())  
  
model.add(layers.Flatten())  
model.add(layers.Dense(120, activation = 'relu'))  
model.add(layers.Dense(84, activation = 'relu'))  
model.add(layers.Dense(10, activation = 'softmax'))  
model.summary()  
  
model.compile(optimizer='adam',  
              loss='sparse_categorical_crossentropy',  
              metrics=['accuracy'])  
  
model.fit(train_images, train_labels, epochs=5)
```

CNN SUMMARY



- CNNs are extremely computational expensive requiring GPUs to train
- CNNs include several hyperparameters such as kernel sizes, number of filters, pooling layers, number of neurons in the dense layers.
- The following architecture is the most common:
 1. Conv2D – performs the convolution on a 2D image
 2. MaxPooling2D – downsampling
 3. Dropout – regularization technique
 4. Flatten – convert 2D layer to a 1D to be fed to Dense (fully connected) network
 5. Dense – fully connected ANN
 6. Softmax
- CNN work best with images of the following dimensions:
 - width x length x color channels
 - Example: 32 x 32 x 3



ADVANCED OFF THE SHELF CNNs



- There are many trained off the shelf convolutional neural networks that are readily available such as:
 - LeNet-5 (1998): 7 level convolutional neural network developed by LeCun that works in classifying hand writing numbers.
 - AlexNet (2012): Offered massive improvement, error reduction from 26% to 15.3%
 - ZFNet (2013): achieved error of 14.8%
 - Googlenet/Inception (2014): error reduction to 6.67% which is at par with human level accuracy.
 - VGGNet (2014)
 - ResNet (2015): Residual Neural Network includes “skip connection” feature and therefore enabled training of 152 layers without vanishing gradient issues. Error of 3.57% which is superior than humans.

<https://medium.com/analytics-vidhya/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5>



RECURRENT NEURAL NETWORKS INTUITION

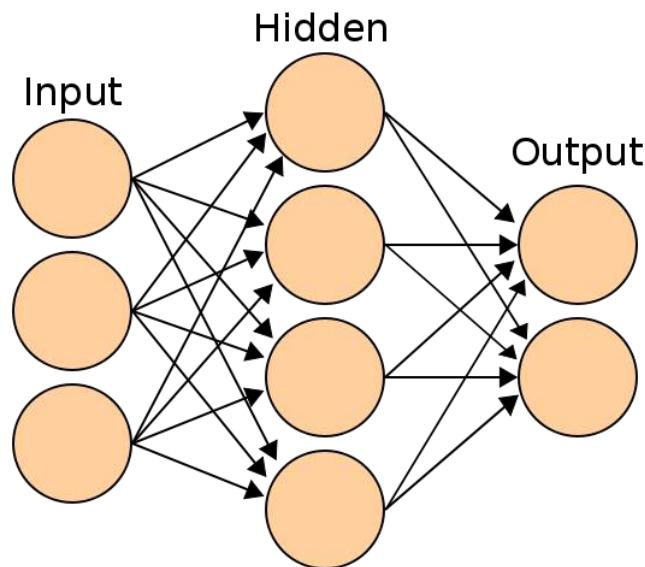


RECURRENT NEURAL NETWORKS (RNN): WHAT ARE THEY?



- We covered Feedforward Neural Networks (vanilla networks) that map a fixed size input (such as image) to a fixed size output (classes or probabilities).
- A drawback in Feedforward networks is that they do not have any time dependency or memory effect.
- A RNN is a type of ANN that is designed to take temporal dimension into consideration by having a memory (internal state) (feedback loop).

FEED FORWARD ANN



RECURRENT NEURAL NETWORK

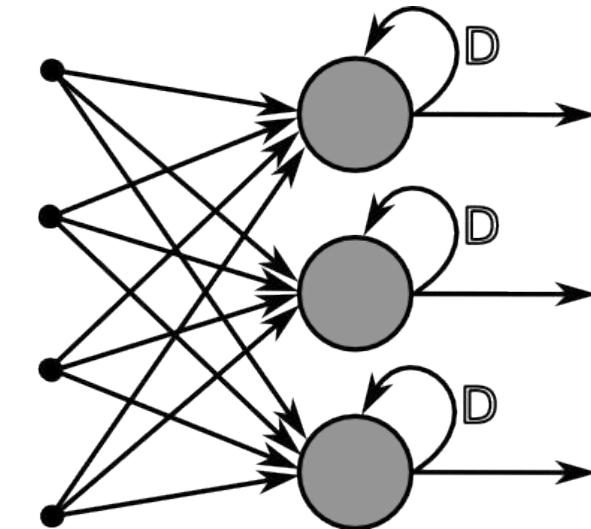


Photo Credit: https://commons.wikimedia.org/wiki/File:RecurrentLayerNeuralNetwork_english.png

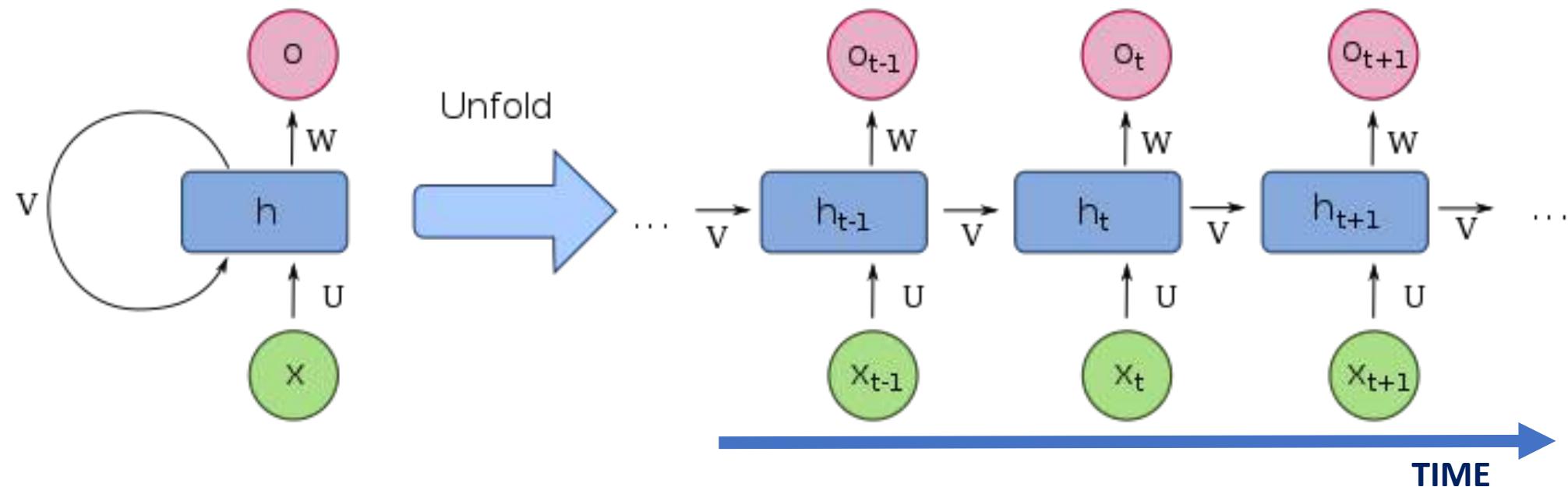
Photo Credit: https://commons.wikimedia.org/wiki/File:Artificial_neural_network.svg



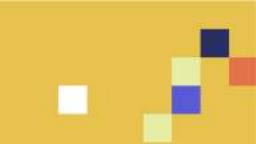
RNN ARCHITECTURE



- A RNN contains a temporal loop in which the hidden layer not only gives an output but it feeds itself as well.
- An extra dimension is added which is time!
- RNN can recall what happened in the previous time stamp so it works great with sequence of text.

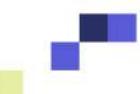
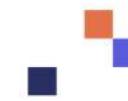


RNNs WORK LIKE MAGIC!



We'll train RNNs to generate text character by character and ponder the question "how is that even possible?"

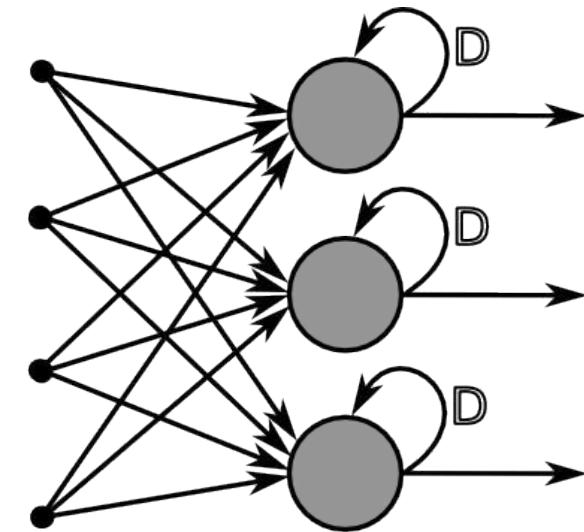
Source: The Unreasonable Effectiveness of Recurrent Neural Networks by Andrej Karpathy
<http://karpathy.github.io/2015/05/21/rnn-effectiveness/>



WHAT MAKES RNNs SO SPECIAL?

- Feedforward ANNs are so constrained with their fixed number of input and outputs.
- For example, a CNN will have fixed size image (28x28) and generates a fixed output (class or probabilities).
- Feedforward ANN have a fixed configuration, i.e.: same number of hidden layers and weights.
- Recurrent Neural Networks offer huge advantage over feedforward ANN and they are much more fun!
- RNN allow us to work with a sequence of vectors:
 - Sequence in inputs
 - Sequence in outputs
 - Sequence in both!

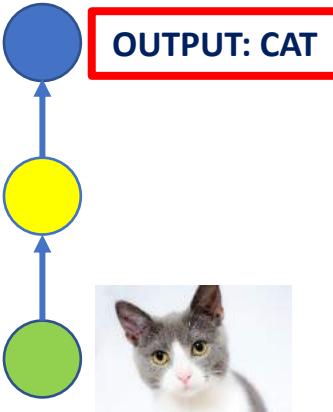
RECURRENT NEURAL NETWORK



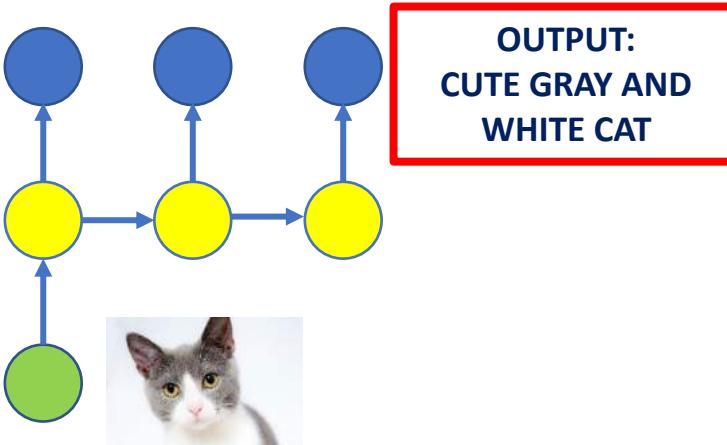
Source: The Unreasonable Effectiveness of Recurrent Neural Networks by Andrej Karpathy
<http://karpathy.github.io/2015/05/21/rnn-effectiveness/>

Photo Credit: https://commons.wikimedia.org/wiki/File:RecurrentLayerNeuralNetwork_english.png

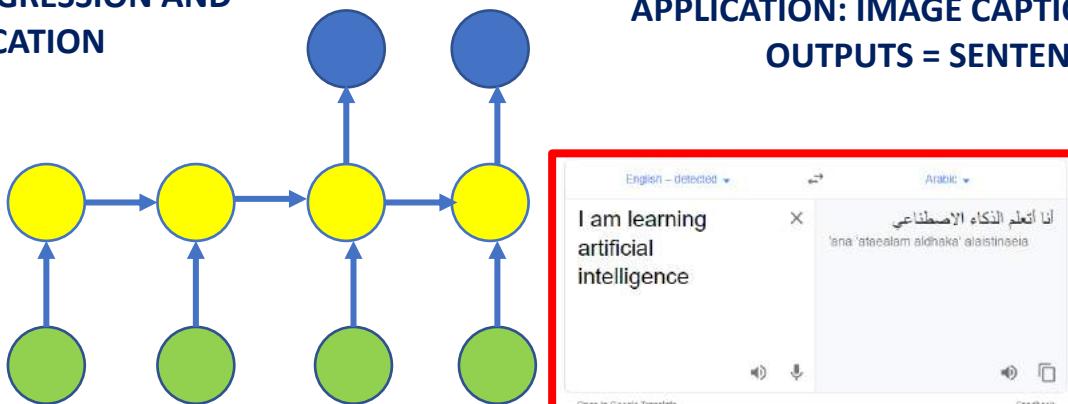
WHAT MAKES RNNs SO SPECIAL?



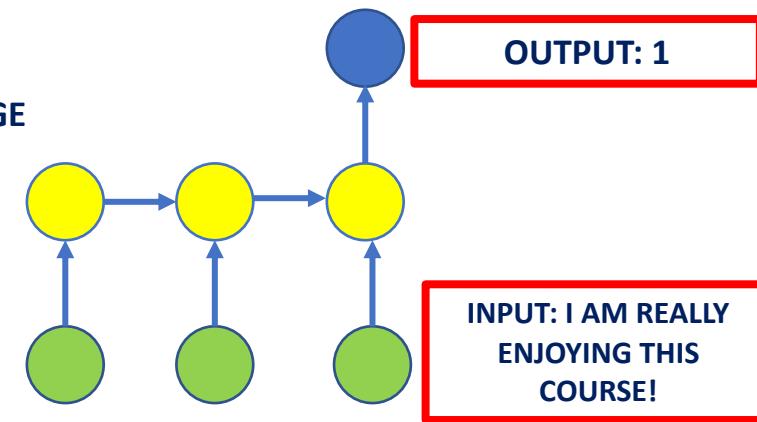
ONE TO ONE (VANILLA)
APPLICATION: REGRESSION AND
CLASSIFICATION



ONE TO MANY (SEQUENCE OUTPUT)
APPLICATION: IMAGE CAPTIONING, INPUT = IMAGE
OUTPUTS = SENTENCE OF WORDS



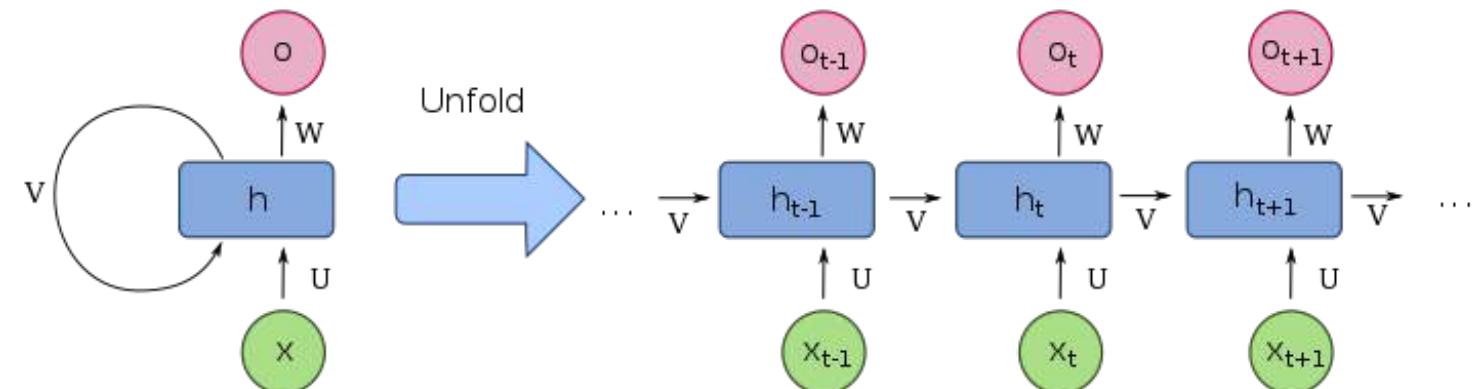
MANY TO MANY (SEQUENCE INPUT AND OUTPUT)
APPLICATION: LANGUAGE TRANSLATION,



MANY TO ONE (SEQUENCE INPUT)
APPLICATION: SENTIMENT ANALYSIS,
EX: IS A REVIEW POSITIVE OR NEGATIVE?



- A RNN accepts an input x and generate an output o .
- The output o does not depend on the input x alone, however, it depends on the entire history of the inputs that have been fed to the network in previous time steps.
- Two equations that govern the RNN are as follows:
 - **INTERNAL STATE UPDATE:**
$$h_t = \tanh(X_t * U + h_{t-1} * V)$$
 - **OUTPUT UPDATE:**
$$o_t = \text{softmax}(W * h_t)$$



LET'S WATCH THIS MOVIE WRITTEN BY AN RNN!

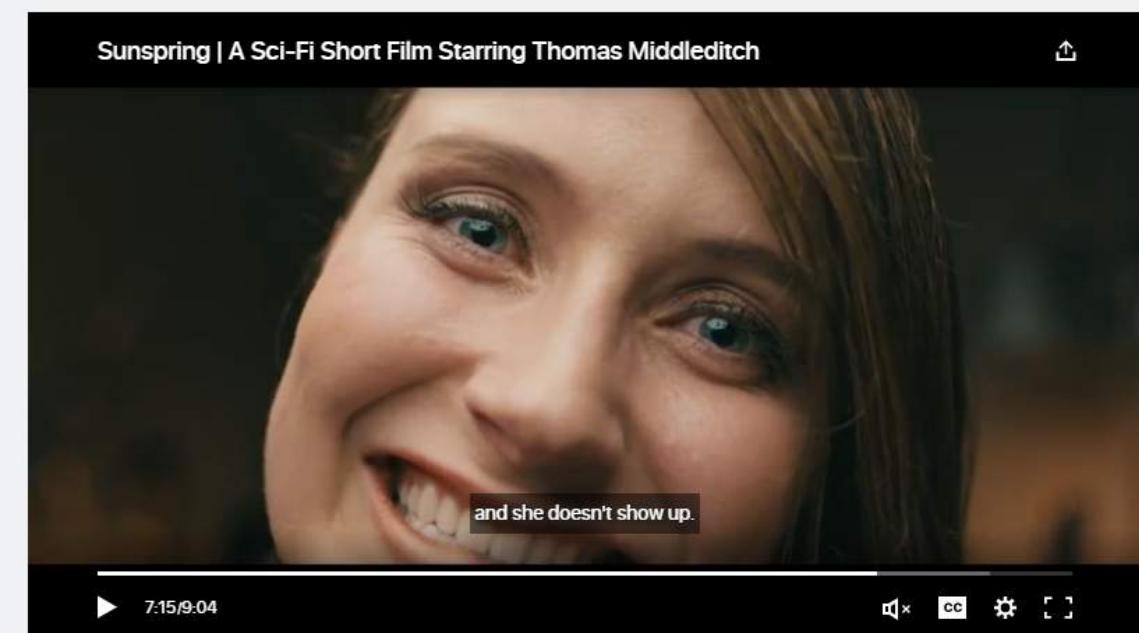
- Let's watch a movie written by AI!
<https://arstechnica.com/gaming/2016/06/an-ai-wrote-this-movie-and-its-strangely-moving/>
- The movie was written by an LSTM recurrent neural network
- The LSTM network was trained with a corpus of dozens of sci-fi screenplays from movies from the 1980s and 90s.

GAMING & CULTURE —

Movie written by algorithm turns out to be hilarious and intense

For *Sunspring*'s exclusive debut on Ars, we talked to the filmmakers about collaborating with an AI.

ANNALEE NEWITZ - 6/9/2016, 6:30 AM



Sunspring, a short science fiction movie written entirely by AI, debuts exclusively on Ars today.

Photo Credit: https://fr.wikipedia.org/wiki/Fichier:Recurrent_neural_network_unfold.svg

VANISHING GRADIENT PROBLEM



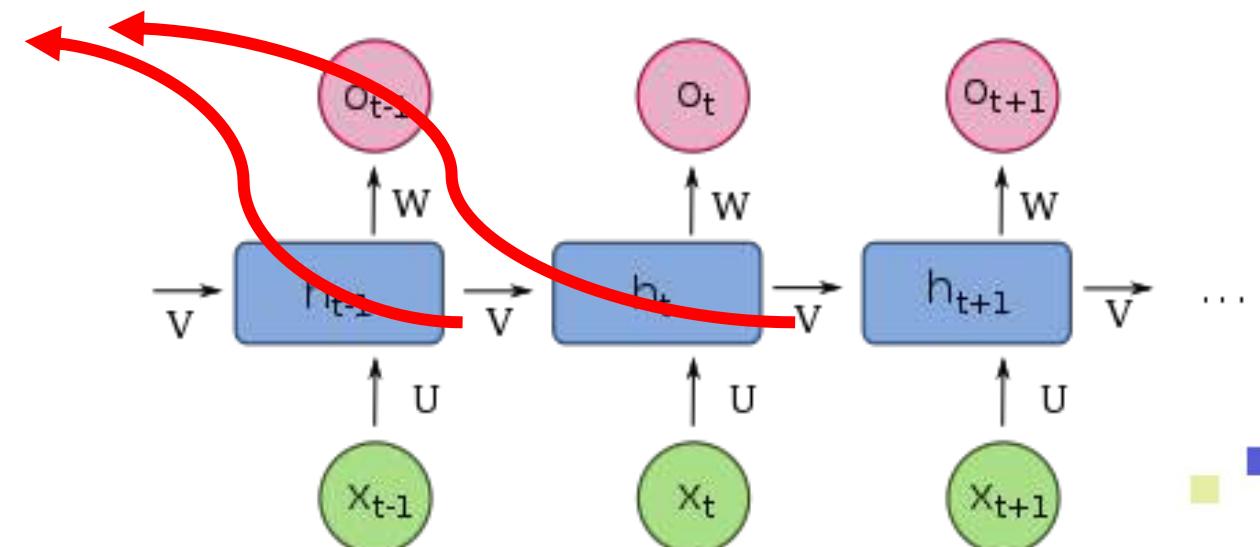
VANISHING GRADIENT PROBLEM



- LSTM networks work much better compared to vanilla RNN since they overcome the vanishing gradient problem.
- The error has to propagate through all the previous layers resulting in a vanishing gradient.
- As the gradient goes smaller, the network weights are no longer updated.
- As more layers are added, the gradients of the loss function approaches zero, making the network hard to train.

EACH LAYER DEPENDS ON THE OUTPUT FROM THE PREVIOUS LAYERS, THE “V” IS MULTIPLIED SEVERAL TIMES RESULTING IN VANISHING GRADIENT

$$0.1 * 0.1 * 0.1 * \dots * 0.1 = 1e-10$$

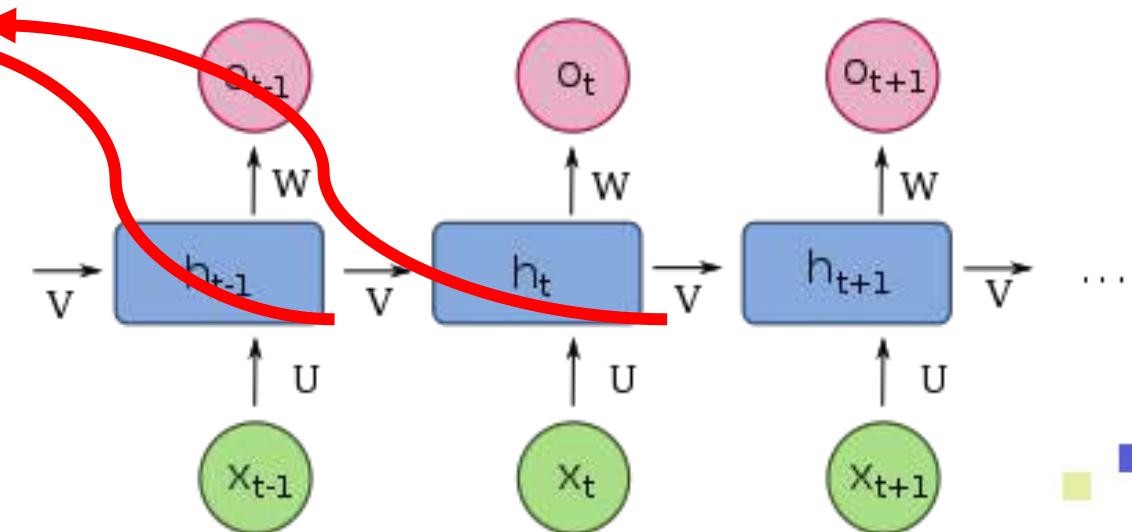


VANISHING GRADIENT PROBLEM

- ANN gradients are calculated during backpropagation.
- In backpropagation, we calculate the derivatives of the network by moving from the outermost layer (close to output) back to the initial layers (close to inputs).
- The chain rule is used during this calculation in which the derivatives from the final layers are multiplied by the derivatives from early layers.
- The gradients keeps diminishing exponentially and therefore the weights and biases are no longer being updated.

EACH LAYER DEPENDS ON THE OUTPUT FROM THE PREVIOUS LAYERS, THE "V" IS MULTIPLIED SEVERAL TIMES RESULTING IN VANISHING GRADIENT, (ex: $0.1 * 0.1 * 0.1 * \dots * 0.1 = 1e-10$)

$$\begin{aligned}\text{New Weight} &= \text{Old Weight} - \text{Learning rate} * \text{gradient} \\ 9.09999 &= 10.1 - 1 * 0.001\end{aligned}$$



VANISHING GRADIENT PROBLEM SOLUTION

- **Choose Proper Activation Function:** Use RELU activation function instead of Tanh or sigmoid activation functions
- **Use Long short term memory networks (LSTM):** LSTM could be used instead of pure vanilla Recurrent neural networks
- **Use ResNet (residual networks)**
- **Multi-level hierarchy:** multi-level hierarchy of networks pre-trained one level at a time through unsupervised learning, fine-tuned through backpropagation.
- **Gradient Checking:** a debugging strategy used to numerically track and assess gradients during training

RELU
 $\text{ReLU}(x) \triangleq \max(0, x)$

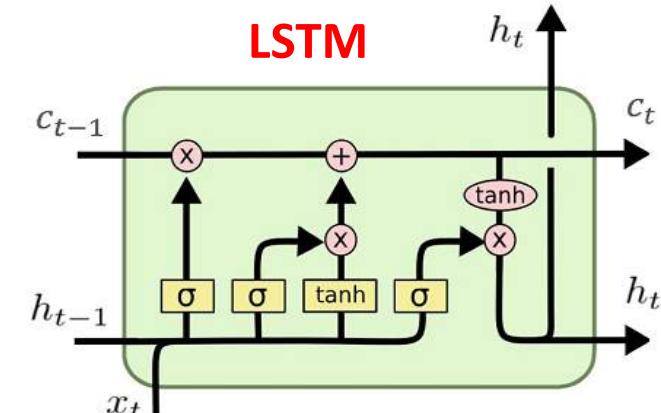
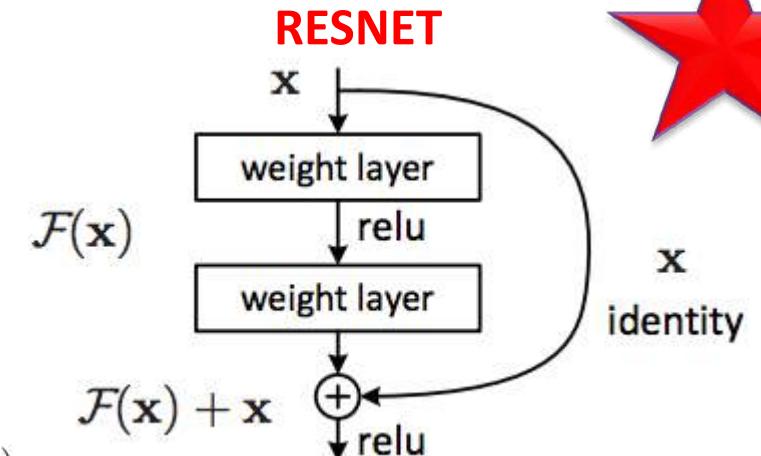
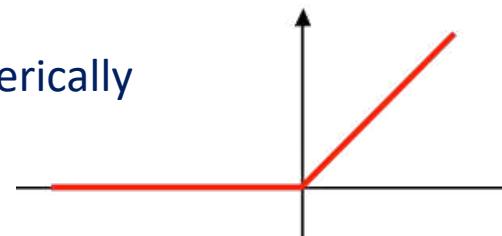


Photo Credit: <https://commons.wikimedia.org/wiki/File:Resnet.png>

Reference and Photo Credit: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

Photo Credit: <https://commons.wikimedia.org/wiki/File:LSTM.png>

LONG SHORT TERM MEMORY (LSTM) NETWORKS

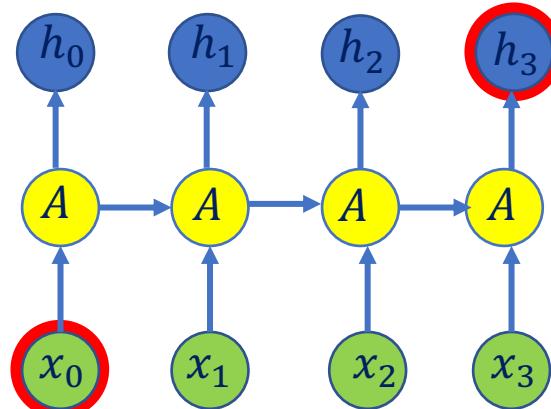


LSTM INTUITION



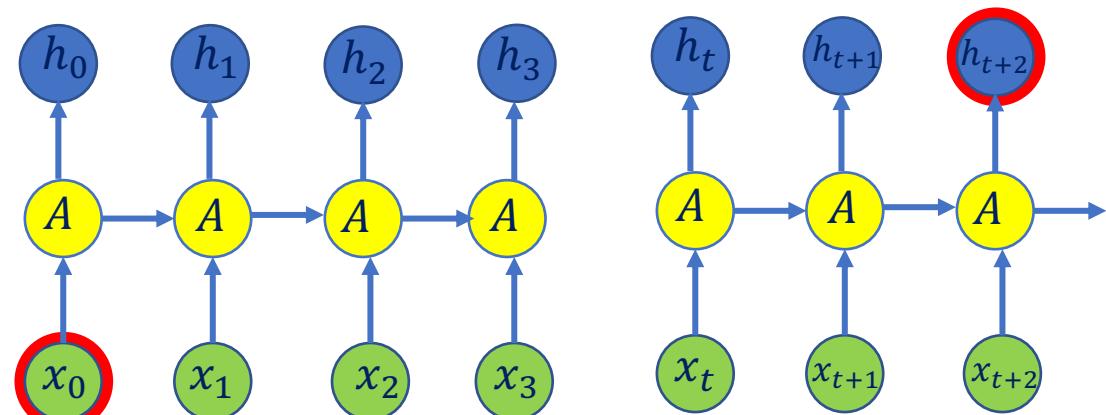
- LSTMs work better compared to vanilla RNN since they overcome vanishing gradient problem.
- In practice, RNN fail to establish long term dependencies.
- Reference: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

The tree color is “green”



RNN PERFORMS WELL SINCE THE GAP BETWEEN THE PREDICTION “GREEN” AND THE NECESSARY CONTEXT INFORMATION “TREE” IS SMALL

I live in Quebec in Northern Canada.....where I live, the weather is generally “cold” most of the year



RNN PERFORMS POORLY WHEN THE GAP BETWEEN THE PREDICTION “COLD” AND THE NECESSARY CONTEXT INFORMATION “CANADA” IS LARGE

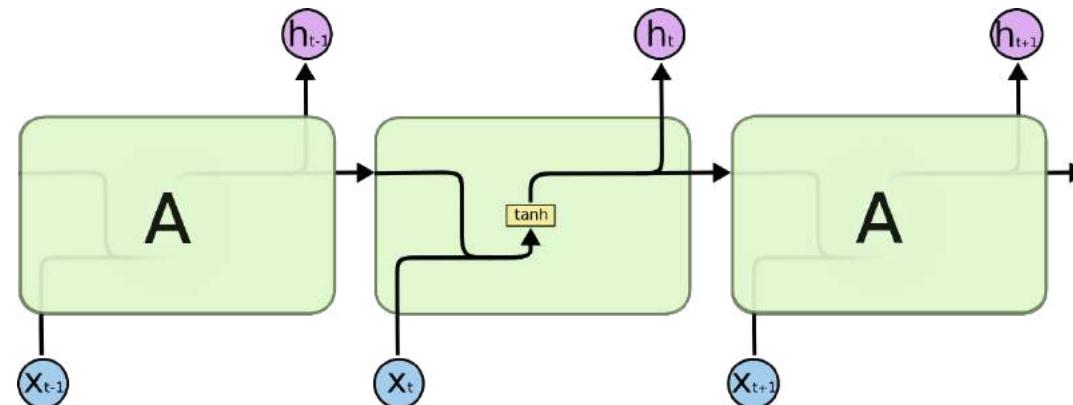


LSTM INTUITION

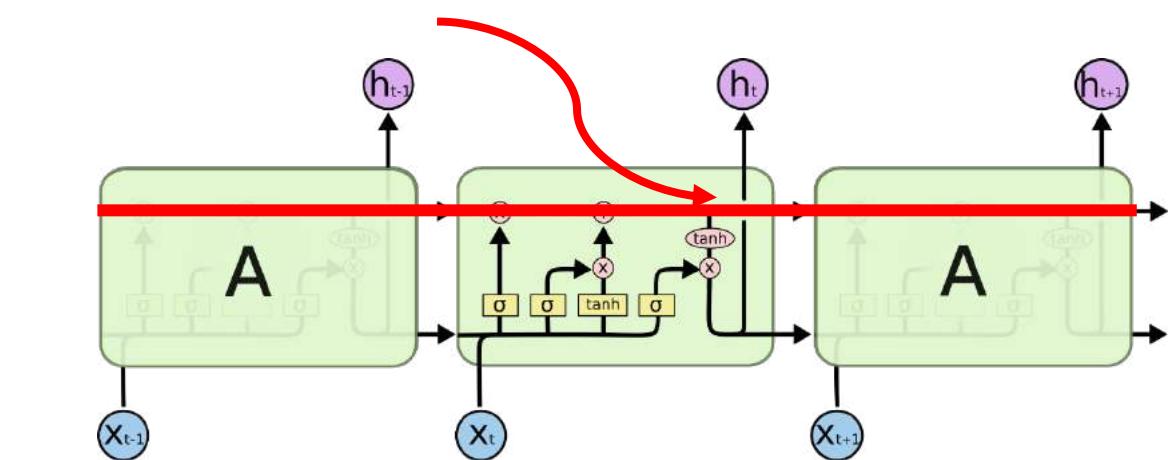


- LSTM networks are type of RNN that are designed to remember long term dependencies by default.
- LSTM can remember and recall information for a prolonged period of time.
- Recall that each line represents a full vector.

THIS HORIZONTAL LINE (MEMORY) OR CELL STATE
ENABLES LSTM TO REMEMBER VERY OLD INFORMATION



VANILLA RECURRENT NEURAL NETWORK



LONG SHORT TERM MEMORY NETWORK

Reference and Photo Credit:

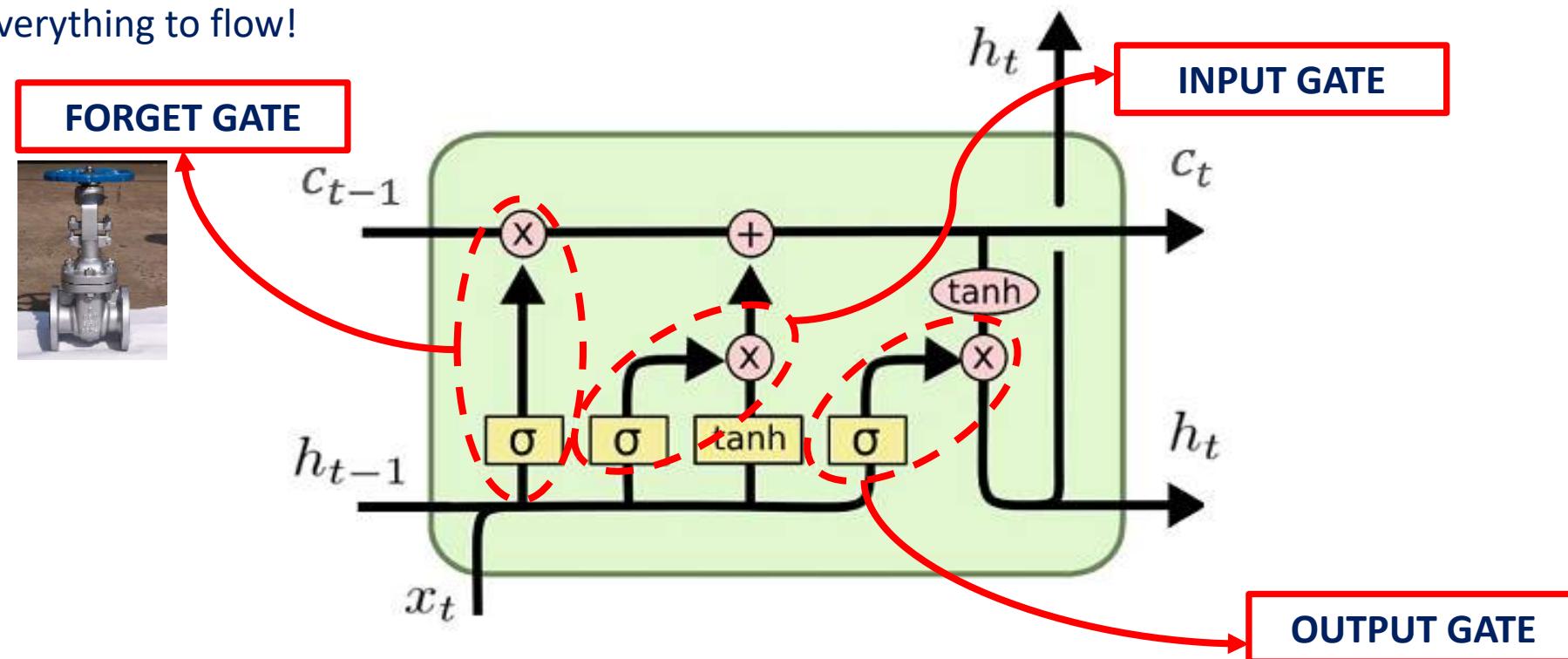
<https://colah.github.io/posts/2015-08-Understanding-LSTMs/>



LSTM INTUITION – GATES



- LSTM contains gates that can allow or block information from passing by.
- Gates consist of a sigmoid neural net layer along with a pointwise multiplication operation.
- Sigmoid output ranges from 0 to 1:
 - 0 = Don't allow any data to flow
 - 1 = Allow everything to flow!



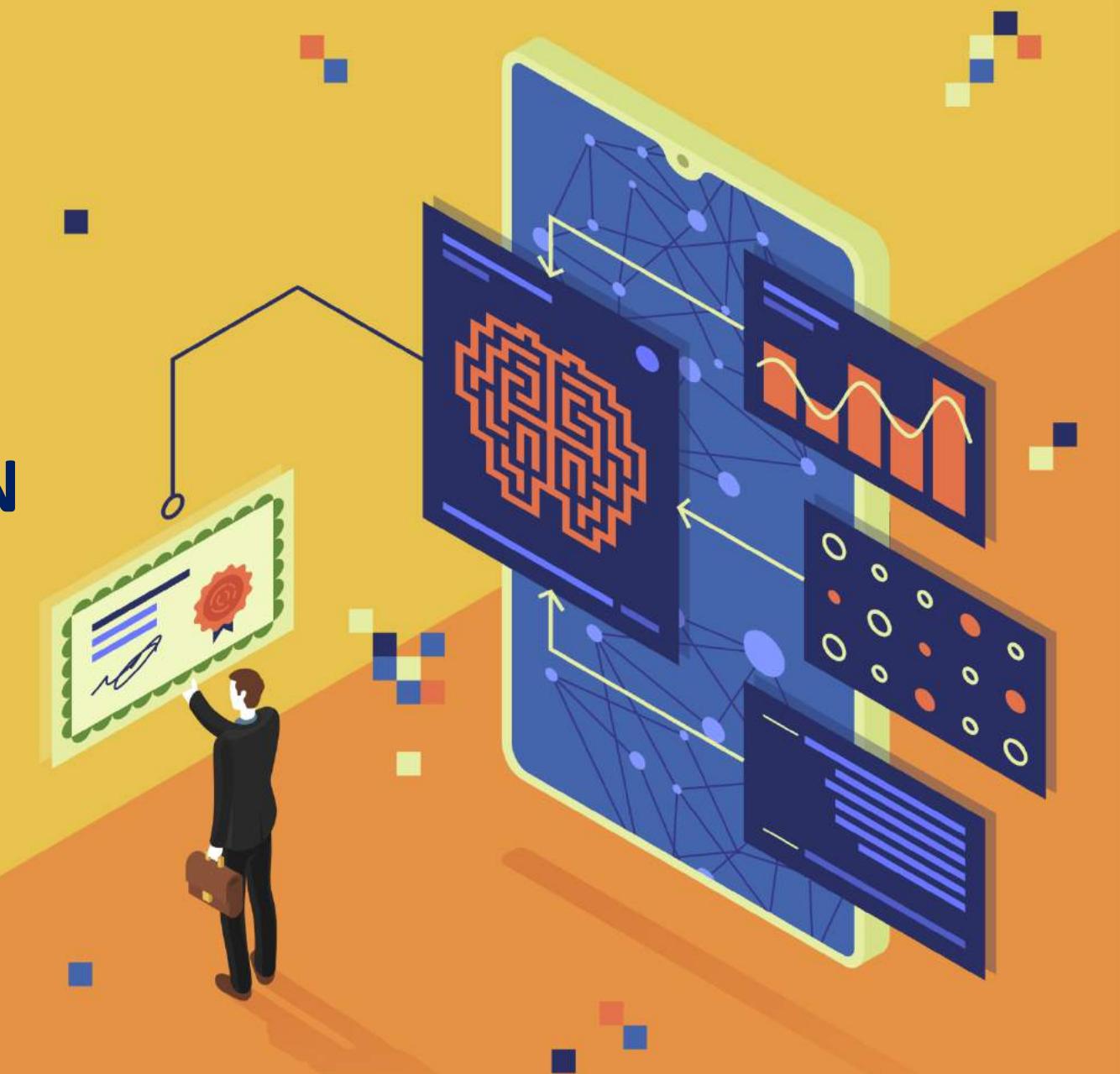
Reference and Photo Credit: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

Photo Credit: <https://commons.wikimedia.org/wiki/File:LSTM.png>

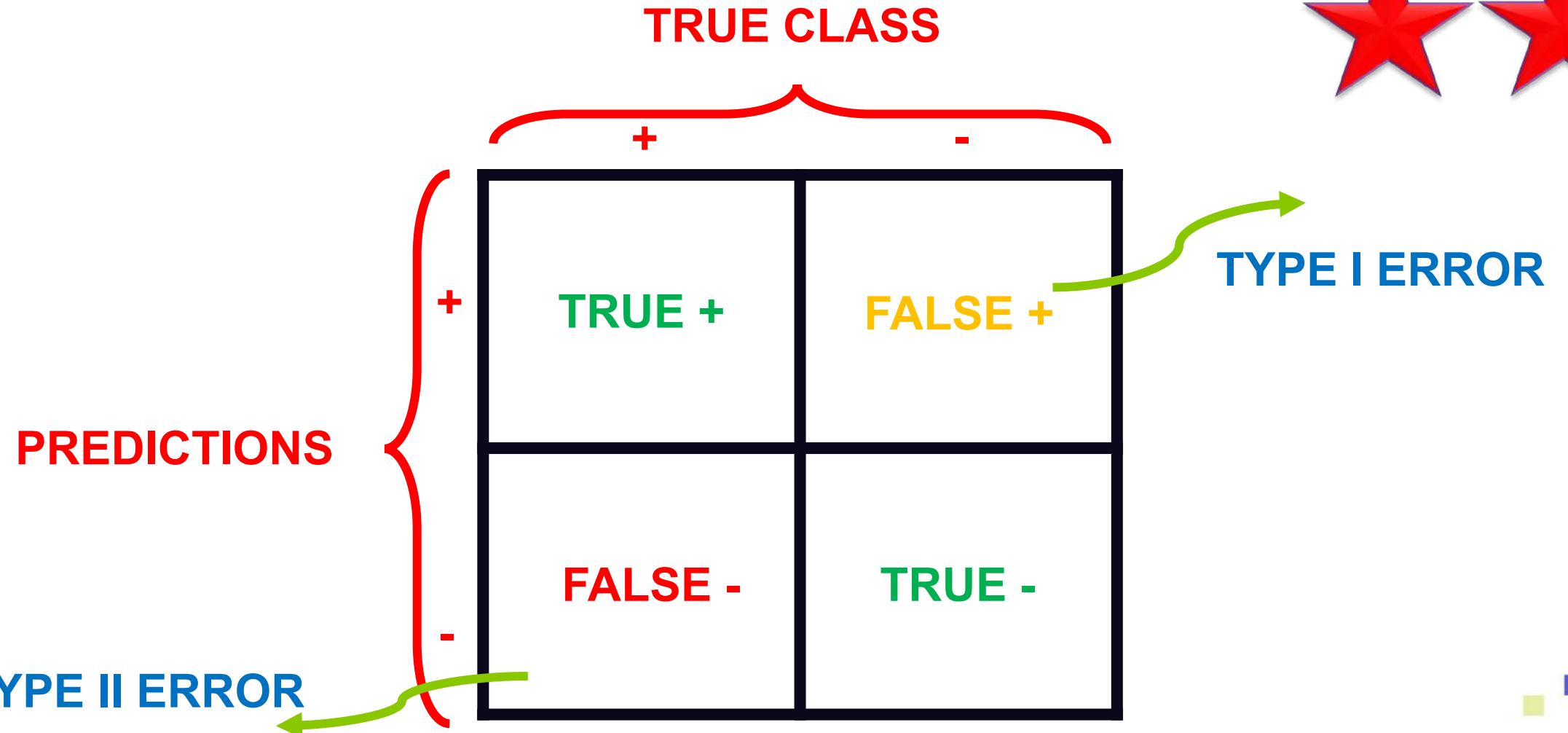
RNN Training

- RNNs are trained using Backpropagation through time
- Similar to backpropagation on feedforward ANN, but applied to each time step.
- Since we have several time steps, RNNs training is computationally expensive
- Backpropagation can be limited to a certain window in time by truncating backpropagation
- RNN training is sensitive to network topologies and hyperparameters.

MODEL PERFORMANCE ASSESSMENT – CONFUSION MATRIX



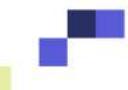
CONFUSION MATRIX



CONFUSION MATRIX



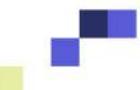
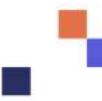
- A confusion matrix is used to describe the performance of a classification model:
 - True positives (TP): cases when classifier predicted TRUE (they have the disease), and correct class was TRUE (patient has disease).
 - True negatives (TN): cases when model predicted FALSE (no disease), and correct class was FALSE (patient do not have disease).
 - False positives (FP) (Type I error): classifier predicted TRUE, but correct class was FALSE (patient did not have disease).
 - False negatives (FN) (Type II error): classifier predicted FALSE (patient do not have disease), but they actually do have the disease



KEY PERFORMANCE INDICATORS (KPI)



- Classification Accuracy = $(TP+TN) / (TP + TN + FP + FN)$
- Misclassification rate (Error Rate) = $(FP + FN) / (TP + TN + FP + FN)$
- Precision = TP/Total TRUE Predictions = TP/ (TP+FP) (When model predicted TRUE class, how often was it right?)
- Recall = TP/ Actual TRUE = TP/ (TP+FN) (when the class was actually TRUE, how often did the classifier get it right?)



MODEL PERFORMANCE ASSESSMENT – PRECISION, RECALL AND F1-SCORE



PRECISION Vs. RECALL EXAMPLE

PREDICTIONS

TRUE CLASS	
+	-
TP = 1	FP = 1
-	FN = 8 TN = 90

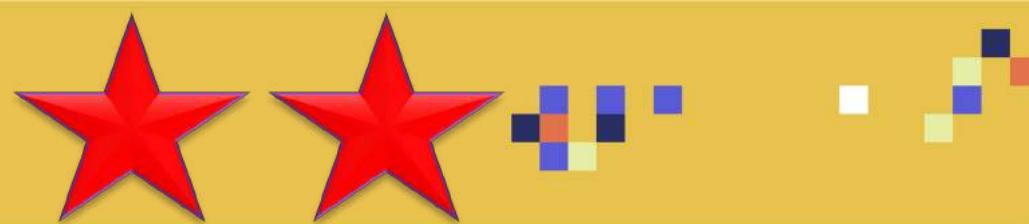
- Classification Accuracy = $(TP+TN) / (TP + TN + FP + FN)$ = 91%
- Precision = TP/Total TRUE Predictions = TP/ (TP+FP) = $\frac{1}{2}$ =50%
- Recall = TP/ Actual TRUE = TP/ (TP+FN) = $1/9$ = 11%

FACTS:
100 PATIENTS TOTAL
91 PATIENTS ARE HEALTHY
9 PATIENTS HAVE CANCER

- Accuracy is generally misleading and is not enough to assess the performance of a classifier.
- Recall is an important KPI in situations where:
 - Dataset is highly imbalanced; cases when you have small cancer patients compared to healthy ones.



PRECISION DEEP DIVE



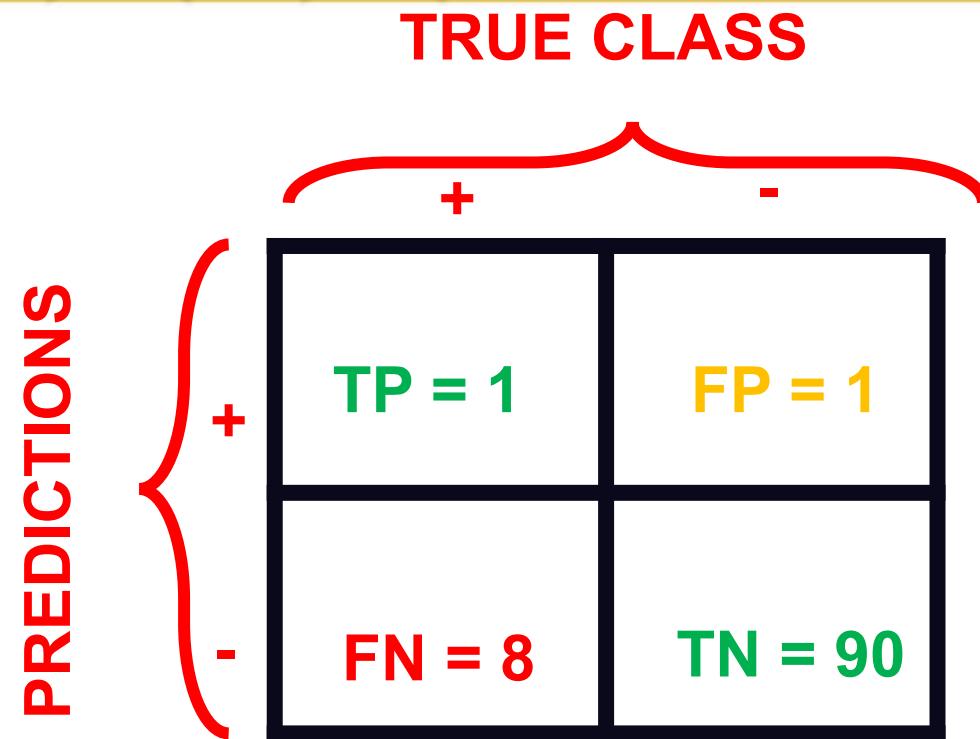
$$Precision = \frac{\text{TRUE POSITIVES}}{\text{TOTAL TRUE PREDICTIONS}}$$

$$Precision = \frac{\text{TRUE POSITIVES}}{\text{TRUE POSITIVES} + \text{FALSE POSITIVES}}$$

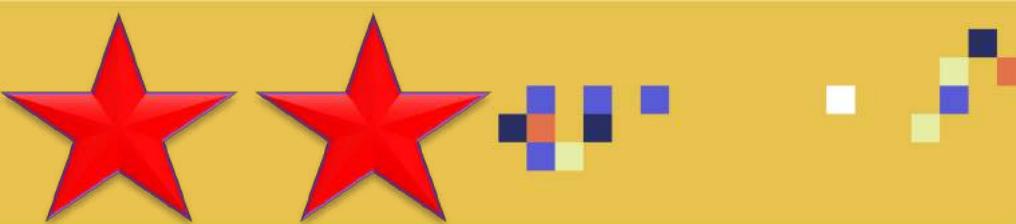
$$Precision = \frac{1}{1+1} = 50\%$$

NOTES:

- Precision is a measure of Correct Positives, in this example, the model predicted two patients were positive classes (has cancer), only one of the two was correct.
- Precision is an important metric when False positives are important (how many times a model says a pedestrian was detected and there was nothing there!)
- Examples include self driving cars and drug testing



RECALL DEEP DIVE



$$Recall = \frac{TRUE\ POSITIVES}{ACTUAL\ TRUE}$$

$$Recall = \frac{TRUE\ POSITIVES}{TRUE\ POSITIVES + FALSE\ NEGATIVES}$$

$$Recall = \frac{1}{1+8} = 11\%$$

PREDICTIONS

TRUE CLASS

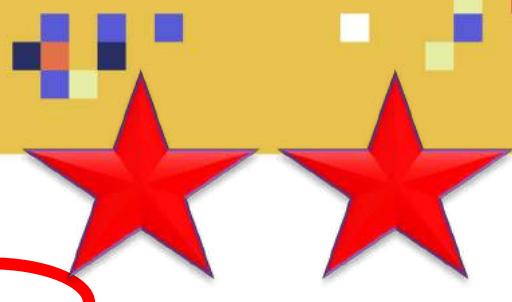
		+	-
+	TP = 1	FP = 1	
-	FN = 8	TN = 90	

NOTES:

- Recall is also called True Positive rate or sensitivity
- In this example, I had 9 cancer patients but the model only detected 1 of them
- Important metric when we care about false negatives
- Example: Self driving cars and fraud detection



EX1: BANK FRAUD DETECTION



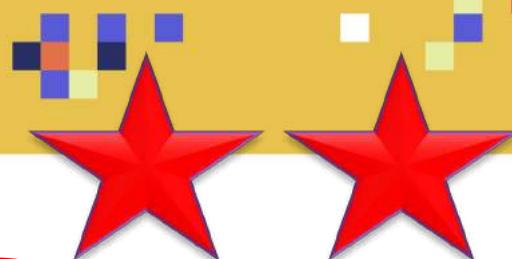
		TRUE CLASS	
		+	-
PREDICTIONS	+	THERE WAS FRAUD AND MODEL PREDICTED FRAUD	THERE WAS NO FRAUD AND MODEL PREDICTED FRAUD
	-	THERE WAS FRAUD AND MODEL PREDICTED NO FRAUD	THERE WAS NO FRAUD AND MODEL PREDICTED NO FRAUD

"This is the only case the bank loses money so bank cares about recall"

BANK LOSES MONEY

PISSED OFF CUSTOMER BUT THE BANK IS OK!

EX2: SPAM EMAIL DETECTION



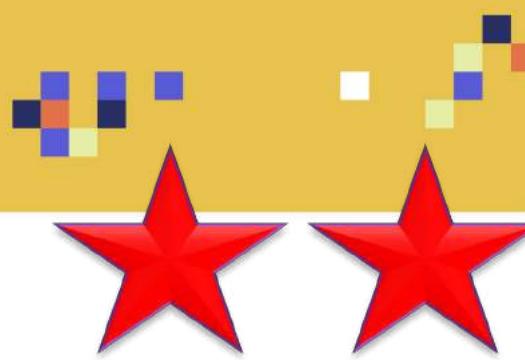
"This is a case when we care about precision and it's OK if we mess up recall a little bit"

NOT A BIG DEAL!

		TRUE CLASS	
		+	-
PREDICTIONS	+	THERE WAS SPAM EMAIL AND MODEL PREDICTED SPAM (BLOCKED IT)	THERE WAS NO SPAM EMAIL AND MODEL PREDICTED SPAM (BLOCKED IT)
	-	THERE WAS A SPAM EMAIL AND MODEL PREDICTED NO SPAM (WENT TO INBOX)	THERE WAS NO SPAM EMAIL AND MODEL PREDICTED NO SPAM (WENT TO INBOX)

BLOCKED IMPORTANT EMAILS (DREAM JOB!)

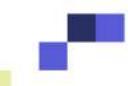
F1 SCORE



$$F1\ Score = \frac{2 * (PRECISION * RECALL)}{(PRECISION + RECALL)}$$

$$F1\ Score = \frac{2 * TP}{2 * TP + FP + FN}$$

- F1 Score is an overall measure of a model's accuracy that combines precision and recall.
- F1 score is the harmonic mean of precision and recall.
- What is the difference between F1 Score and Accuracy?
- In unbalanced datasets, if we have large number of true negatives (healthy patients), accuracy could be misleading. Therefore, F1 score might be a better KPI to use since it provides a balance between recall and precision in the presence of unbalanced datasets.



SPECIFICITY

$$SPECIFICITY = TRUE\ NEGATIVE\ RATE = \frac{TRUE\ NEGATIVES}{TRUE\ NEGATIVES + FALSE\ POSITIVES}$$

$$SPECIFICITY = \frac{90}{90 + 1} = 98.9\%$$

- Specificity measures the proportion of actual negatives that are correctly identified as such
- Example: true negative rate indicates the percentage of healthy people who are correctly classified as healthy!

PREDICTIONS

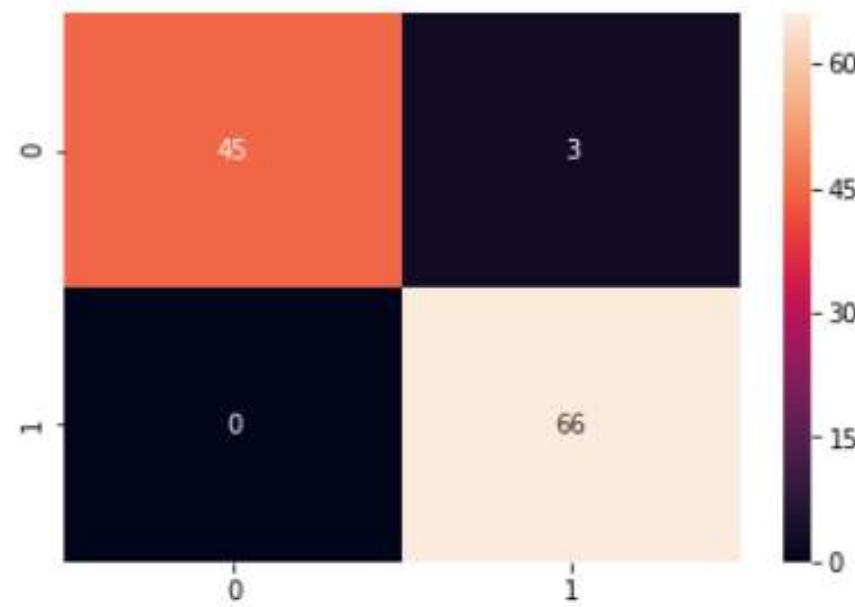
		TRUE CLASS	
		+	-
+	+	TP = 1	FP = 1
	-	FN = 8	TN = 90

F1-SCORE PER CLASS: CANCER CLASSIFICATION DATASET

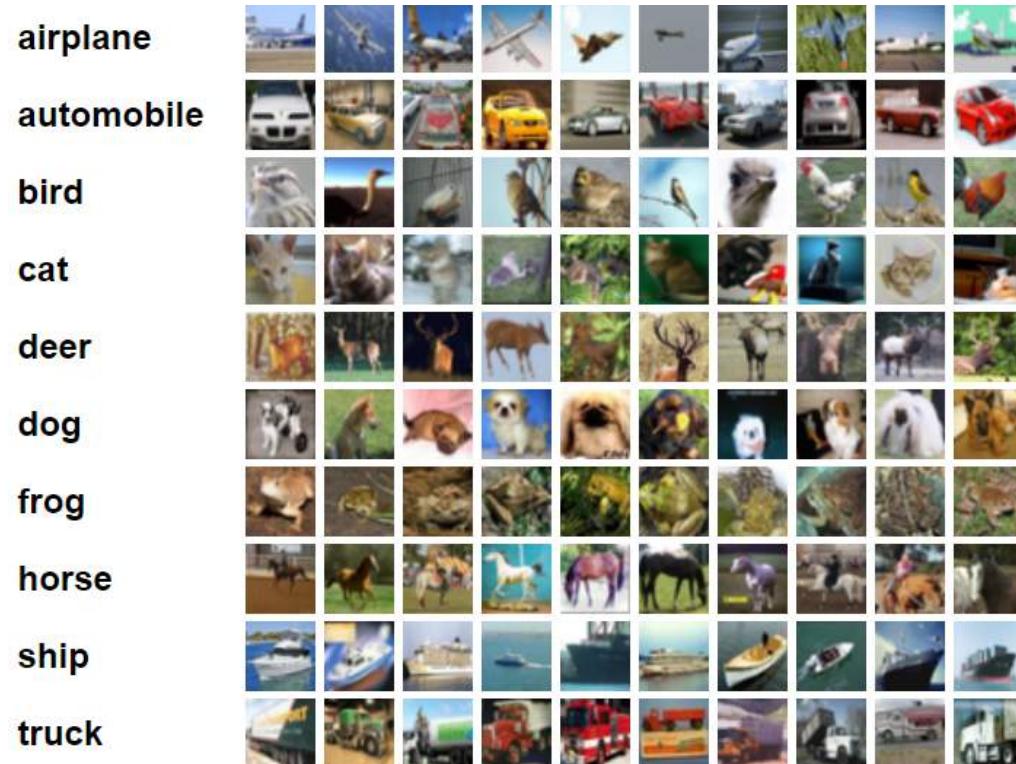
	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry	mean fractal dimension	...	worst texture	worst perimeter	worst area	worst smoothness	worst comp
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419	0.07871	...	17.33	184.60	2019.0	0.1622	
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812	0.05667	...	23.41	158.80	1956.0	0.1238	
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069	0.05999	...	25.53	152.50	1709.0	0.1444	
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597	0.09744	...	26.50	98.87	567.7	0.2098	
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809	0.05883	...	16.67	152.20	1575.0	0.1374	

5 rows × 31 columns

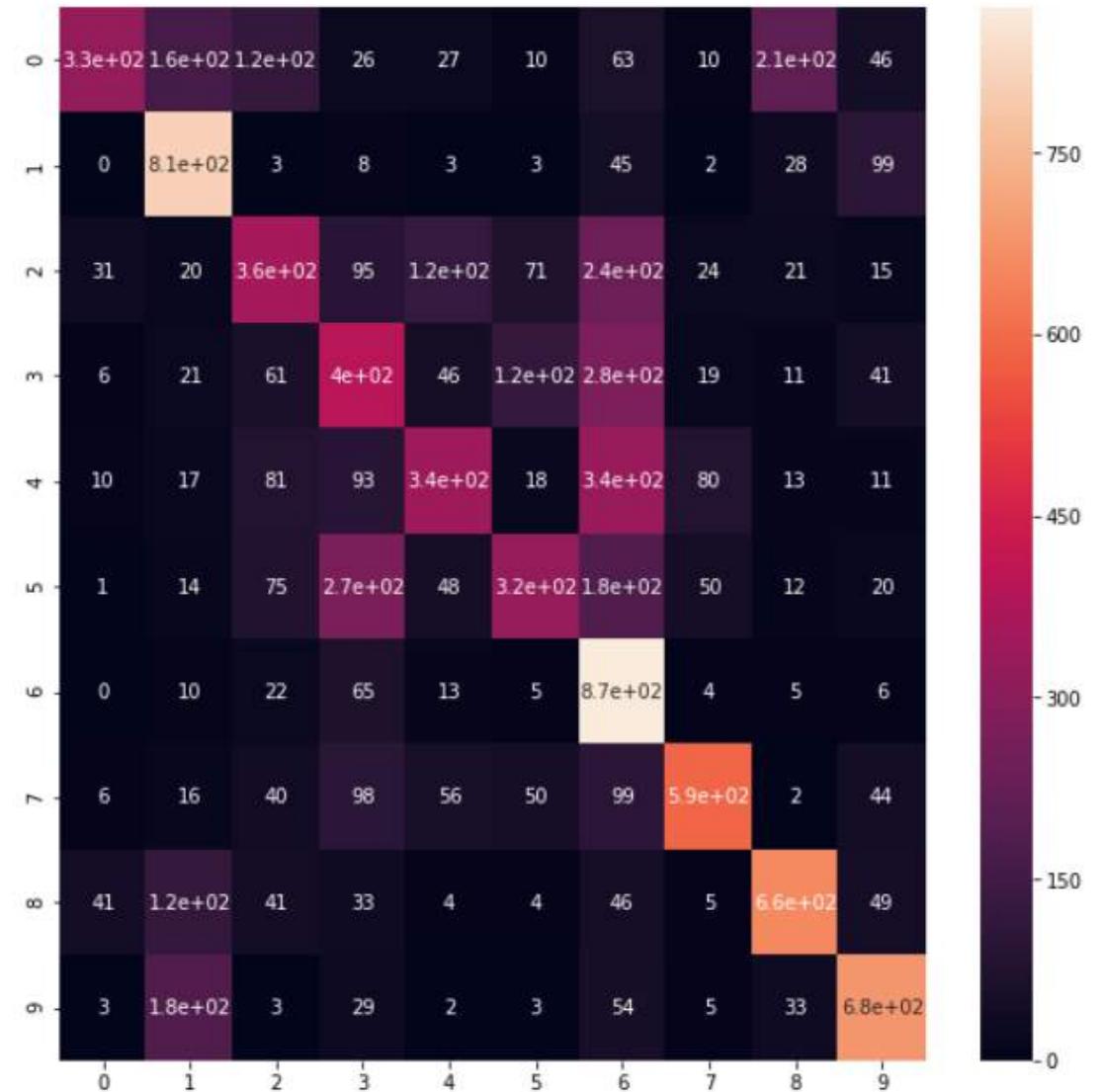
	precision	recall	f1-score
0.0	1.00	0.94	0.97
1.0	0.96	1.00	0.98
avg / total	0.97	0.97	0.97



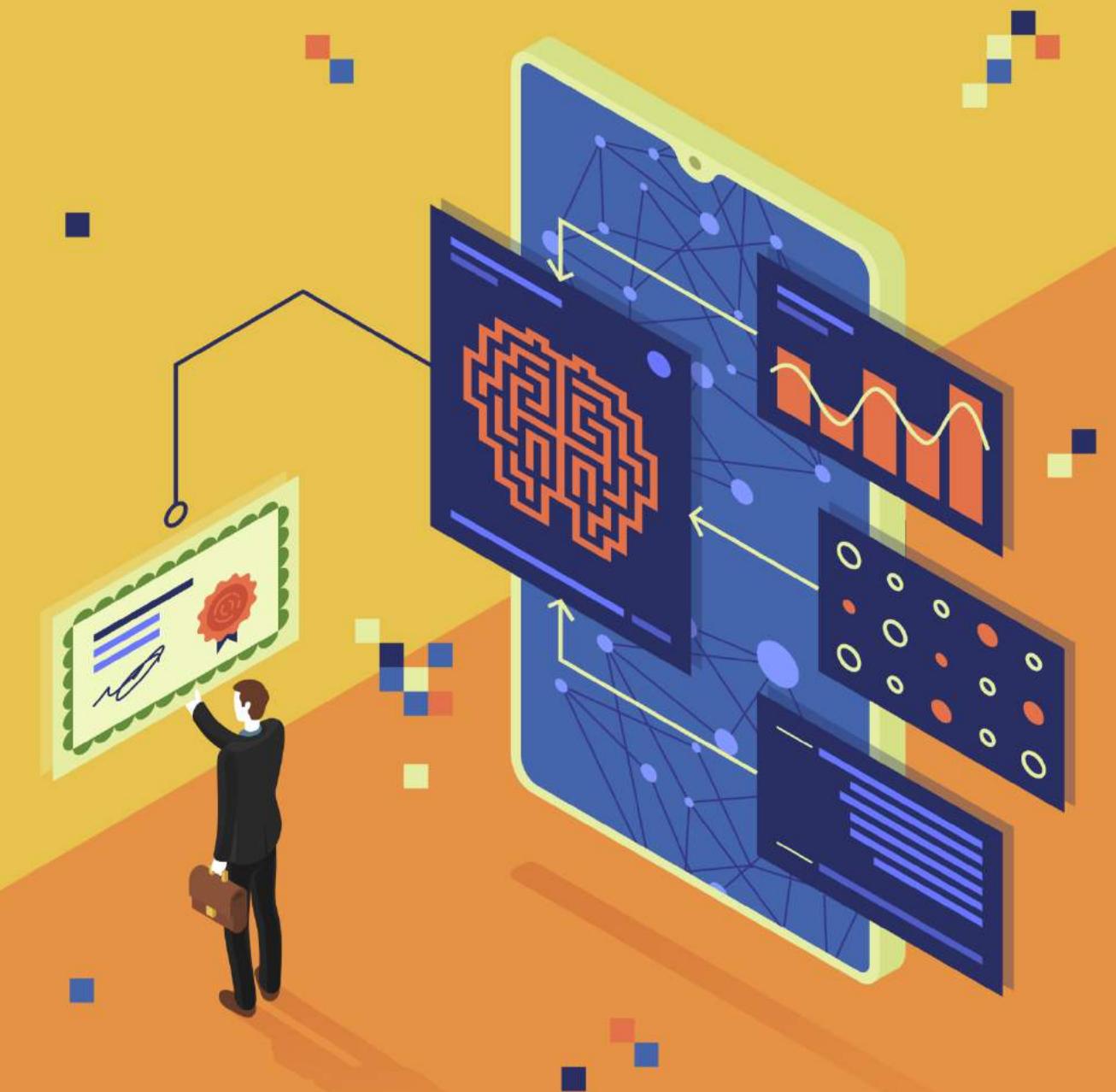
MULTICLASS CLASSIFICATION



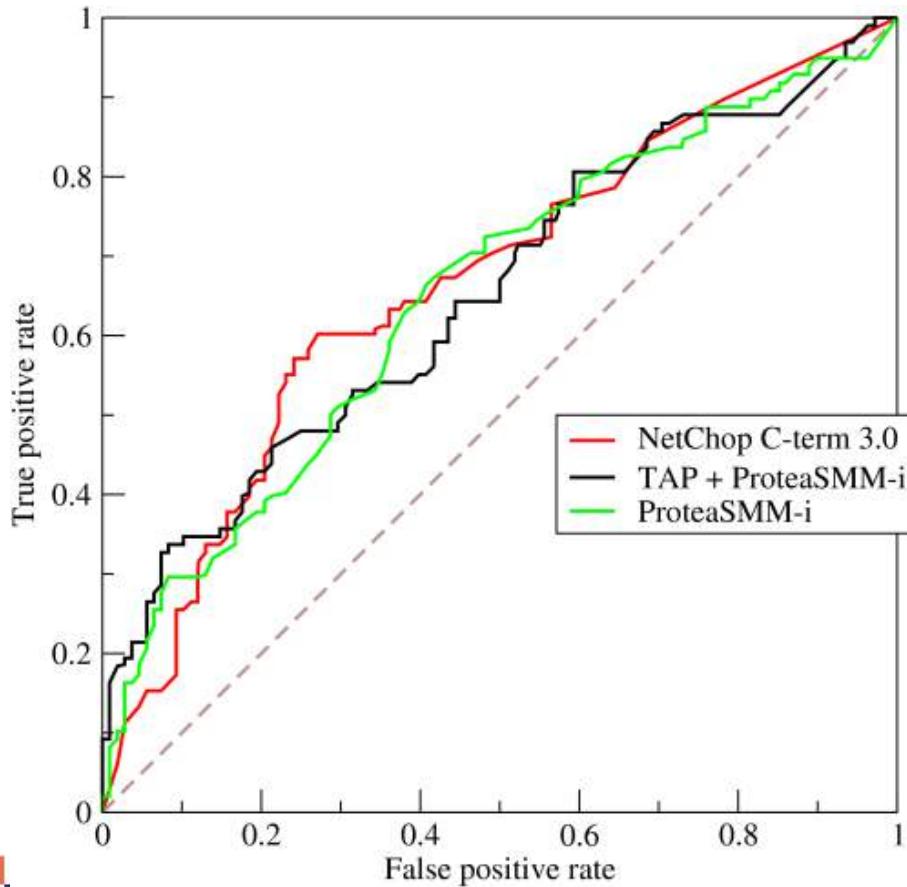
<https://www.cs.toronto.edu/~kriz/cifar.html>



MODEL PERFORMANCE ASSESSMENT – ROC, AUC, HEATMAP, RMSE

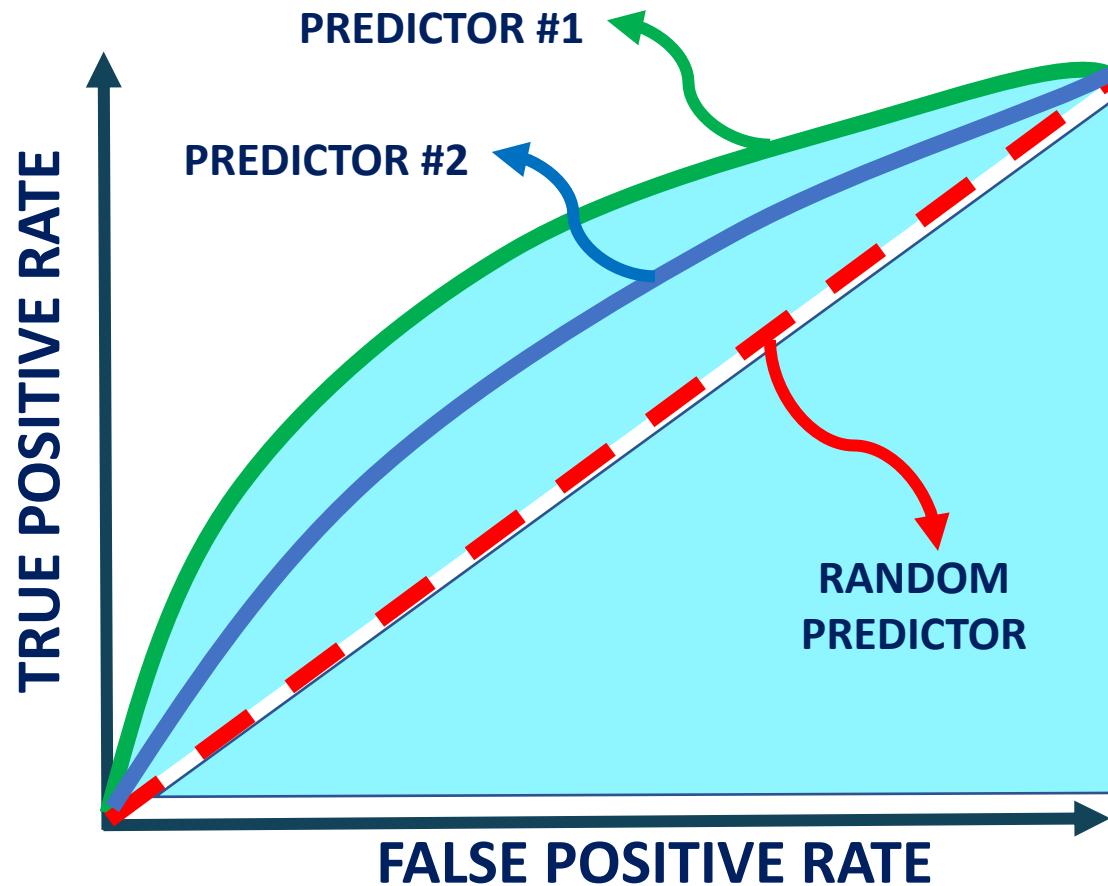


ROC (RECEIVER OPERATING CHARACTERISTIC CURVE)



- ROC Curve is a metric that assesses the model ability to distinguish between binary (0 or 1) classes.
- The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.
- The true-positive rate is also known as sensitivity, recall or probability of detection in machine learning.
- The false-positive rate is also known as the probability of false alarm and can be calculated as $(1 - \text{specificity})$.
- Points above the diagonal line represent good classification (better than random)
- The model performance improves if it becomes skewed towards the upper left corner.

AUC (AREA UNDER CURVE)



- The light blue area represents the area Under the Curve of the Receiver Operating Characteristic (AUROC).
- The diagonal dashed red line represents the ROC curve of a random predictor with AUROC of 0.5.
- If ROC AUC = 1, perfect classifier
- Predictor #1 is better than predictor #2
- Higher the AUC, the better the model is at predicting 0s as 0s and 1s as 1s.



CONFUSION MATRIX AND HEAT MAP



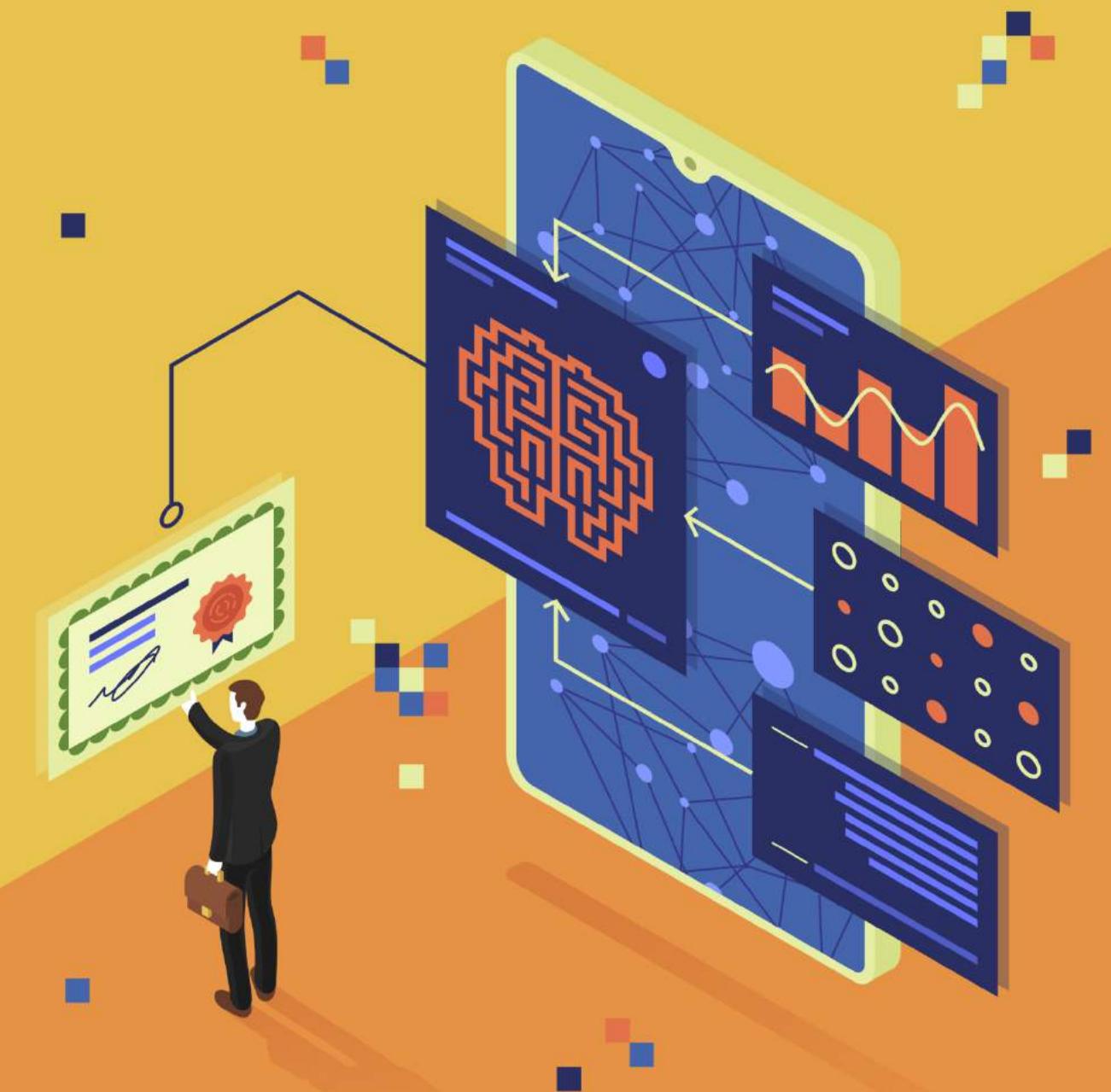
REGRESSION METRICS: ROOT MEAN SQUARE ERROR (RMSE)

- Root Mean Square Error (RMSE) represents the **standard deviation of the residuals** (i.e.: differences between the model predictions and the true values (training data)).
- RMSE can be **easily interpreted** compared to MSE because RMSE units match the units of the output.
- RMSE provides an estimate of how large the residuals are being dispersed.
- The RMSE is calculated as follows:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

- RMSE is calculated by following these steps:
 1. Calculate the residual for every data point
 2. Calculate the squared value of the residuals
 3. Calculate the average of the squared residuals
 4. Obtain the square root of the result

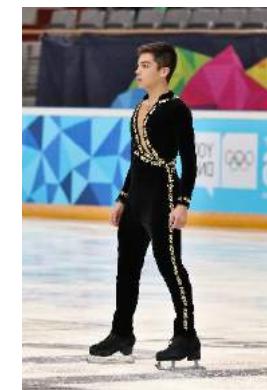
TRANSFER LEARNING BASICS



WHAT IS TRANSFER LEARNING?



- Transfer learning is a machine learning technique in which a network that has been trained to perform a specific task is being **reused (repurposed)** as a starting point for another similar task.
- Transfer learning is widely used since starting from a pre-trained models can dramatically **reduce the computational time** required if training is performed from scratch.



KNOWLEDGE TRANSFER



Photo Credit: https://commons.wikimedia.org/wiki/File:Lillehammer_2016_-_Figure_Skating_Men_Short_Program_-_Camden_Pulkkinen_2.jpg

Photo Credit: https://commons.wikimedia.org/wiki/Alpine_skiing#/media/File:Andrej_%C5%A0oporn_at_the_2010_Winter_Olympic_downhill.jpg

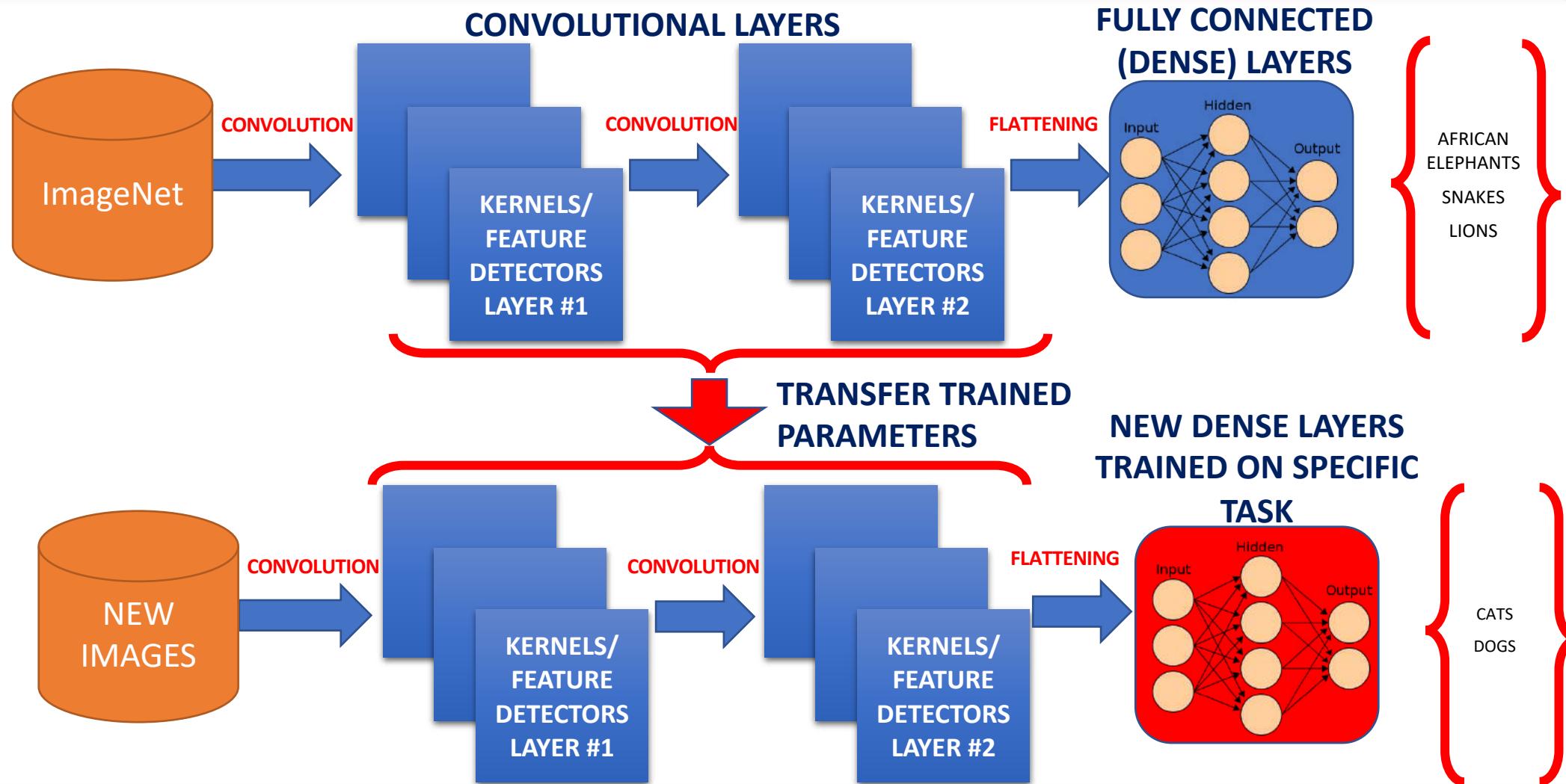
Citations: Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei.

ImageNet Large Scale Visual Recognition Challenge. arXiv:1409.0575, 2014.

WHAT IS TRANSFER LEARNING

- “Transfer learning is the improvement of learning in a new task through the **transfer of knowledge** from a related task that has already been learned”—Transfer Learning, Handbook of Research on Machine Learning Applications, 2009.
- In transfer learning, a **base (reference)** Artificial Neural Network on a base dataset and function is being trained. Then, this trained network weights are then **repurposed in a second ANN** to be trained on a new dataset and function.
- Transfer learning works great if the **features are general**, such that trained weights can effectively repurposed.
- Intelligence is being transferred from the base network to the newly target network.

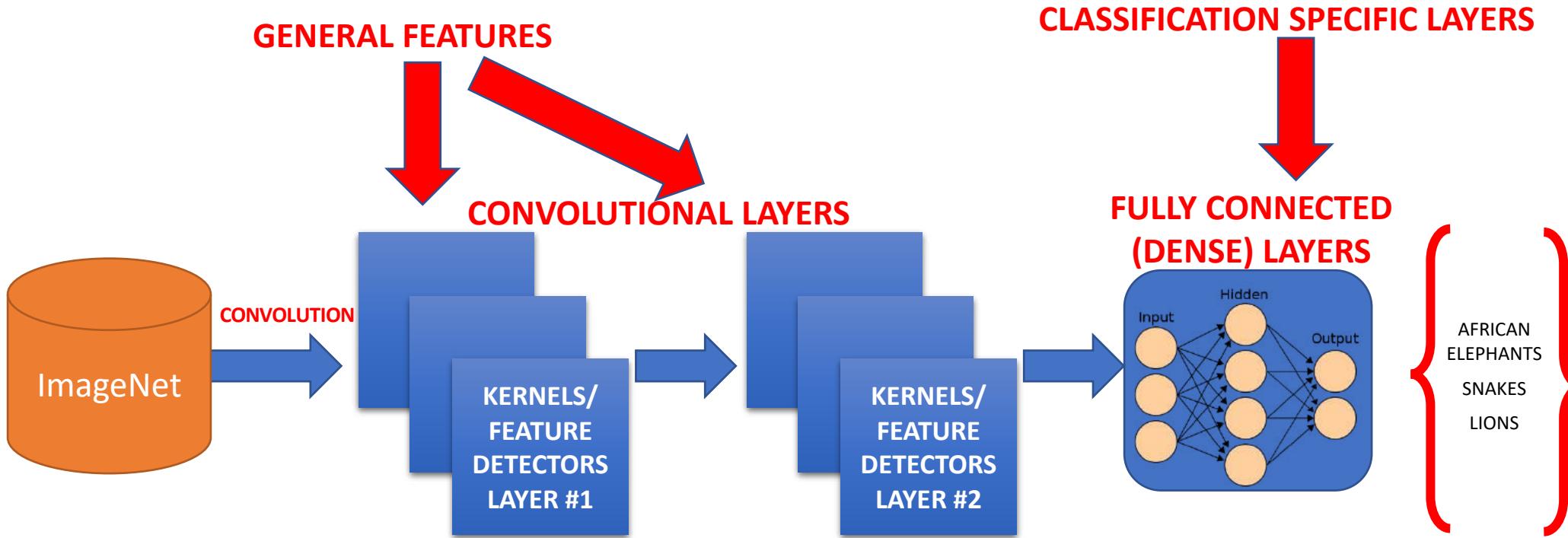
TRANSFER LEARNING PROCESS



WHY DO WE KEEP THE FIRST LAYERS?



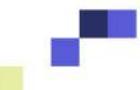
- The first CNN layers are used to **extract high level general features**.
- The last couple of layers are used to perform classification (on a specific task).
- So we copy the first trained layers (**base model**) and then we **add a new custom layers** in the output to perform classification on a specific new task.



TRANSFER LEARNING TRAINING STRATEGIES



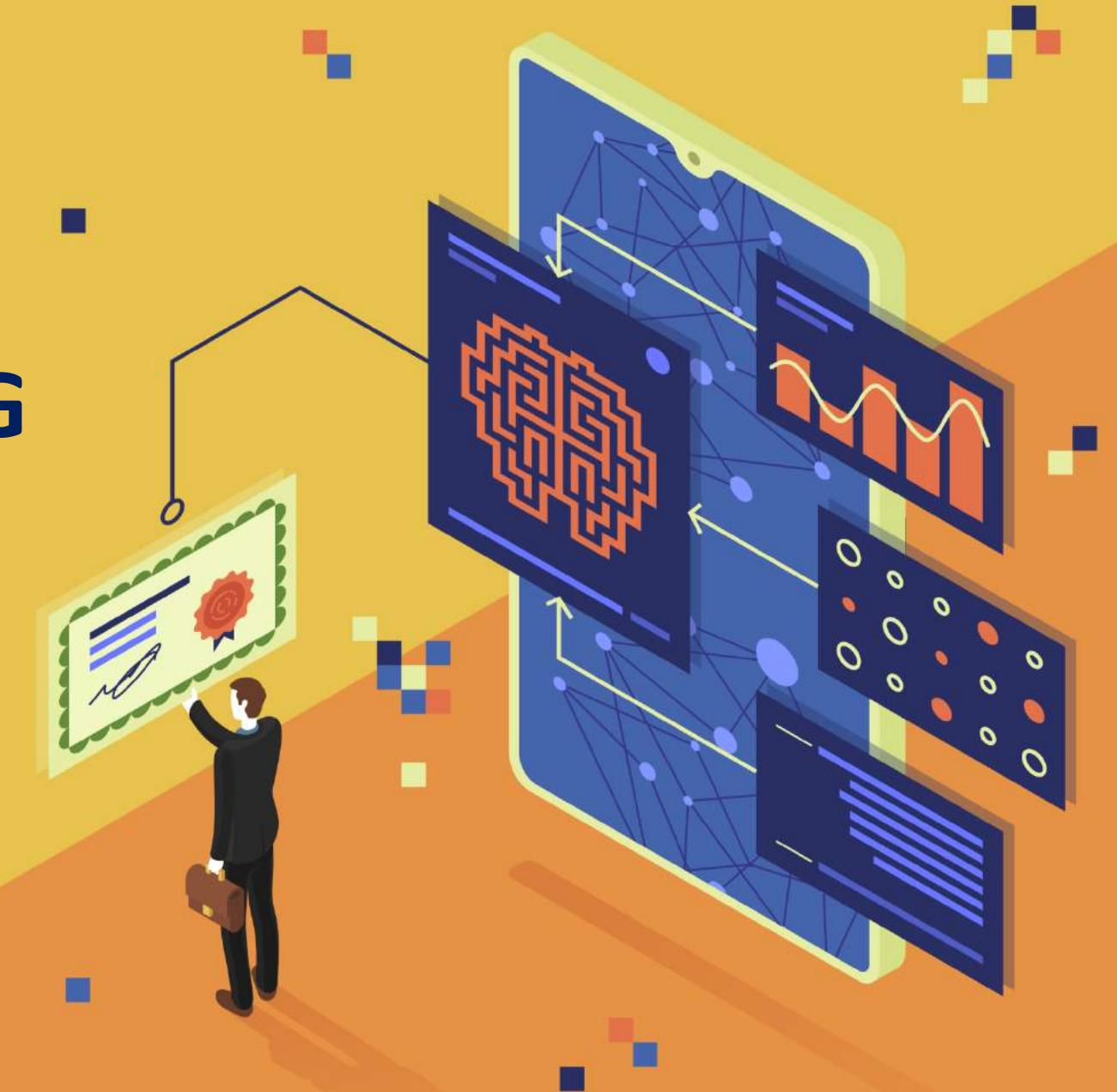
- **Strategy #1 Steps:**
 - Freeze the trained CNN network weights from the first layers.
 - Only train the newly added dense layers (with randomly initialized weights).
- **Strategy #2 Steps:**
 - Initialize the CNN network with the pre-trained weights
 - Retrain the **entire CNN** network while setting the learning rate to be very small, this is critical to ensure that you do not aggressively change the trained weights.



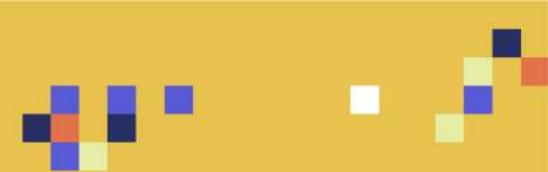
TRANSFER LEARNING ADVANTAGES

- Transfer learning advantages are:
 - Provides fast training progress, you don't have to start from scratch using randomly initialized weights
 - You can use small training dataset to achieve incredible results
- When to use transfer learning?
 - When there is a small training dataset available for your new task but there exist a large dataset in a different domain (such as ImageNet)
 - When you have limited computational resources (GPU, TPU)

ENSEMBLE LEARNING METHODS



ENSEMBLE LEARNING



- Ensemble techniques such as bagging and boosting can offer an extremely powerful algorithm by combining a group of relatively weak/average ones.
- For example, you can combine several decision trees to create a powerful random forest algorithm
- By Combining votes from a pool of experts, each will bring their own experience and background to solve the problem resulting in a better outcome.
- Bagging and Boosting can reduce variance and overfitting and increase the model robustness.
- Example: Blind men and the elephant!

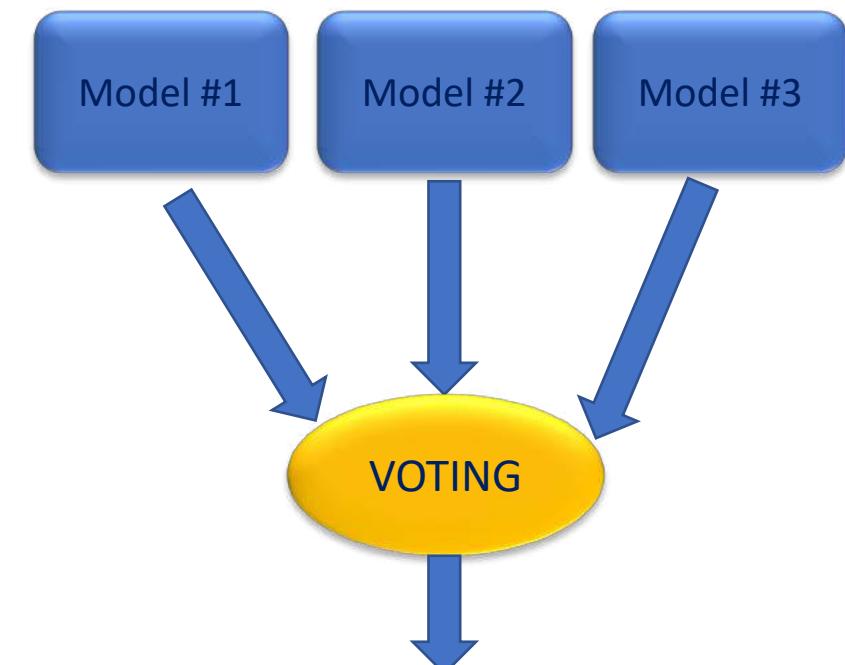
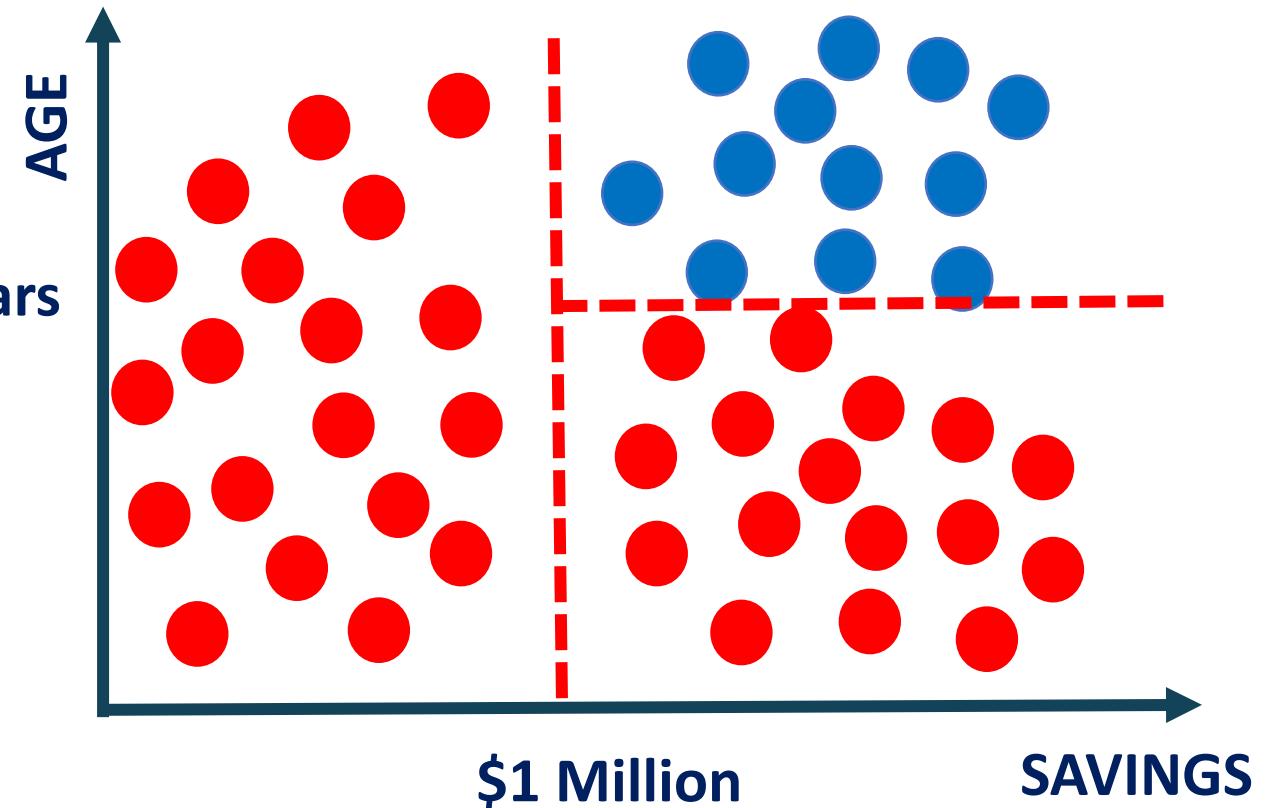
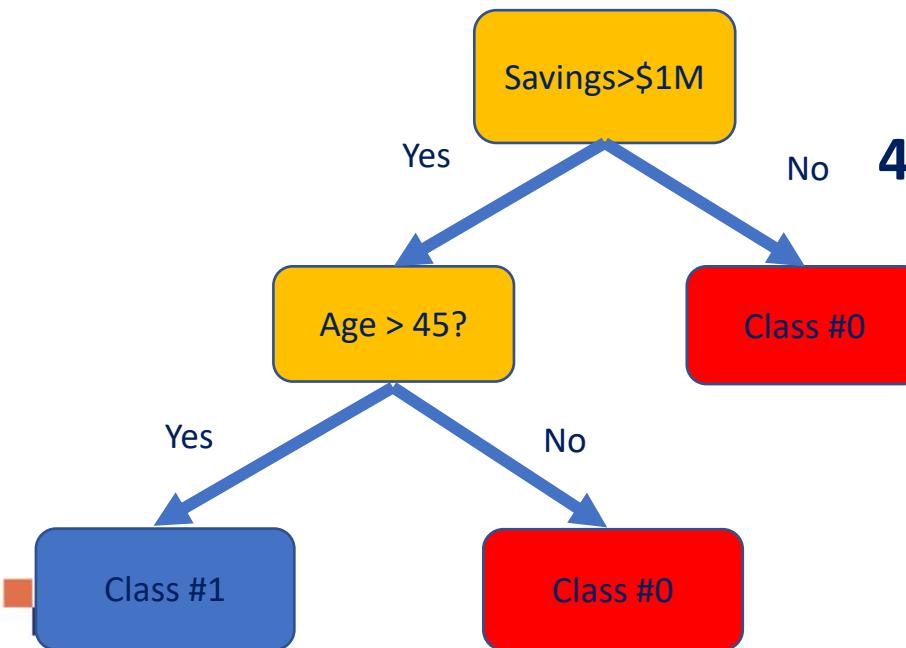


Photo Credit: https://commons.wikimedia.org/wiki/File:Blind_men_and_elephant.png

DECISION TREES: INTUITION



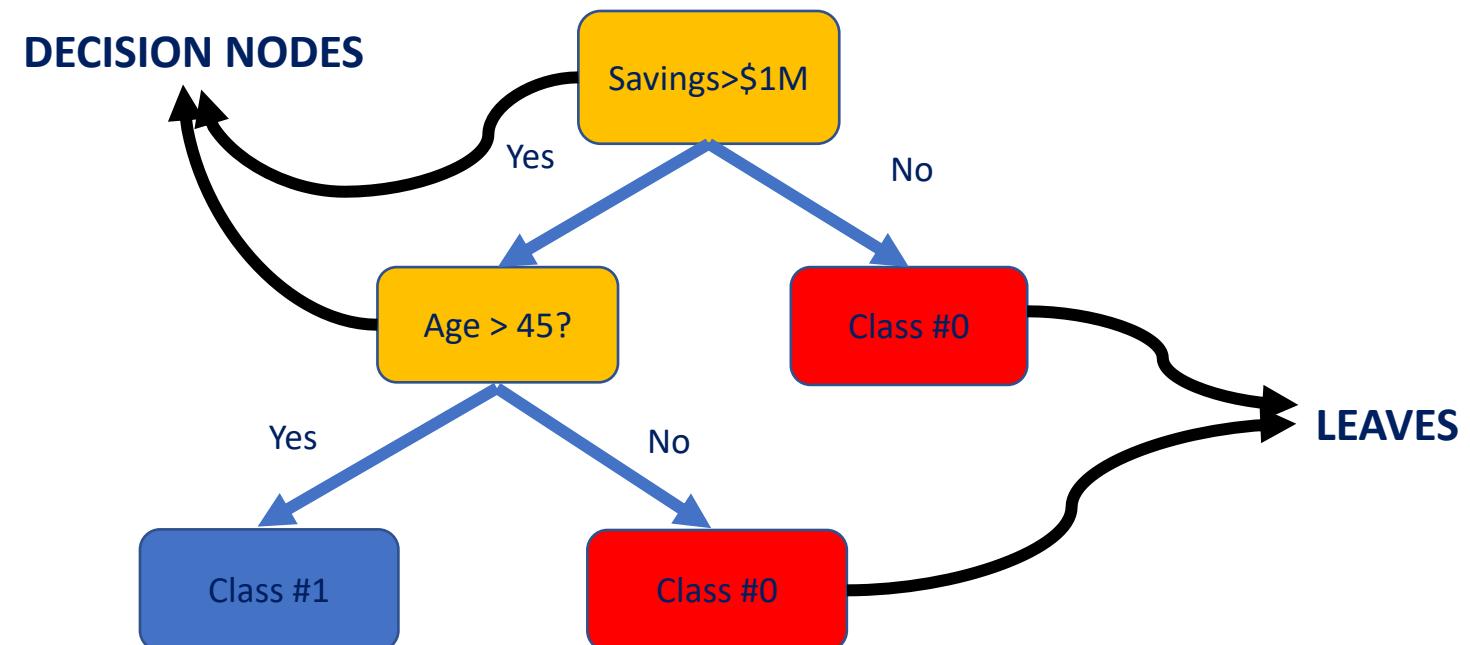
- Decision Trees are supervised Machine Learning technique where the data is split according to a certain condition/parameter.
- Let's assume we want to classify whether a customer could retire or not based on their savings and age.



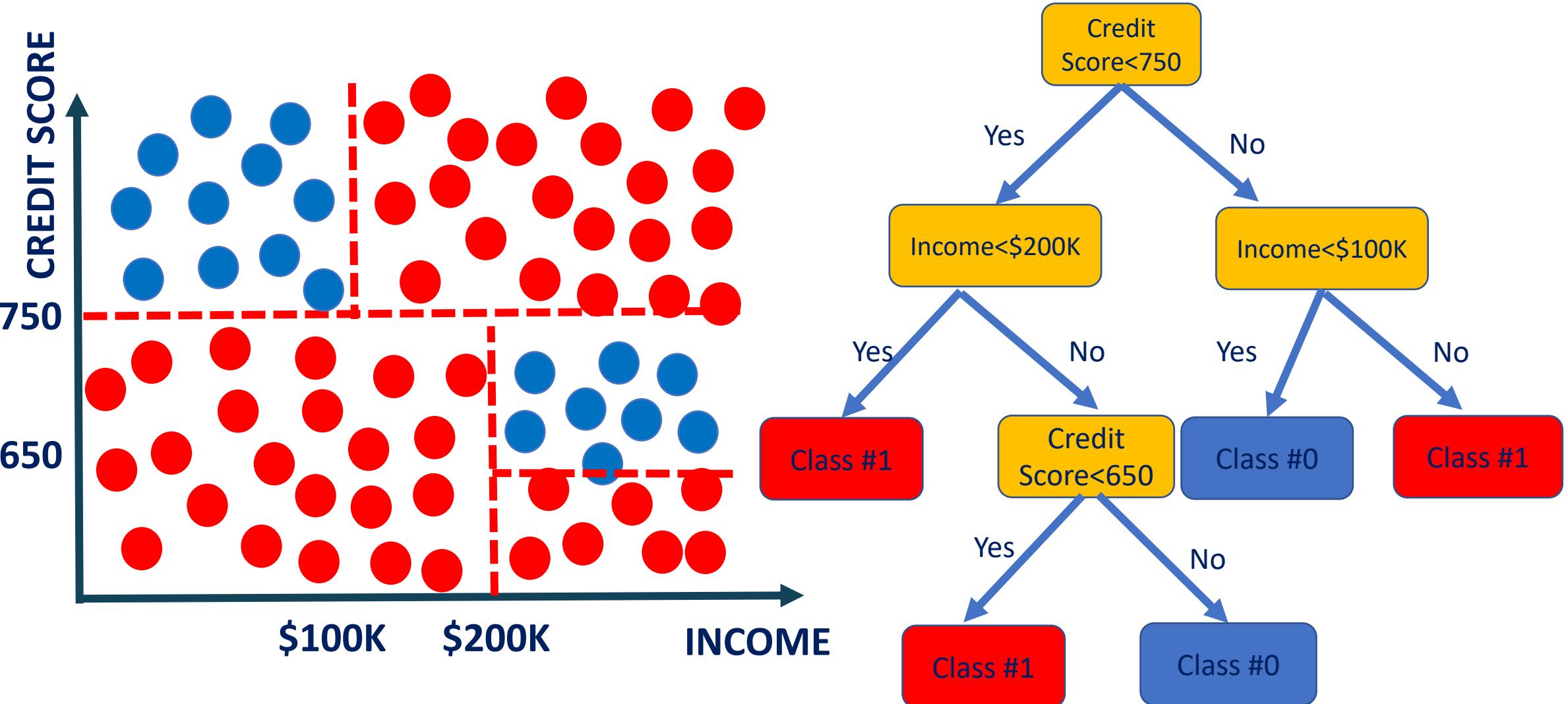
DECISION TREES: DEFINITIONS



- The tree consists of **decision nodes** and **leaves**.
- Leaves are the decisions or the final outcomes.
- Decision nodes are where the data is split based on a certain attribute.
- Objective is to minimize the entropy which provides the optimum split



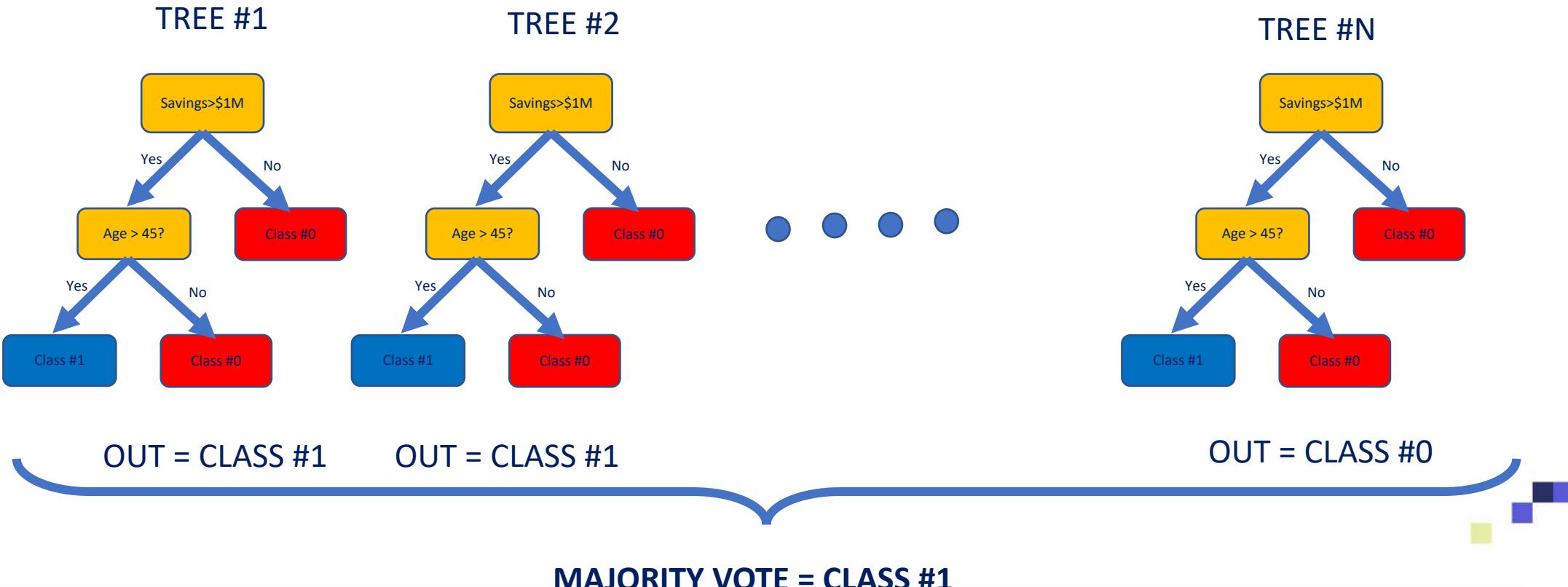
DECISION TREES: CUSTOMER SEGMENTATION



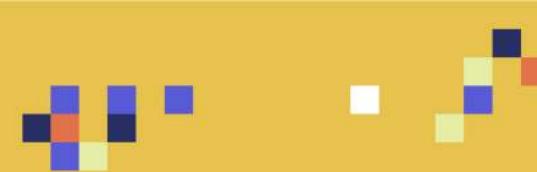
RANDOM FOREST: INTUITION



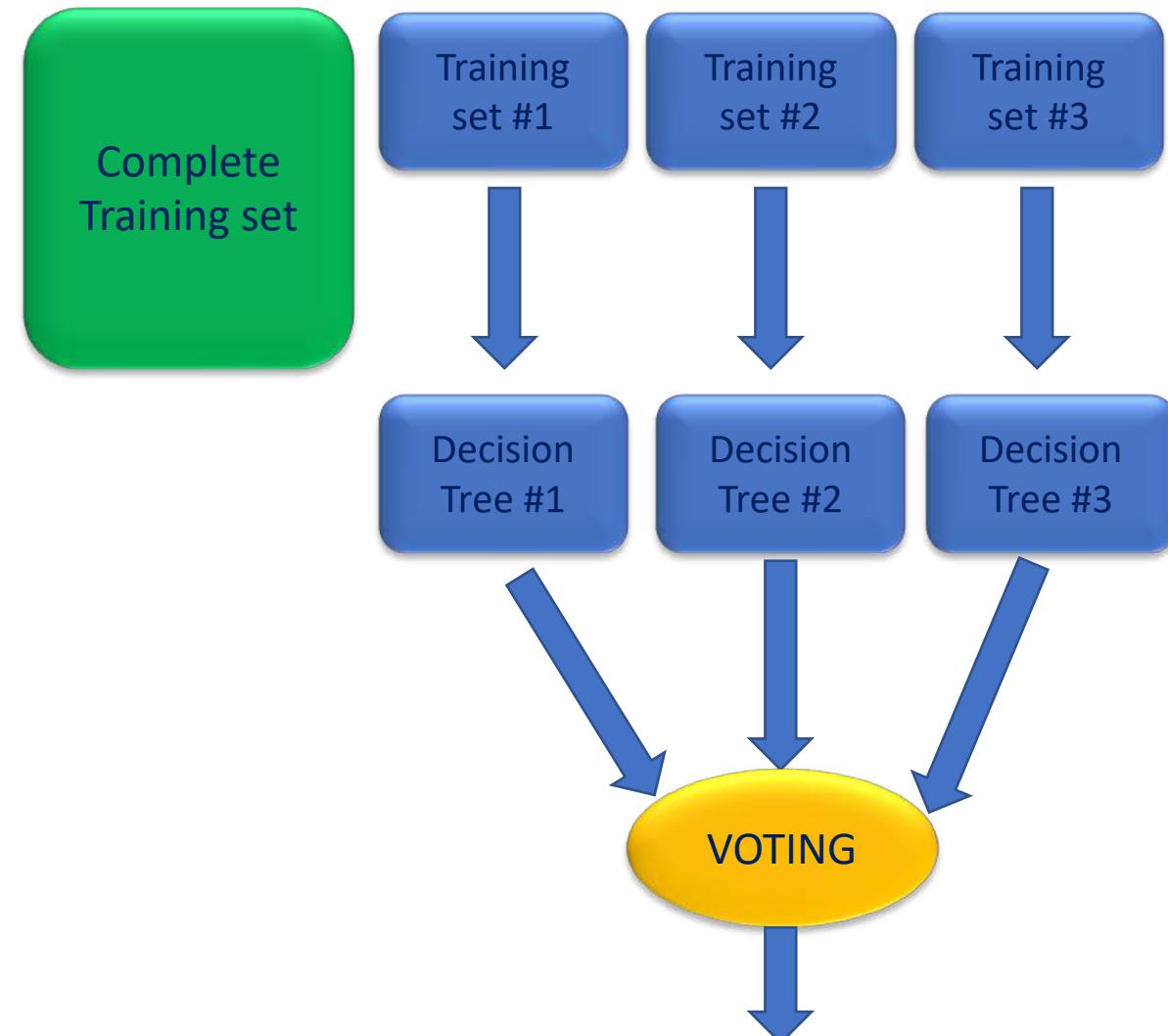
- Random Forest Classifier is a type of **ensemble algorithm**.
- It creates a set of decision trees from randomly selected subset of training set.
- It then **combines votes** from different decision trees to decide the final class of the test object.



RANDOM FOREST: WHY AND HOW?



- It overcomes the issues with single decision trees by reducing the effect of noise.
- Overcomes **overfitting problem** by taking **average of all the predictions**, canceling out biases.
- Suppose training set: $[X_1, X_2, X_3, X_4]$ with labels: $[L_1, L_2, L_3, L_4]$
- Random forest creates three decision trees taking inputs as follows:
 - $[X_1, X_2, X_3], [X_1, X_2, X_4], [X_2, X_3, X_4]$
- Example: Combining votes from a pool of experts, each will bring their own experience and background to solve the problem resulting in a better outcome.

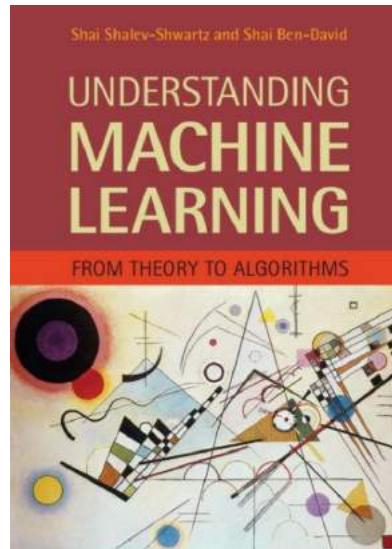


RANDOM FOREST: ADDITIONAL READING MATERIAL



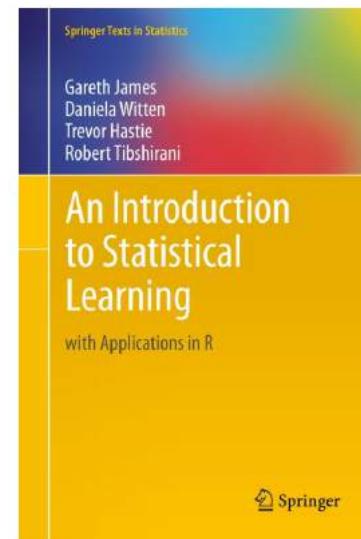
Additional Resources, Page #255:

<http://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/understanding-machine-learning-theory-algorithms.pdf>

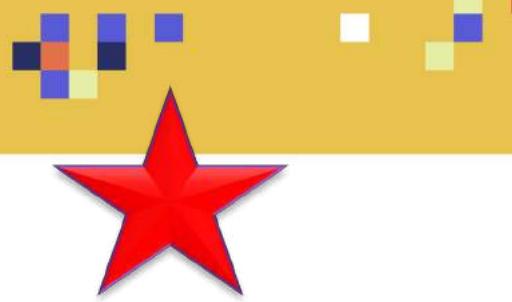


Additional Resources, Page #320:

<http://www-bcf.usc.edu/~gareth/ISL/ISLR%20Seventh%20Printing.pdf>



BAGGING – INDEPENDANTLY AND IN PARALLEL!



- Bagging is a powerful ensemble algorithm in which N new training datasets are generated from a given “main” pool of data.
- Dataset is sampled with replacement.
- Several models are being trained in parallel.
- Prediction based on the aggregate predictions of all models is made.
- Bagging avoids overfitting

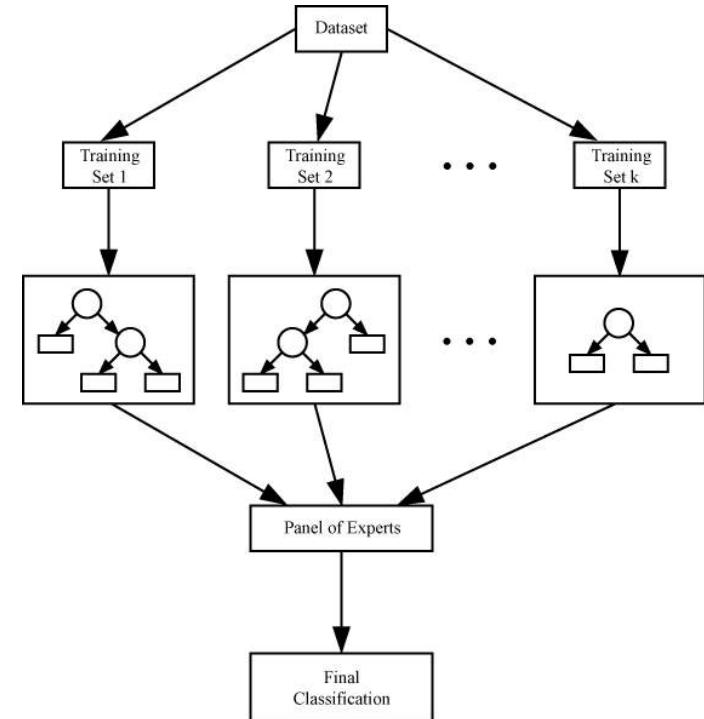
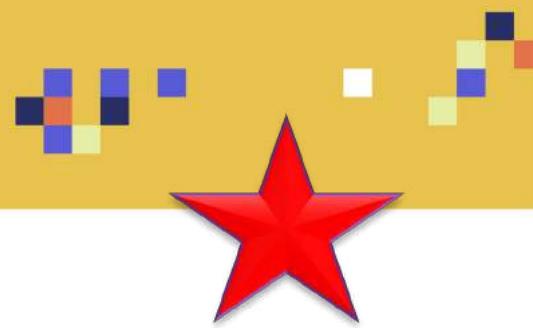


Photo Credit: https://commons.wikimedia.org/wiki/File:DTE_Bagging.png

BOOSTING – SEQUENTIALLY



- Boosting is an ensemble algorithm that combines weighted averages to turn a weak model into an extremely powerful one!
- Boosting differs from bagging in which various models work cooperatively to achieve the best results. The output from the first model is reweighted and fed to another model.
- Recall that in bagging, each model ran independently and then we combined the output at the end!
- XGBoost is an extremely powerful algorithm that won several Kaggle competitions recently.
- Boosting could potentially lead to a better accuracy compared to bagging.

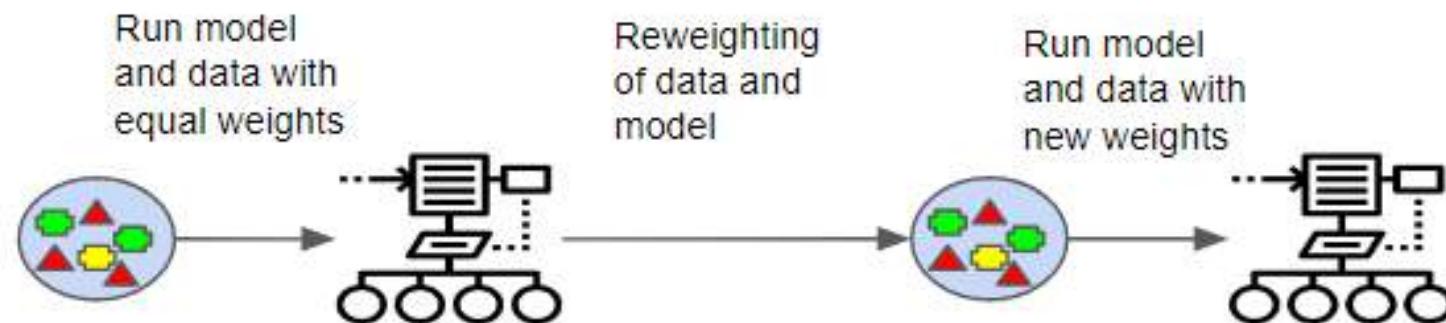
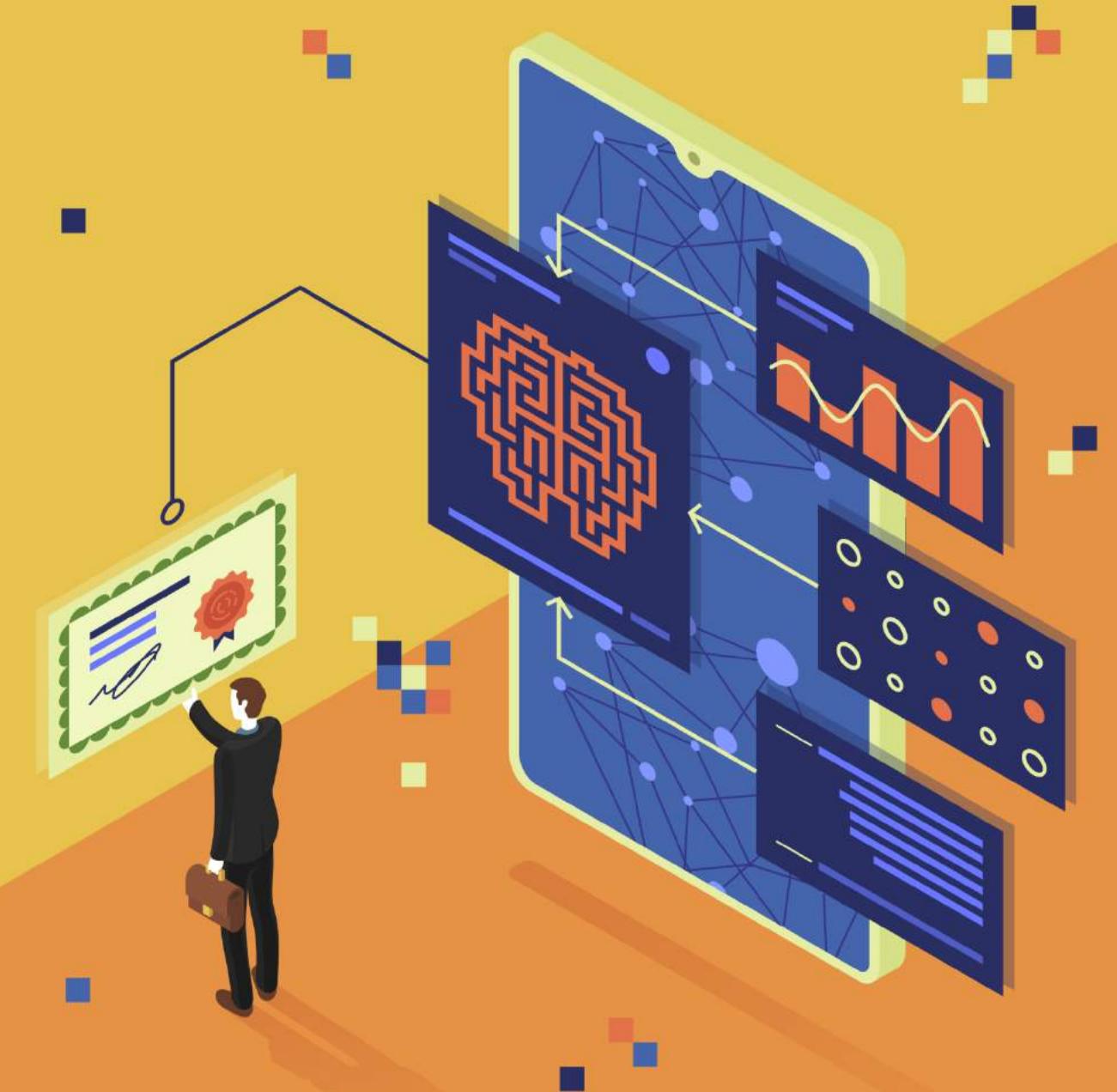
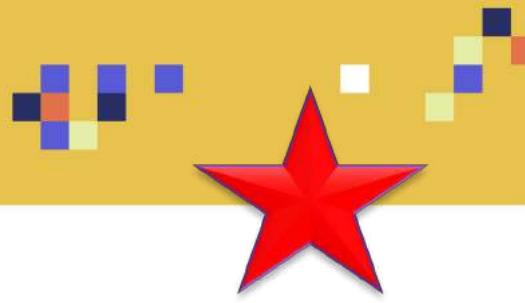


Photo Credit: <https://commons.wikimedia.org/wiki/File:Boosting.png>

K-FOLD CROSS VALIDATION



K-FOLD CROSS VALIDATION



- Cross-validation is used to assess the generalization capability of the model (with data that has never been seen before).
- K-fold cross validation is a technique used to assess the performance of machine learning models.
- K-Fold works by randomly dividing the data into equal sized K groups (folds).
- First fold is considered the validation dataset, and the model is trained on the remaining $k - 1$ folds.
- By comparing the performance of the model with these many folds, you can now know if the model is over fitted or not.
- If the error is comparable across various runs then the model is generalized.

Model1: Trained on Fold1 + Fold2 and Tested on Fold3

Model2: Trained on Fold2 + Fold3 and Tested on Fold1

Model3: Trained on Fold1 + Fold3 and Tested on Fold2

AWS MACHINE LEARNING CERTIFICATION



DOMAIN #3: MODELING (36% EXAM)



AWS ML CERTIFICATION EXAM DOMAINS



Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #3 OVERVIEW:

SECTION #8: MACHINE AND DEEP LEARNING BASICS – PART #1

- Artificial Neural Networks Basics: Single Neuron Model
- Activation Functions
- Multi-Layer Perceptron Model
- How do Artificial Neural Networks Train?
- ANN Parameters Tuning – Learning rate and batch size
- Tensorflow playground
- Gradient Descent and Backpropagation
- Overfitting and Under fitting
- How to overcome overfitting?
- Bias Variance Trade-off
- L1 Regularization
- L2 Regularization

SECTION #9: MACHINE AND DEEP LEARNING BASICS – PART #2

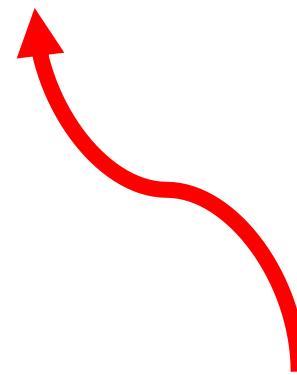
- Artificial Neural Networks Architectures
- Convolutional Neural Networks
- Recurrent Neural Networks
- Vanishing Gradient Problem
- LSTM Networks
- Model Performance Assessment – Confusion Matrix
- Model Performance Assessment – Precision, recall, F1-score
- Model Performance Assessment – ROC, AUC, Heatmap, and RMSE
- K-Fold Cross validation
- Transfer Learning
- Ensemble Learning – Bagging and Boosting

DOMAIN #3 OVERVIEW:



SECTION #10: MACHINE AND DEEP LEARNING IN AWS – BUILT-IN ALGORITHMS PART #1

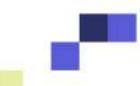
- AWS SageMaker
- Deep Learning on AWS
- SageMaker Built-in algorithms
- Object Detection
- Image Classification
- Semantic Segmentation
- SageMaker Linear Learner
- Factorization Machines
- XG-Boost
- SageMaker Seq2Seq
- SageMaker DeepAR
- SageMaker Blazing Text



WE ARE HERE!

SECTION #11: MACHINE AND DEEP LEARNING IN AWS – BUILT-IN ALGORITHMS PART #2

- Object2Vec
- Random Cut Forest
- Neural Topic Model
- LDA
- K-Nearest Neighbours (KNN)
- K Means
- Principal Component Analysis (PCA)
- IP insights
- Reinforcement Learning
- Automatic Model Tuning
- SageMaker and Spark

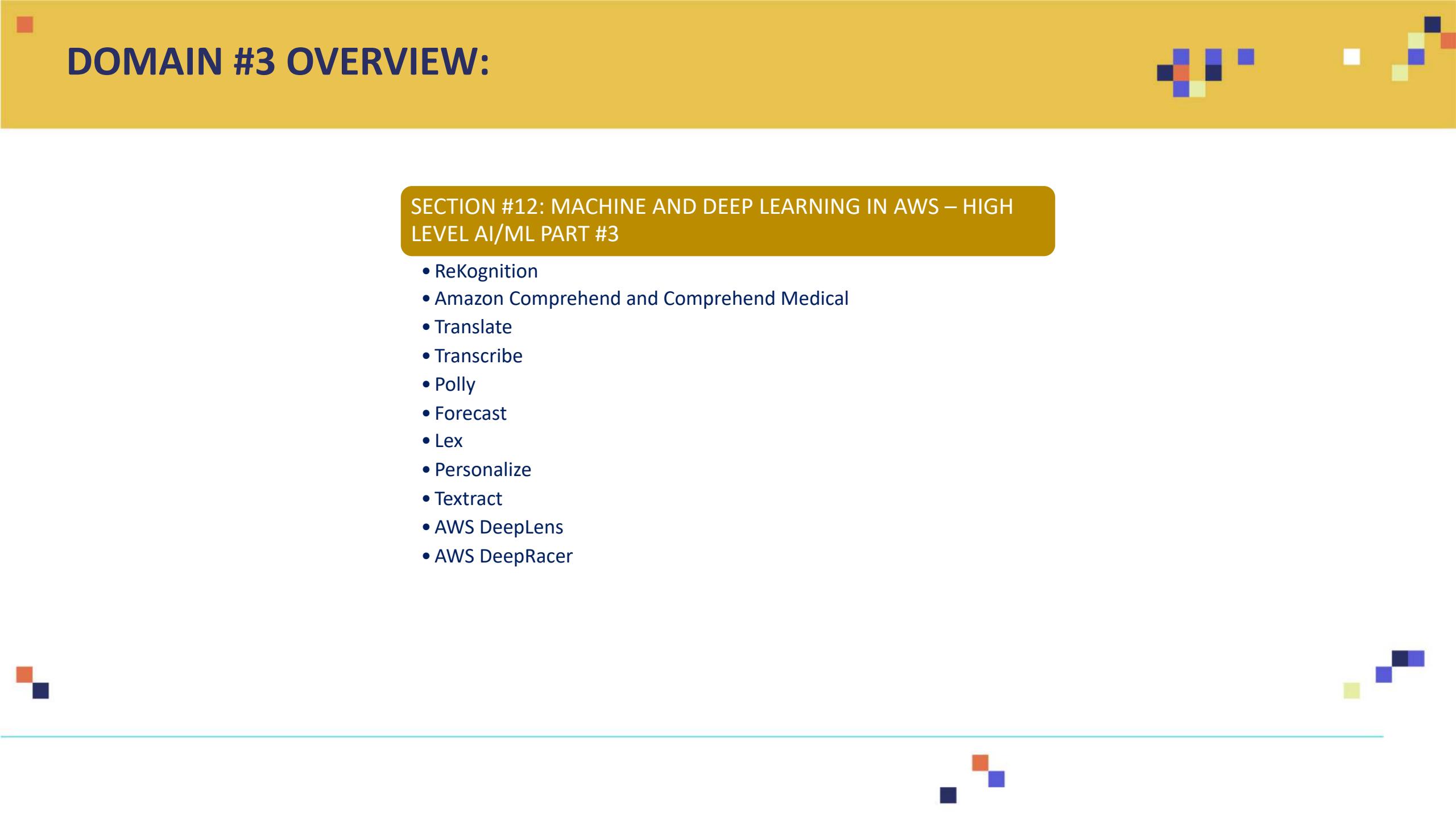


DOMAIN #3 OVERVIEW:



SECTION #12: MACHINE AND DEEP LEARNING IN AWS – HIGH LEVEL AI/ML PART #3

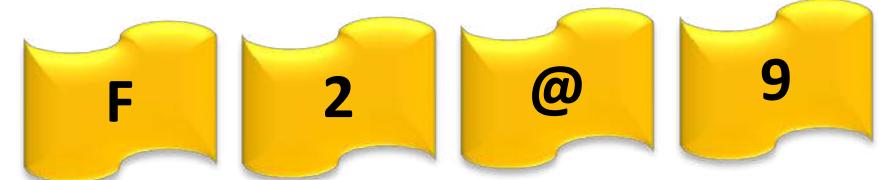
- ReKognition
- Amazon Comprehend and Comprehend Medical
- Translate
- Transcribe
- Polly
- Forecast
- Lex
- Personalize
- Textract
- AWS DeepLens
- AWS DeepRacer



RECALL OUR MINI CHALLENGE AND PRIZE!



- For those of you who will successfully complete the entire section and watch the videos till the end, they will receive a valuable prize!



AWS SAGEMAKER – PART #1



AMAZON SAGEMAKER



- Amazon SageMaker is a fully-managed machine learning workflow platform that provides services on data labeling, model building, training, tuning and deployment.
- SageMaker allows data scientists and developers to build scalable AI/ML models easily and efficiently.
- Models could be deployed in production at a much faster rate and with a fraction of the cost.
- Let's explore SageMaker:
<https://aws.amazon.com/sagemaker/#>

BUILD

- SageMaker offers data labeling service
- Prebuilt available notebooks with state of the art algorithms on AWS marketplace

TRAIN

- Train models using EC2 instances (on-demand and spot)
- Manage environments for training
- Hyperparameters optimization for model tuning

DEPLOY

- Easily deploy and scale models
- Autoscaling with 75% savings

AMAZON SAGEMAKER SERVICES



Amazon
SageMaker
Ground Truth

Amazon
SageMaker Neo

Amazon Textract

Amazon
Transcribe

Amazon
Translate

AWS Deep
Learning AMIs

AWS DeepLens

AWS DeepRacer

Amazon
Comprehend

Amazon Elastic
Inference

Amazon Forecast

Amazon Lex

Amazon
Personalize

Amazon Polly

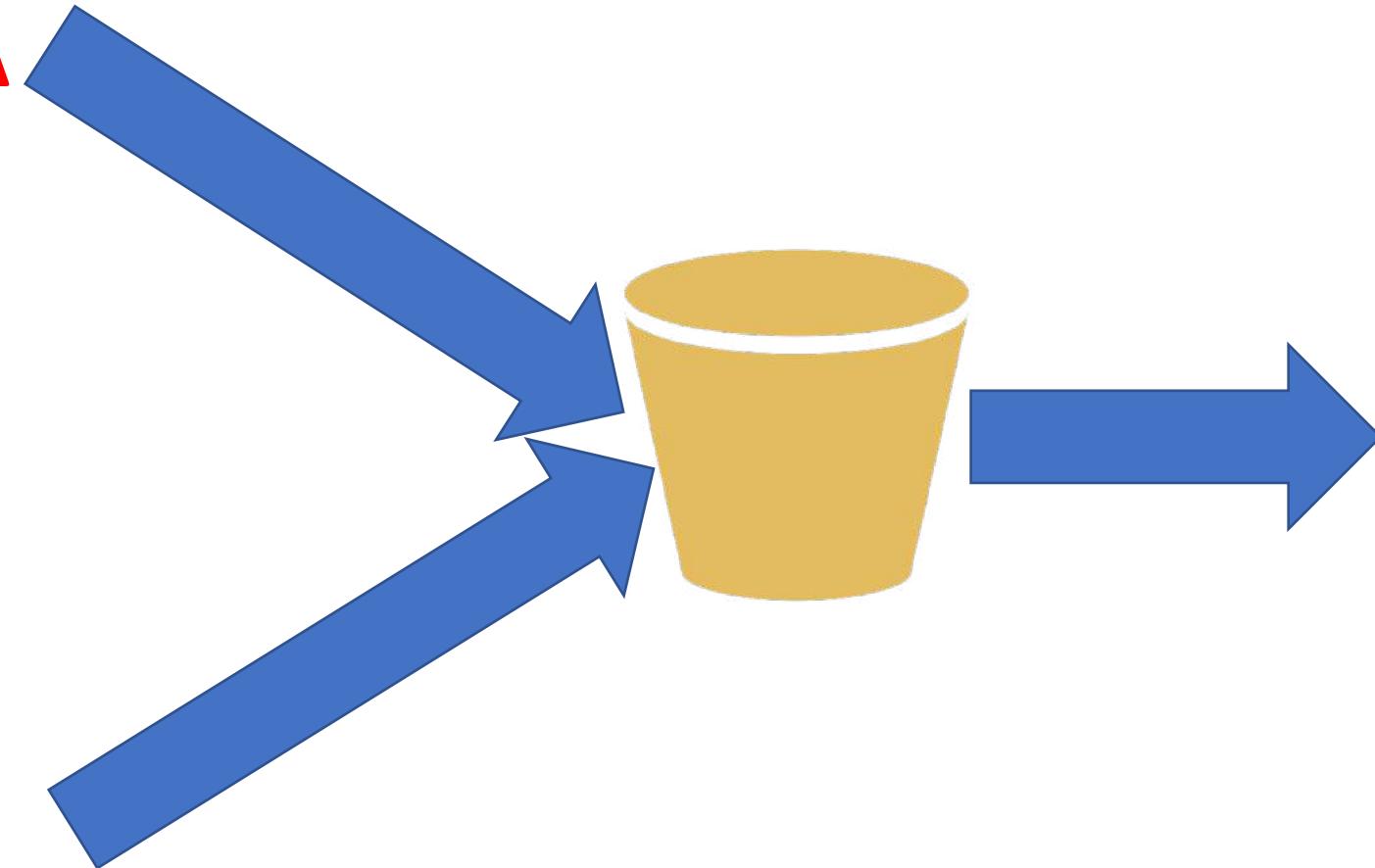
Amazon
Rekognition



TRAINING/TESTING DATA IN S3



TRAINING DATA

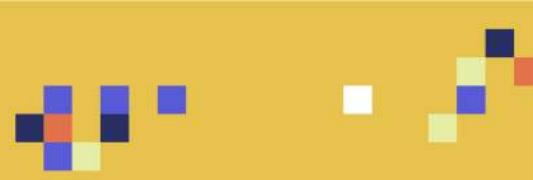


TESTING DATA

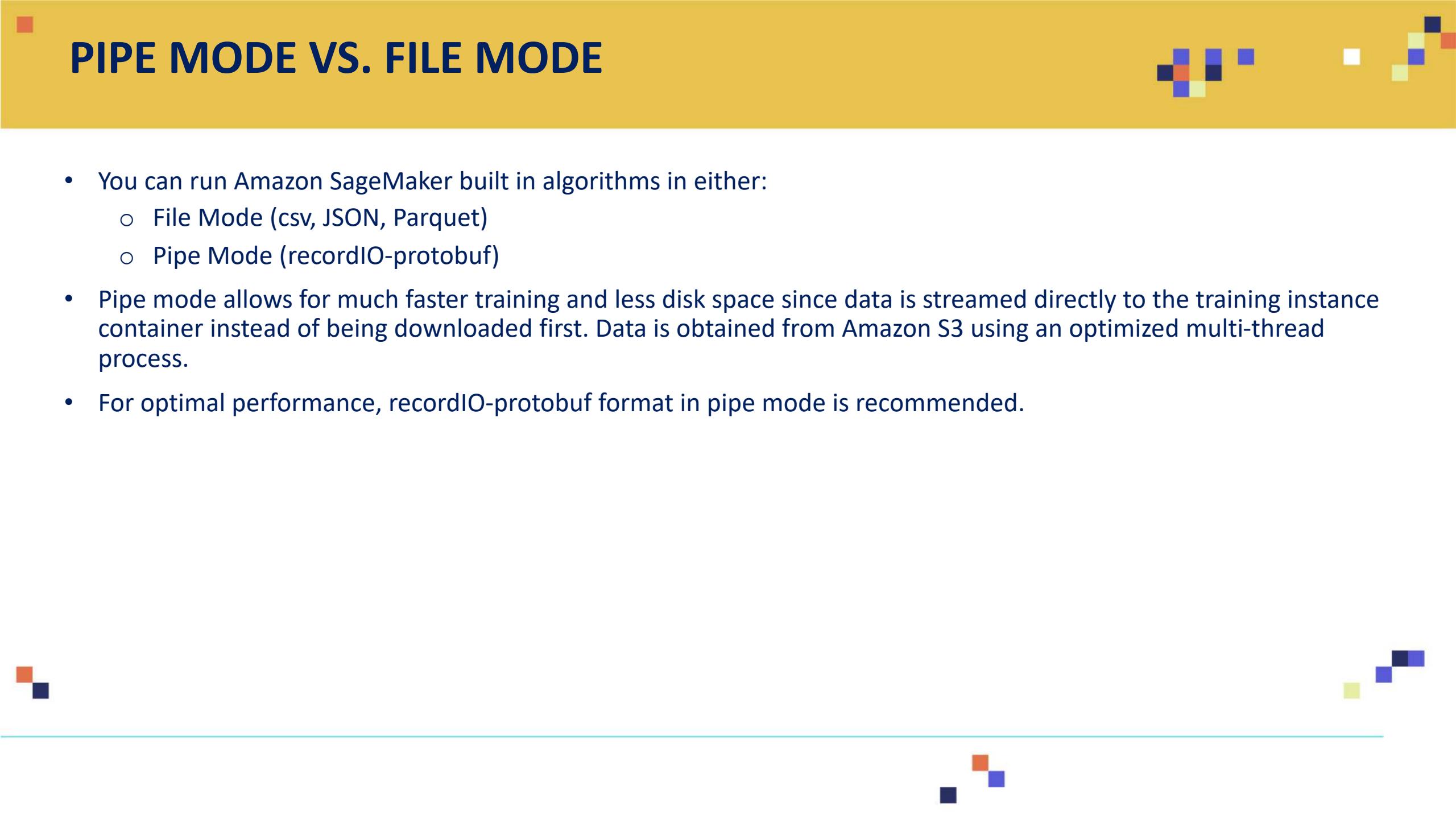
MODEL



PIPE MODE VS. FILE MODE



- You can run Amazon SageMaker built in algorithms in either:
 - File Mode (csv, JSON, Parquet)
 - Pipe Mode (recordIO-protobuf)
- Pipe mode allows for much faster training and less disk space since data is streamed directly to the training instance container instead of being downloaded first. Data is obtained from Amazon S3 using an optimized multi-thread process.
- For optimal performance, recordIO-protobuf format in pipe mode is recommended.

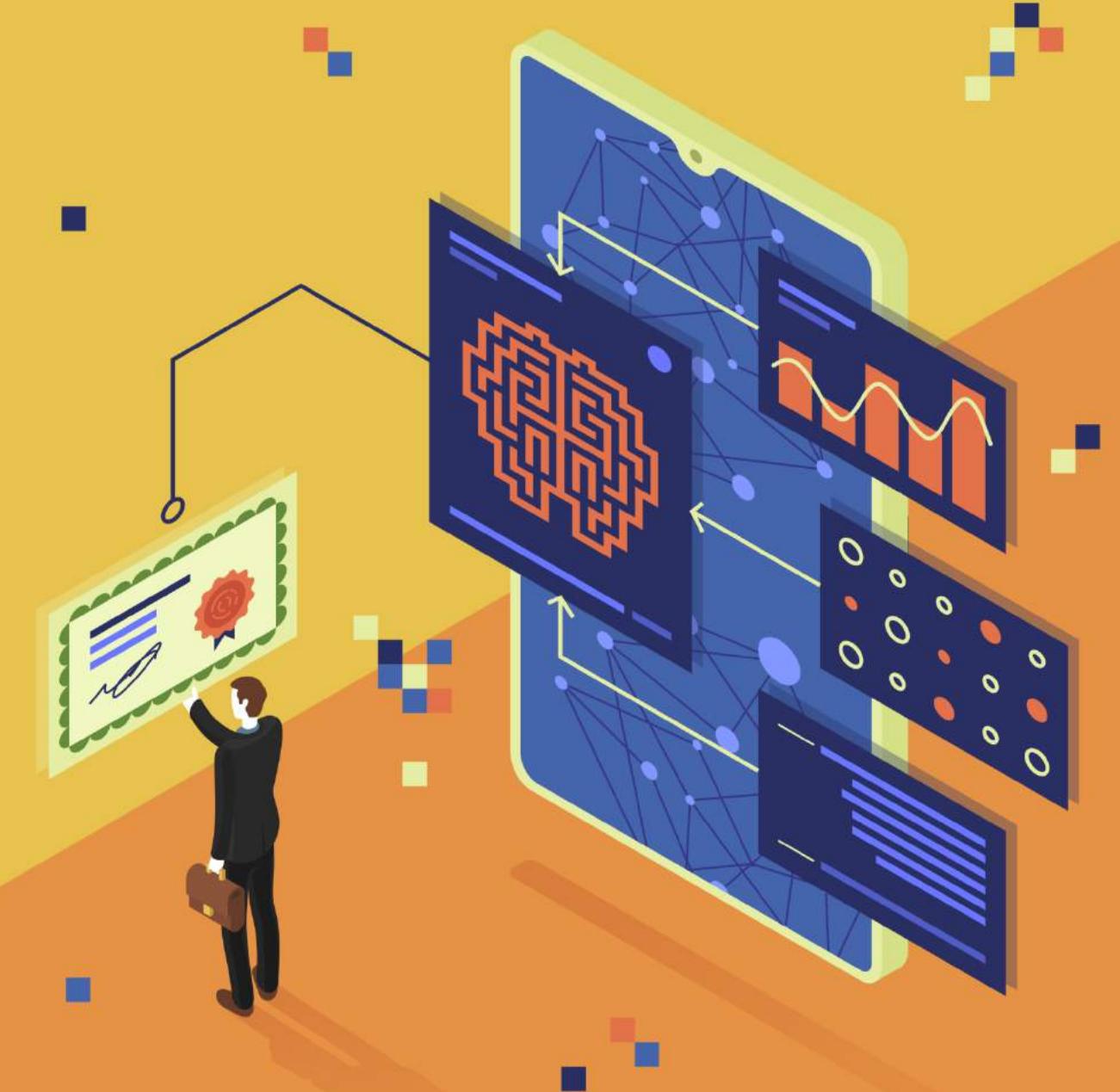


GPU Vs. CPU Vs. FPGA Vs. ASIC



	CPU	FPGA	GPU	ASIC
Acronym?	Center Processing Unit (CPU)	Field-Programmable Gate Array (FPGA)	Graphics Processing Unit (GPU)	Application Specific Integrated Circuit (ASIC)
What is it?	CPU is a basic sequential processor	FPGA is an integrated circuit designed to be configurable after manufacturing.	GPU is designed to process images and graphics	ASIC is a circuit designed for a specific target application.
Advantages?	<ul style="list-style-type: none">Easy to programGeneralist that could be applied in many applications.Cheap	<ul style="list-style-type: none">Could be configured after being installed in the field.Supports parallel processing	<ul style="list-style-type: none">Geared towards image analysis and graphics applications	<ul style="list-style-type: none">Optimized for a specific application with ultimate performance and power consumption
Disadvantages?	<ul style="list-style-type: none">Slow with no or limited parallel processing.	<ul style="list-style-type: none">Hard to program	<ul style="list-style-type: none">High power consumption	<ul style="list-style-type: none">RigidHigh costLong development time

AWS SAGEMAKER – PART #2



AMAZON SAGEMAKER COMPONENTS

- Two components are present in Amazon SageMaker:
 - Model training
 - Model deployment.
- To start training an AI/ML model using Amazon SageMaker, you will need to create a training job with the following:
 - **Amazon S3 bucket URL (training data)**: where the training data is located.
 - **Compute resources**: Amazon SageMaker will train the model using instances managed by Amazon SageMaker.
 - **Amazon S3 bucket URL (Output)**: this bucket will host the output from the training.
 - **Amazon Elastic Container Registry path**: where the training code is stored.
- Amazon SageMaker launches an ML compute instances once a training job is initiated.
- Amazon SageMaker uses: (1) training code and (2) training dataset to train the model.
- Amazon SageMaker saves the trained model artifacts in an S3 bucket.

Source: <https://docs.aws.amazon.com/sagemaker/latest/dg/how-it-works-training.html>

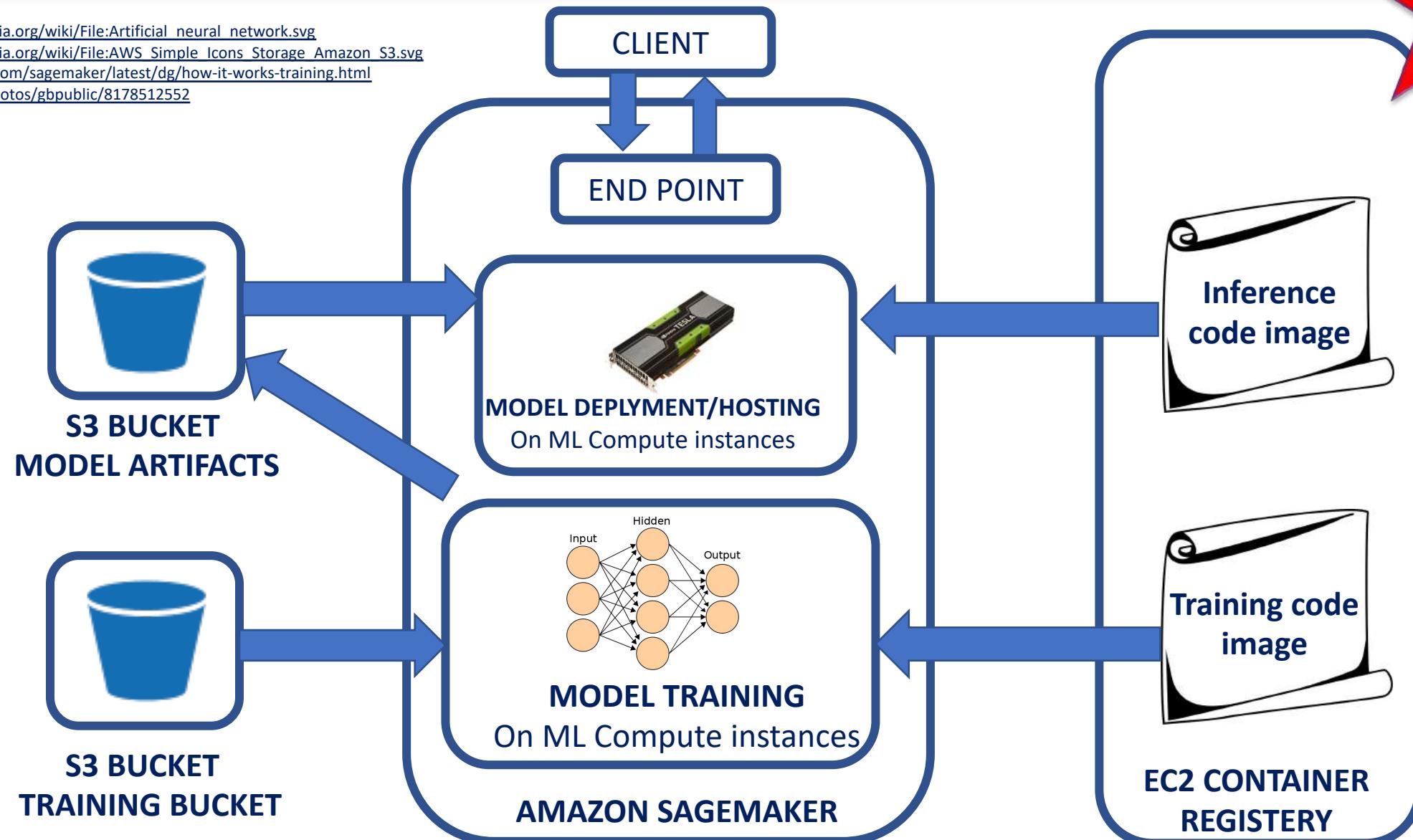
AMAZON SAGEMAKER MODEL TRAINING AND DEPLOYMENT OVERVIEW

Source:

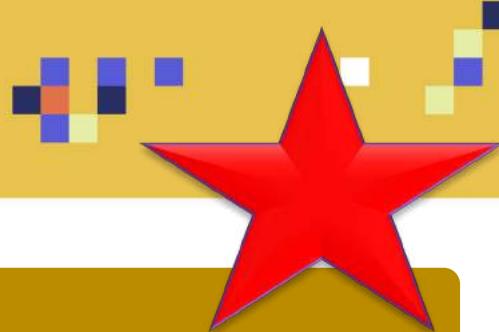
<https://docs.aws.amazon.com/sagemaker/latest/dg/how-it-works-training.html>



https://commons.wikimedia.org/wiki/File:Artificial_neural_network.svg
https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg
<https://docs.aws.amazon.com/sagemaker/latest/dg/how-it-works-training.html>
<https://www.flickr.com/photos/gbpublic/8178512552>



TRAINING OPTIONS OFFERED BY SAGEMAKER



USE AN ALGORITHM PROVIDED BY AMAZON SAGEMAKER

- Amazon SageMaker provides ready, off the shelf training algorithms such as: Linear Learner Algorithm and the XGBoost Algorithm, K Means, Principal Component Analysis, image classification, LDA, Sequence to Sequence Algorithm.

USE APACHE SPARK WITH AMAZON SAGEMAKER

- Apache Spark can be used to train models with Amazon SageMaker.

CUSTOM CODE TRAINING USING POPULAR DEEP LEARNING FRAMEWORKS

- custom python code with TensorFlow or Apache MXNet for model training.

USE YOUR OWN CUSTOM ALGORITHMS:

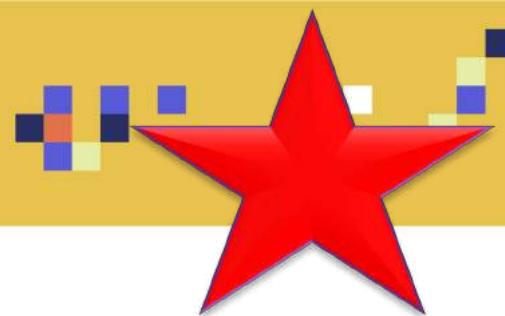
- The code could be placed in a docker container and then the registry path of the image could be provided in an Amazon SageMaker CreateTrainingJob API call.

AWS MARKETPLACE

- choose an algorithm from Amazon marketplace, <https://aws.amazon.com/marketplace/solutions/machine-learning>

Source: <https://docs.aws.amazon.com/sagemaker/latest/dg/how-it-works-training.html>

MODEL DEPLOYMENT BY SAGEMAKER



- After the AI/ML model is trained, it can be deployed as follows:
 - **For Single Prediction at a time:** set up a persistent endpoint using Amazon SageMaker hosting services.
 - **For Multiple Predictions:** Use Amazon SageMaker batch transform in order to obtain predictions for an entire dataset.
- **SageMaker Inference Pipeline:** SageMaker offers tools to process batch transforms in a pipeline format.
- **Batch Transform:** is used to allow preprocessing of large datasets and performing model inferencing quickly/efficiently without a need to have a persistent endpoint.
- **SageMaker Automatically Scaling:** as the workload changes throughout the day, SageMaker can dynamically adjust the number of instances provisioned so it could save money and compute resources.
- **Amazon SageMaker Elastic Inference (EI):** EI could be used to speed up inference and reduce latency by adding an accelerator without the need of having a dedicated GPU (which will cost much more)
- **Amazon SageMaker Neo:** Used to train TensorFlow, Apache MXNet, PyTorch, ONNX, and XGBoost models once and optimize them for deployment on ARM, Intel, and Nvidia processors. (*Before Neo, you will need to spend major man-hour efforts to deploy AI/ML Models on a specific hardware with specific compiler, memory, operating systems...etc.*)

DEEP LEARNING ON AWS



DEEP LEARNING ON AWS EC2



- AWS Deep Learning AMIs provide AI/ML developers with the infrastructure to quickly develop and scale deep learning in the cloud.
- Amazon EC2 instance could be easily launched on Amazon Linux or Ubuntu. The instance comes readily available with all the deep learning frameworks.
- Frameworks such as Apache MXNet, TensorFlow, the Microsoft Cognitive Toolkit (CNTK), Caffe, Caffe2, Theano, Torch and Keras.
- These tools can be used to build and train advanced AI/ML models.
- Instance types for deep learning include:
 - P3: 8 Tesla V100 GPU's
 - P2: 16 K80 GPU's
 - G3: 4 M60 GPU's (all Nvidia chips)

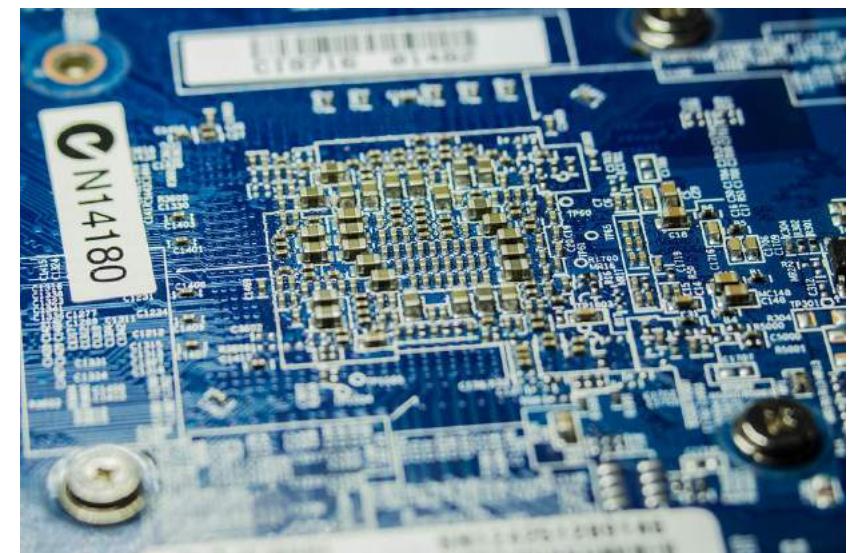


Photo Credit: <https://pixabay.com/photos/transistors-gpu-processor-pc-chip-1137502/>



DEEP LEARNING ON AWS EMR (ELASTIC MAP REDUCE)



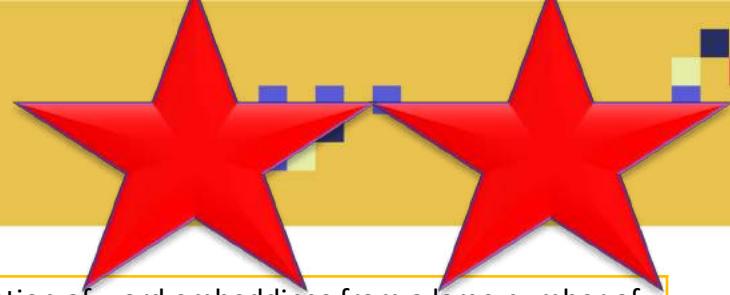
- EMR supports Apache MXNet and GPU instance types
- Recall that:
 - Amazon Elastic MapReduce (EMR) is an AWS tool for big data processing.
 - Amazon EMR allows developers to build an expandable low configuration service at a fraction of the cost of using an in-house cluster computing.
 - Amazon EMR is capable of processing big data across a Hadoop cluster of virtual servers on Amazon Elastic Compute Cloud (EC2) and Amazon Simple Storage Service (S3).
 - The term elastic means that EMR offers dynamic resizing ability meaning it could increase or decrease resources based on the demand.



AWS SAGEMAKER BUILT-IN ALGORITHMS



SAGEMAKER AVAILABLE ALGORITHMS



BlazingText Word2Vec	BlazingText implementation of the Word2Vec algorithm for scaling and accelerating the generation of word embeddings from a large number of documents.
DeepAR	An algorithm that generates accurate forecasts by learning patterns from many related time-series using recurrent neural networks (RNN).
Factorization Machines	A model with the ability to estimate all of the interactions between features even with a very small amount of data.
Gradient Boosted Trees (XGBoost)	Short for “Extreme Gradient Boosting”, XGBoost is an optimized distributed gradient boosting library.
Image Classification (ResNet)	A popular neural network for developing image classification systems.
IP Insights	An algorithm to detect malicious users or learn usage patterns of IP addresses.
K-Means Clustering	One of the simplest ML algorithms. It's used to find groups within unlabeled data.
K-Nearest Neighbor (k-NN)	An index based algorithm to address classification and regression based problems.
Latent Dirichlet Allocation (LDA)	A model that is well suited to automatically discovering the main topics present in a set of text files.
Linear Learner (Classification)	Linear classification uses an object's characteristics to identify the appropriate group that it belongs to.
Linear Learner (Regression)	Linear regression is used to predict the linear relationship between two variables.
Neural Topic Modelling (NTM)	A neural network based approach for learning topics from text and image datasets.
Object2Vec	A neural-embedding algorithm to compute nearest neighbors and to visualize natural clusters.
Object Detection	Detects, classifies, and places bounding boxes around multiple objects in an image.
Principal Component Analysis (PCA)	Often used in data pre-processing, this algorithm takes a table or matrix of many features and reduces it to a smaller number of representative features.
Random Cut Forest	An unsupervised machine learning algorithm for anomaly detection.
Semantic Segmentation	Partitions an image to identify places of interest by assigning a label to the individual pixels of the image.
Sequence2Sequence	A general-purpose encoder-decoder for text that is often used for machine translation, text summarization, etc.

Source: <https://aws.amazon.com/sagemaker/build/>

OBJECT DETECTION



OBJECT DETECTION: OVERVIEW

- SagMaker object detection uses deep learning to:
 - Detects objects
 - Classify them
- Object detection is a supervised training algorithm that is capable of detecting multiple objects in a given image and classifying them.
- The algorithm generates a bounding box along with a confidence interval associated with each object.
- The algorithm uses Single Shot multibox Detector (SSD) framework and supports two base networks:
 - VGG (Visual Geometry Group)
 - ResNet (Residual Network)
- You can start training the network from scratch or start from a pretrained network using transfer learning.

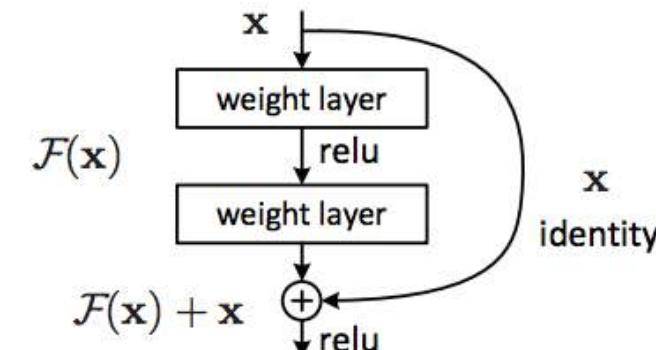
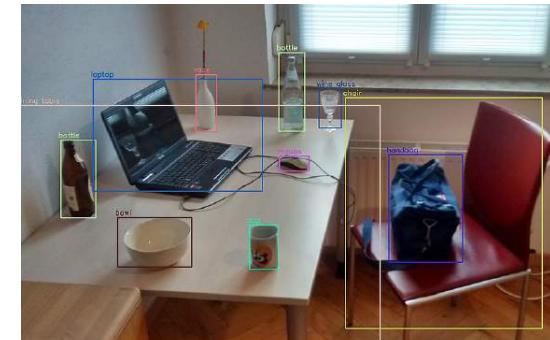


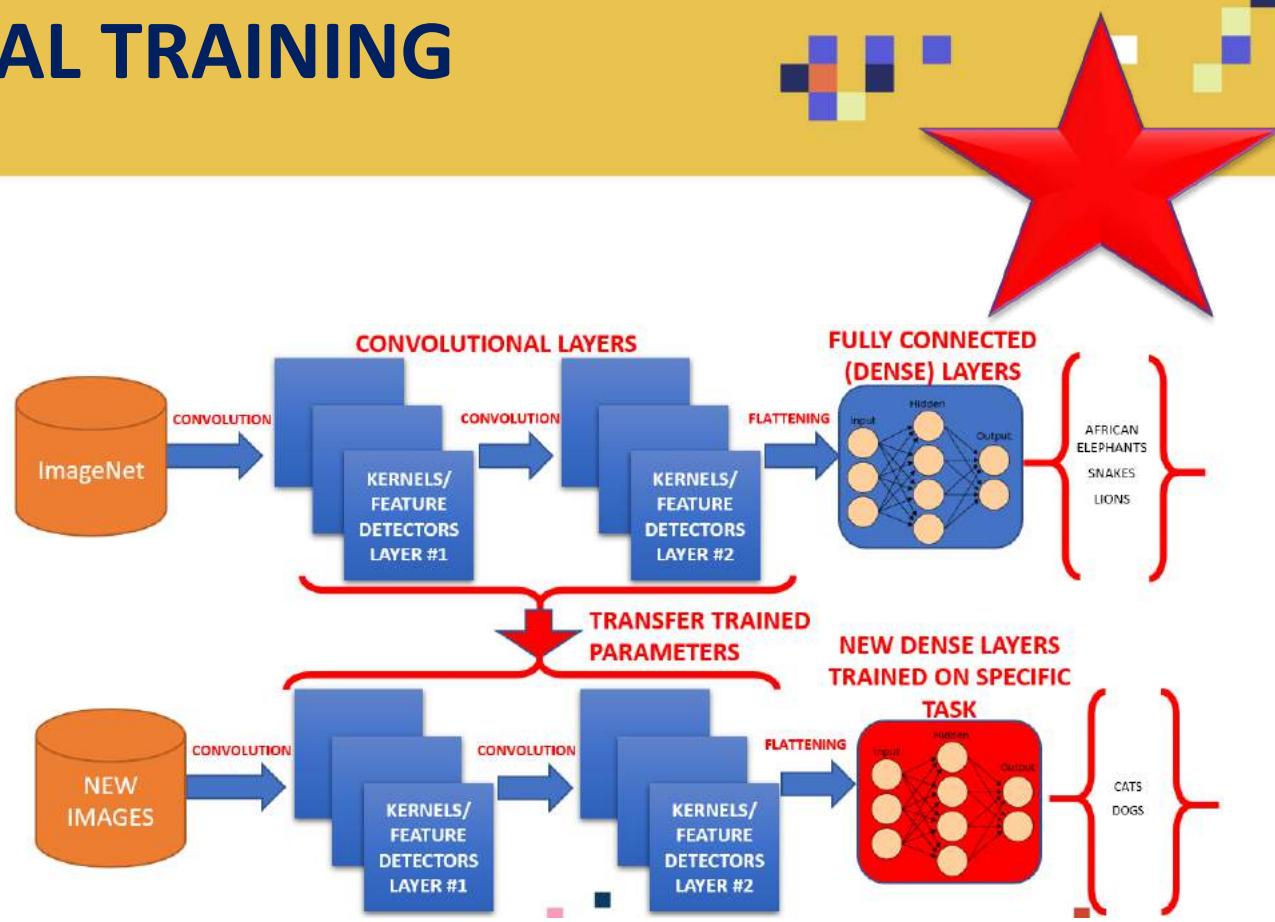
Photo Credit: <https://commons.wikimedia.org/wiki/File:Detected-with-YOLO--Schreibtisch-mit-Objekten.jpg>

Photo Credit: <https://www.flickr.com/photos/drbeachvacation/35477618781>

Photo Credit: <https://commons.wikimedia.org/wiki/File:Resnet.png>

OBJECT DETECTION: INCREMENTAL TRAINING

- Incremental training allows developers to start from a pretrained model instead of training a model from scratch.
- Incremental training is extremely efficient and saves time and resources.
- You can perform image augmentation to avoid model overfitting such as flip, rescale, and jitter



OBJECT DETECTION: INPUT/OUTPUT

- Object detection algorithm supports the following:
 - RecordIO (application/x-recordio) (for file mode)
 - Image (image/png, image/jpeg, and application/x-image) (for file mode).
- Note: for training in pipe mode, RecordIO (application/x-recordio) should be used.
- Each image needs a .json file for annotation, and the .json file should have the same name as the corresponding image.
- 4 properties as follows:
 - "file" specifies the path of the image file.
 - "image_size" property indicates image dimensions.
 - "annotations" property indicates the categories and bounding boxes for objects within the image.
 - "categories" includes mapping between class index and class name.

```
{ "file": "your_image_directory/sample_image1.jpg", "image_size":  
[  
{ "width": 500,  
"height": 400,  
"depth": 3  
}  
],  
"annotations": [  
{ "class_id": 0,  
"left": 111,  
"top": 134,  
"width": 61,  
"height": 128  
},  
{ "class_id": 0,  
"left": 161,  
"top": 250,  
"width": 79,  
"height": 143  
},  
{ "class_id": 1,  
"left": 101,  
"top": 185,  
"width": 42,  
"height": 130 } ],  
"categories": [  
{ "class_id": 0,  
"name": "dog" },  
{ "class_id": 1,  
"name": "cat" } ] }
```



OBJECT DETECTION: HYPERPARAMETERS

- **base_network**: The base network architecture to use.
- **use_pretrained_model**: Indicates whether to use a pre-trained model for training.
- **num_classes**: The number of output classes.
- **image_shape**: The image size for input images.
- **freeze_layer_pattern**: The regular expression (regex) for freezing layers in the base network.
- **Mini_batch_size**
- **Learning_rate**
- **Optimizer**: Sgd, adam, rmsprop, adadelta

For full list of hyperparameters: <https://docs.aws.amazon.com/sagemaker/latest/dg/object-detection-api-config.html>



OBJECT DETECTION: EC2 INSTANCE

- For training:
 - GPU instances for training: ml.p2.xlarge, ml.p2.8xlarge, ml.p2.16xlarge, ml.p3.2xlarge, ml.p3.8xlarge and ml.p3.16xlarge.
 - When training with large batch sizes, GPU instances with more memory is recommended.
 - You can also run the algorithm on multi-GPU and multi-machine settings for distributed training.
- For inference:
 - CPU (C5 and M5) and GPU (P2 and P3) instances can be used.

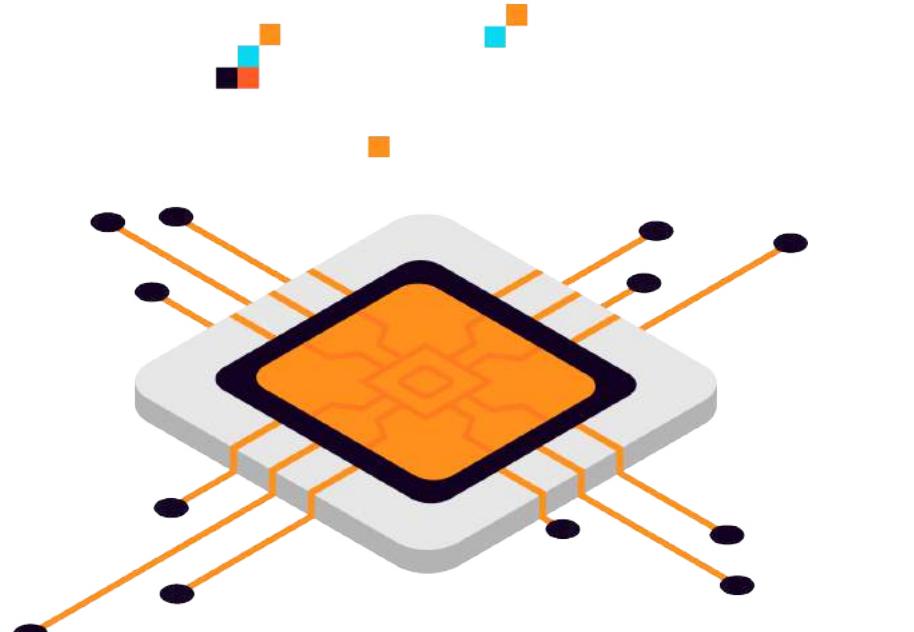


IMAGE CLASSIFICATION



IMAGE CLASSIFICATION: OVERVIEW

- Image classification algorithm is a supervised machine learning algorithm
- The algorithm is capable for classifying multiclass images.
- The algorithm DOES NOT specify the location of the object in the image so there is no bounding box.
- The algorithm takes in an image as an input
- It generates the label as an output such as: *Stop Sign, deer crossing, yield,..etc*
- SageMaker image classification algorithm uses convolutional neural network known as ResNet to perform classification.
- The network could be trained from scratch or starting from a pretrained network using transfer learning.

DEER CROSSING



STOP SIGN

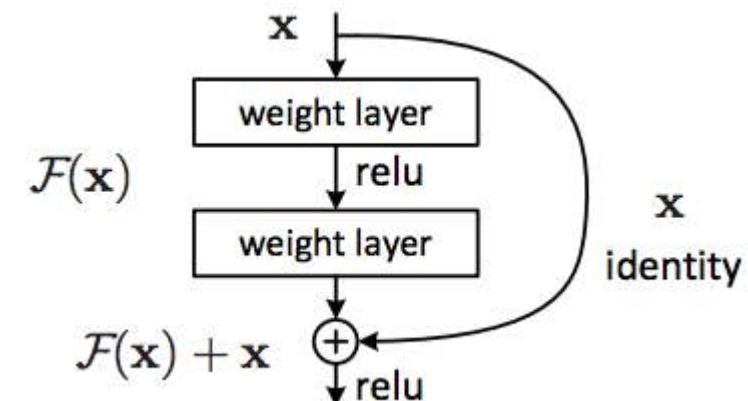
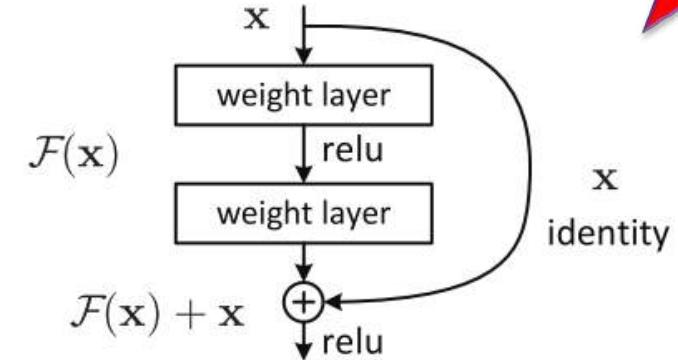
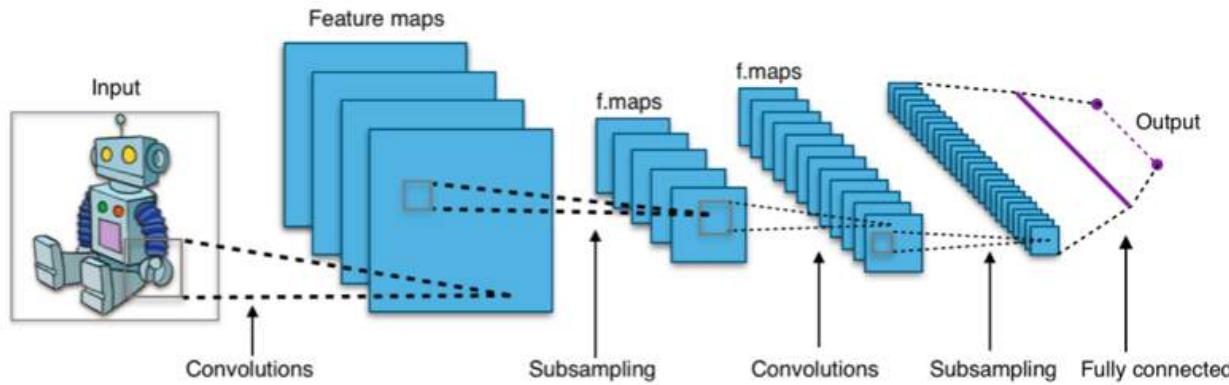
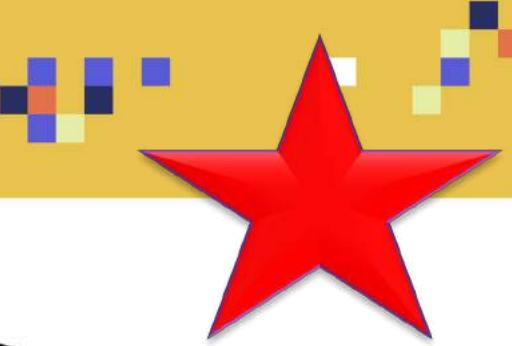


Photo Credit: <https://pixabay.com/illustrations/traffic-sign-road-sign-shield-6627/>

Photo Credit: <https://publicdomainvectors.org/en/free-clipart/Warning-for-deer-traffic-sign-vector/4025.html>

Photo Credit: <https://commons.wikimedia.org/wiki/File:Resnet.png>

IMAGE CLASSIFICATION: DEEPDIVE

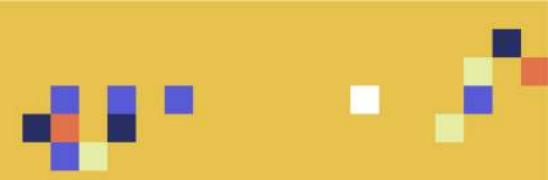


- Great article: <https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035>
- ImageNet is a large dataset with over 11 million images consisting of 11,000 categories.
- ImageNet is used to train ResNet deep network
- Once the networks is trained with ImageNet, the network can be repurposed for specific task using transfer learning.
- There are two modes for Image classification in Amazon SageMaker: (1) full training, (2) transfer learning.
 - In full training mode: network is being trained from scratch and with randomly initialised weights.
 - In transfer learning model: network is initialized with pre-trained weights and the classification head (dense layer) is initialized with random weights. The network is being trained with the new dataset which could be small.
- Expected Image size is 224x224x3

Photo Credit: https://en.wikipedia.org/wiki/File:Typical_cnn.png

Photo Credit: <https://commons.wikimedia.org/wiki/File:Resnet.png>

IMAGE CLASSIFICATION: INPUT/OUTPUT



- Amazon SageMaker recommends Apache MXNet RecordIO for the image input.
- The algorithm supports both RecordIO (application/x-recordio) and image (image/png, image/jpeg, and application/x-image) content types for training in file mode
- The algorithm supports RecordIO (application/x-recordio) content type for training in pipe mode.
- Augmented Manifest Image Format enables Pipe mode
- The algorithm supports image/png, image/jpeg, and application/x-image for inference.
- Image format requires .lst files to associate image index, class label, and path to the image.

IMAGE INDEX CLASS LABEL

5 1 your_image_directory/train_img_dog1.jpg 1000 0
your_image_directory/train_img_cat1.jpg 22 1
your_image_directory/train_img_dog2.jpg

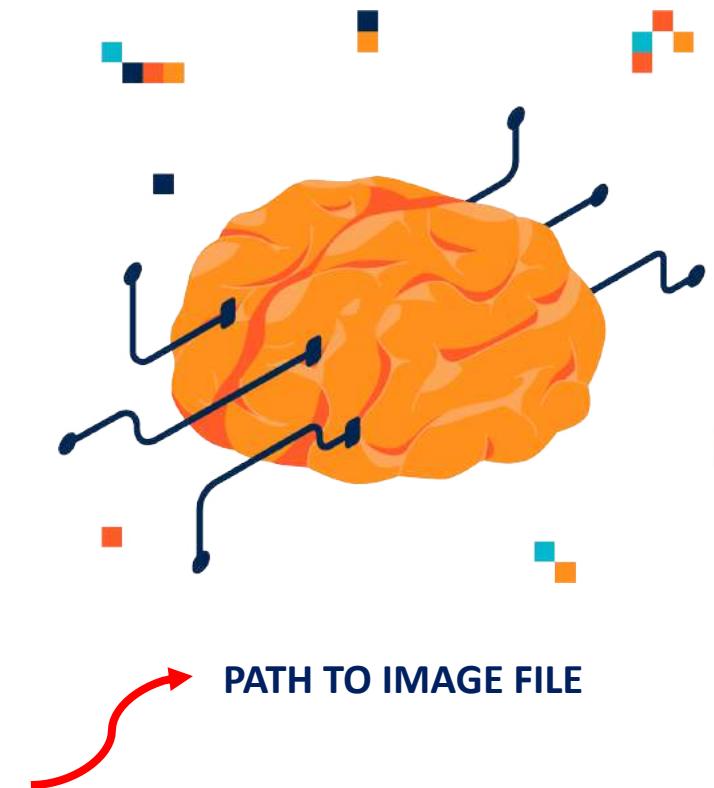


IMAGE CLASSIFICATION: HYPERPARAMETERS

- For the full list of hyperparameters:
<https://docs.aws.amazon.com/sagemaker/latest/dg/IC-Hyperparameter.html>
- `use_pretrained_model`: Indicates whether to use a pre-trained model for training.
- `num_classes`: The number of output classes.
- `augmentation_type`: Data augmentation type
- `image_shape`: The image size for input images.
- `Mini_batch_size`
- Weight decay
- beta 1
- beta 2
- eps
- gamma
- Learning_rate
- Optimizer: Sgd, adam, rmsprop, adadelta



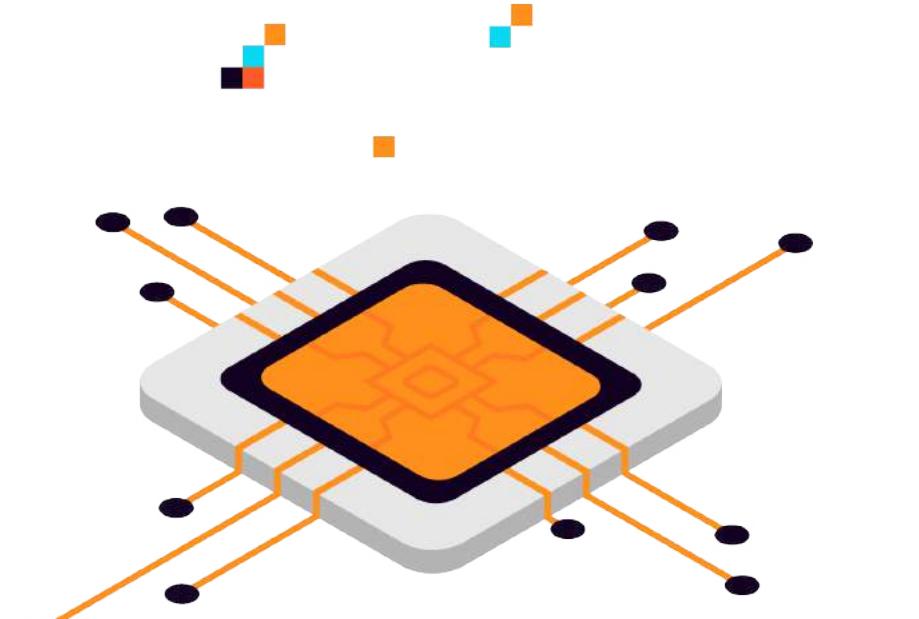
IMAGE CLASSIFICATION: EC2 INSTANCE

For training:

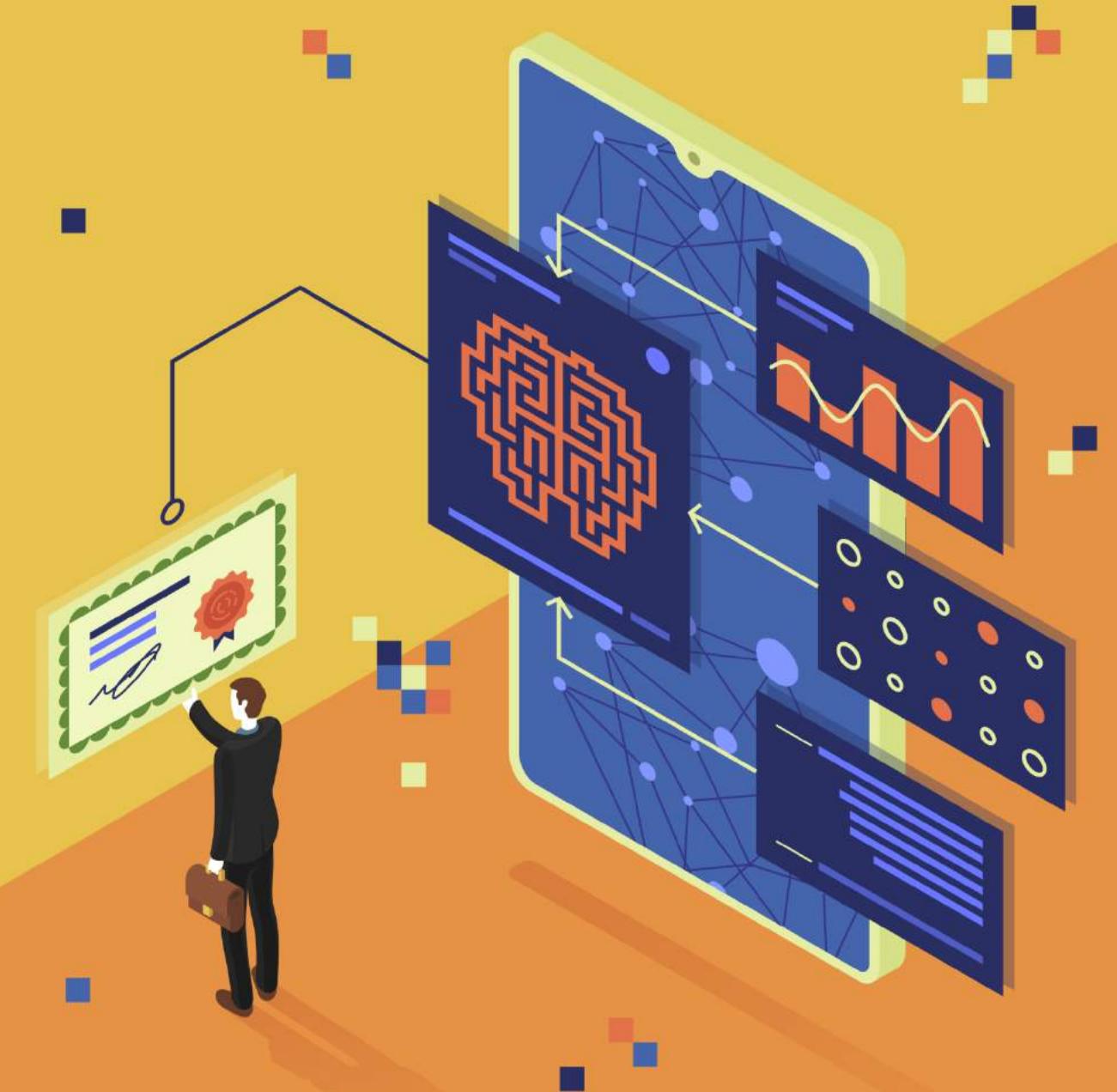
- GPU instances for training: ml.p2.xlarge, ml.p2.8xlarge, ml.p2.16xlarge, ml.p3.2xlarge, ml.p3.8xlarge and ml.p3.16xlarge.
- When training with large batch sizes, GPU instances with more memory is recommended.
- You can also run the algorithm on multi-GPU and multi-machine settings for distributed training.

For inference:

- CPU (C5 and M5) and GPU (P2 and P3) instances can be used.



SEMANTIC SEGMENTATION



SEMANTIC SEGMENTATION: OVERVIEW

- Semantic segmentation algorithm provides image classification on the pixel-level.
- Given a set of classes, SageMaker semantic segmentation algorithm tags every pixel with a class label.
- Semantic segmentation is critical for computer vision applications such as self-driving cars.
- Recall that:
 - **(1) Amazon SageMaker Image Classification:** was a supervised learning algorithm that takes in an entire image and classify it one or more classes.
 - **(2) Object Detection Algorithm:** was a supervised algorithm that **detects and classifies** all objects in a given image by providing a bounding box around the object.

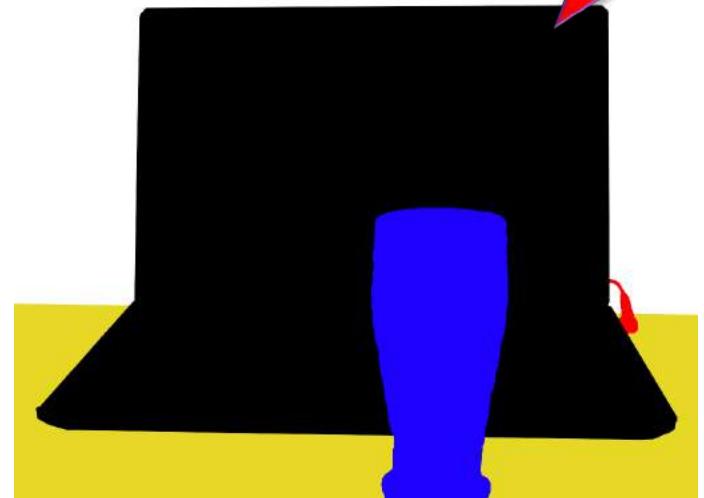


Photo Credit: <https://commons.wikimedia.org/wiki/File:Image-segmentation-example-segmented.png>

SEMANTIC SEGMENTATION: OVERVIEW



- One of the key advantages of semantic segmentation is that it provides details about the shape of the object because it looks at every pixel.
- The segmentation algorithm provides a segmentation mask which is a RGB (or grayscale) image with the same shape as the input image.
- Built using MXNet Gluon framework and Gluon CV toolkit.
- Three deep neural network-based algorithms are available:
 - Fully-Convolutional Network (FCN) algorithm
 - Pyramid Scene Parsing (PSP) algorithm
 - DeepLabV3
- The algorithm consists of two elements:
 - *Encoder (backbone)*: network that generates activation maps of features.
 - *Decoder*: network that takes in the encoded activation maps and constructs segmentation mask from it.

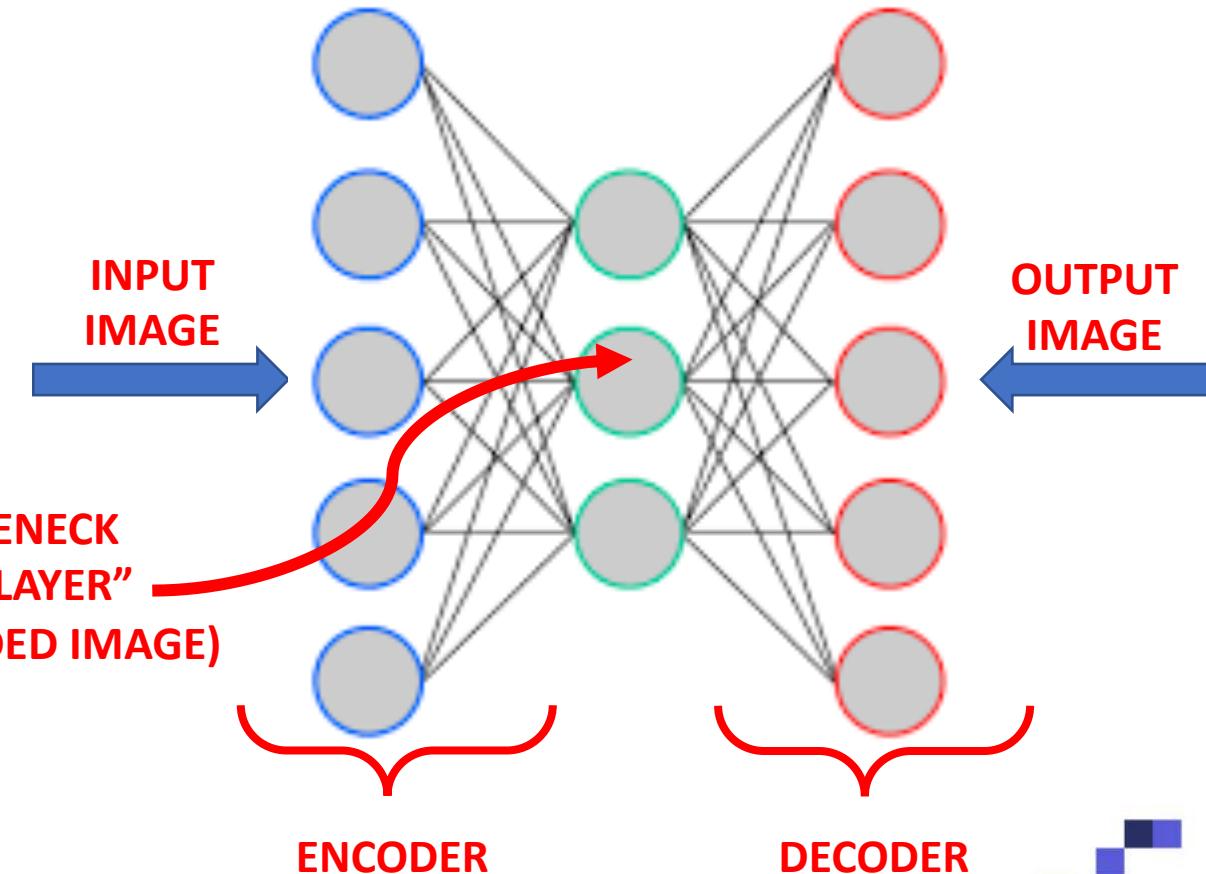
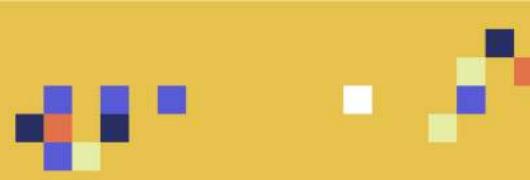


Photo Credit: https://commons.wikimedia.org/wiki/File:Autoencoder_structure.png

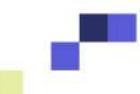
Photo Credit: https://commons.wikimedia.org/wiki/File:Artificial_neural_network_image_recognition.png



SEMANTIC SEGMENTATION: OVERVIEW



- Several backbones are available for the FCN, PSP, and DeepLabV3 algorithms:
 - ResNet50
 - ResNet101
- Backbone networks are trained using ImageNet
- Backbones can be trained from scratch or using pretrained network (transfer learning)
- Decoders MUST be trained from scratch.
- For inference:
 - Amazon SageMaker hosting service is used to deploy trained model.
 - Segmentation mask can be a PNG image or set of probabilities for each class for each pixel.



SEMANTIC SEGMENTATION: CITYSCAPES DATASET



 CITYSCAPES
DATASET

News Overview ▾ Examples ▾ Benchmarks ▾ Download

The Cityscapes Dataset
Semantic, instance-wise, dense pixel annotations of 30 classes

Dataset Overview



A circular map of Germany with red dots indicating dataset locations across the country.

A circular collage of four smaller images showing examples of semantic segmentation results for different scenes.

A diagram illustrating a neural network's feature maps or activation patterns.

<https://www.cityscapes-dataset.com/>

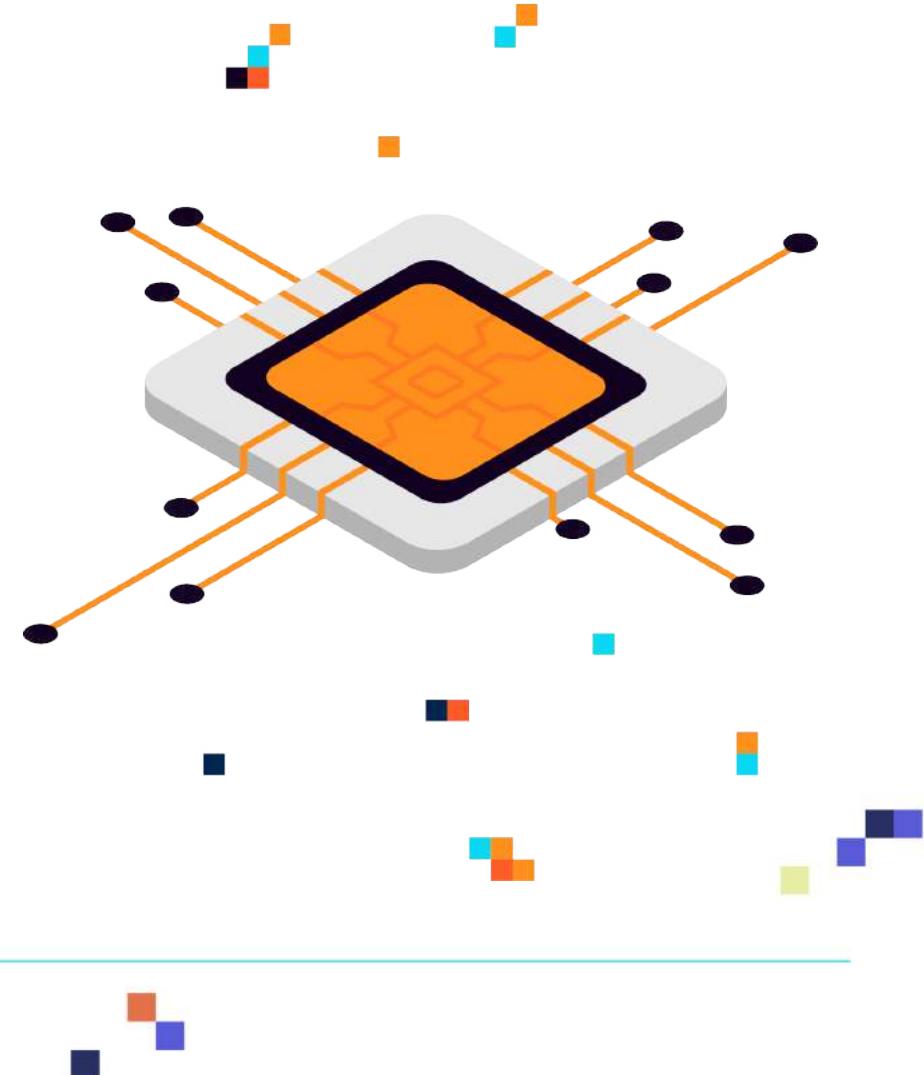
SEMANTIC SEGMENTATION: HYPERPARAMETERS

- For the entire list of hyperparameters, check this out:
<https://docs.aws.amazon.com/sagemaker/latest/dg/segmentation-hyperparameters.html>
- Backbone: The backbone to use for the algorithm's encoder component, examples: resnet-50, resnet-101, resnet-50
- use_pretrained_model: Whether a pretrained model is to be used for the backbone.
- Algorithm: The algorithm to use for semantic segmentation, examples: fcn: Fully-Convolutional Network (FCN) algorithm, psp: Pyramid Scene Parsing (PSP) algorithm, deeplab: DeepLab V3 algorithm
- num_classes: number of classes to segment.
- learning rate
- batch size
- Epochs
- optimizer



SEMANTIC SEGMENTATION: EC2 INSTANCE

- For training:
 - GPU P2 or P3
 - GPU instances for training: ml.p2.xlarge, ml.p2.8xlarge, ml.p2.16xlarge, ml.p3.2xlarge, ml.p3.8xlarge and ml.p3.16xlarge.
- For inference:
 - CPU (C5 and M5) and GPU (P2 and P3) instances can be used.

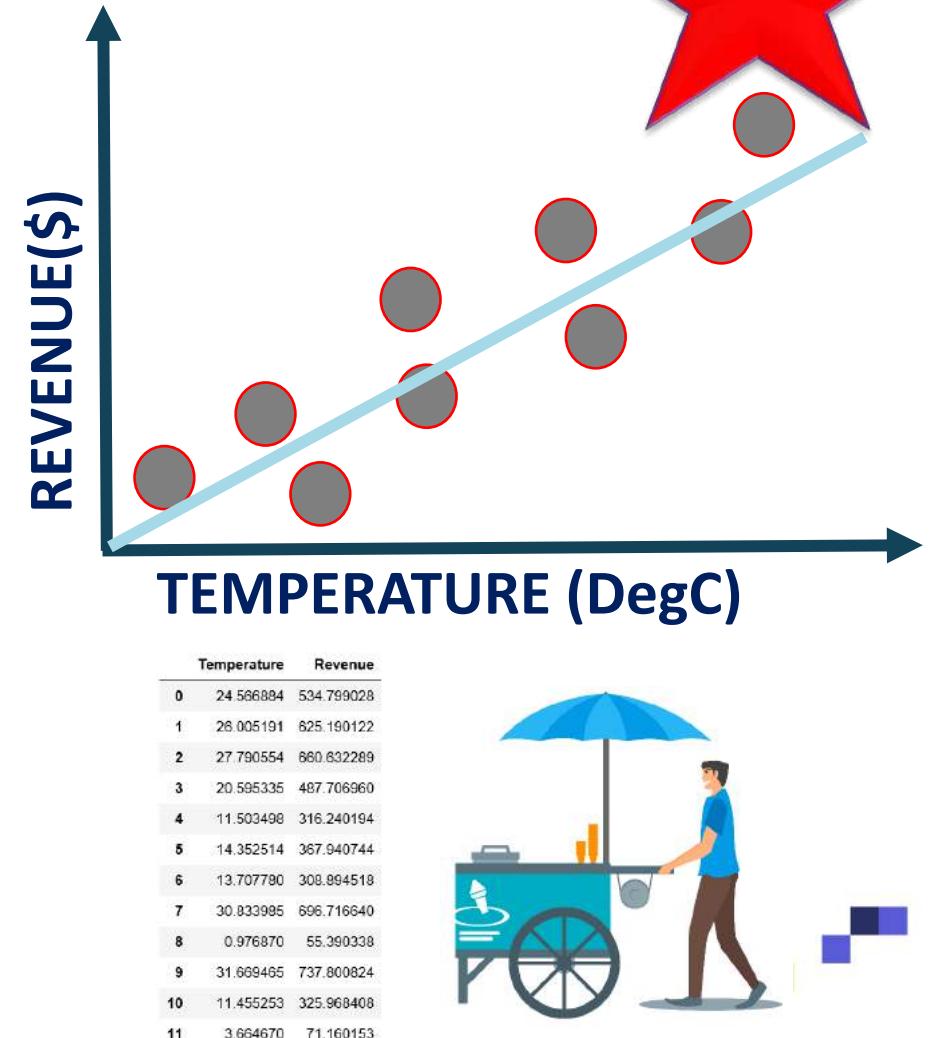


SAGEMAKER LINEAR LEARNER

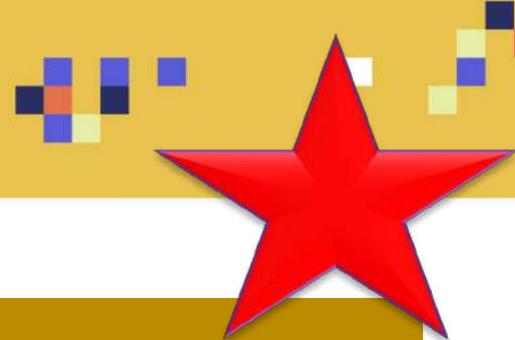


SAGEMAKER LINEAR LEARNER: OVERVIEW

- Linear Learner is a supervised learning algorithm that is used to fit a line to the training data.
- It could be used for both classification and regression tasks as follows:
 - **Regression:** output contains continuous numeric values
 - **Binary classification:** output label must be either 0 or 1 (linear threshold function is used).
 - **Multiclass classification:** output labels must be from 0 to *num_classes - 1*.
- The best model optimizes either of the following:
 - **For regression:** focus on Continuous metrics such as mean square error, root mean squared error, cross entropy loss, absolute error.
 - **For classification:** focus on discrete metrics such as F1 score, precision, recall, or accuracy.



SAGEMAKER LINEAR LEARNER: USE CASES



DISCRETE BINARY CLASSIFICATION

- Does this patient have a disease or not?

DISCRETE MULTICLASS CLASSIFICATION

- Should an autonomous car stop, slow down or accelerate?

REGRESSION TASKS

- Revenue predictions based on previous years performance.



SAGEMAKER LINEAR LEARNER: OVERVIEW



Preprocessing

- Ensure that data is shuffled before training
- Normalization or feature scaling is offered by Linear Learner (which is great!)
- Normalization or feature scaling is a critical preprocessing step to ensure that the model does not become dominated by the weight of a single feature.

Training

- Linear Learner uses stochastic gradient descent to perform the training
- Select an appropriate optimization algorithm such as Adam, AdaGrad, and SGD
- Hyperparameters, such as momentum, learning rate, and the learning rate schedule can be selected.
- Overcome model overfitting using L1, L2 regularization
- Multiple models could be optimized in parallel

Validation

- Trained models are evaluated against a validation dataset and best model is selected based on the following metrics:
 - **For regression:** mean square error, root mean squared error, cross entropy loss, absolute error.
 - **For classification:** F1 score, precision, recall, or accuracy.

SAGEMAKER LINEAR LEARNER HYPERPARAMETERS

- **Learning Rate:** The step size used by the optimizer for parameter updates.
- **L1:** L1 regularization parameter.
- **Momentum:** momentum of the SGD optimizer.
- **Mini_batch_size:** The number of observations per mini-batch
- **Wd:** The weight decay parameter, also known as the L2 regularization parameter.
- Check out the rest of hyperparameters here:
https://docs.aws.amazon.com/sagemaker/latest/dg/ll_hyperparameters.html



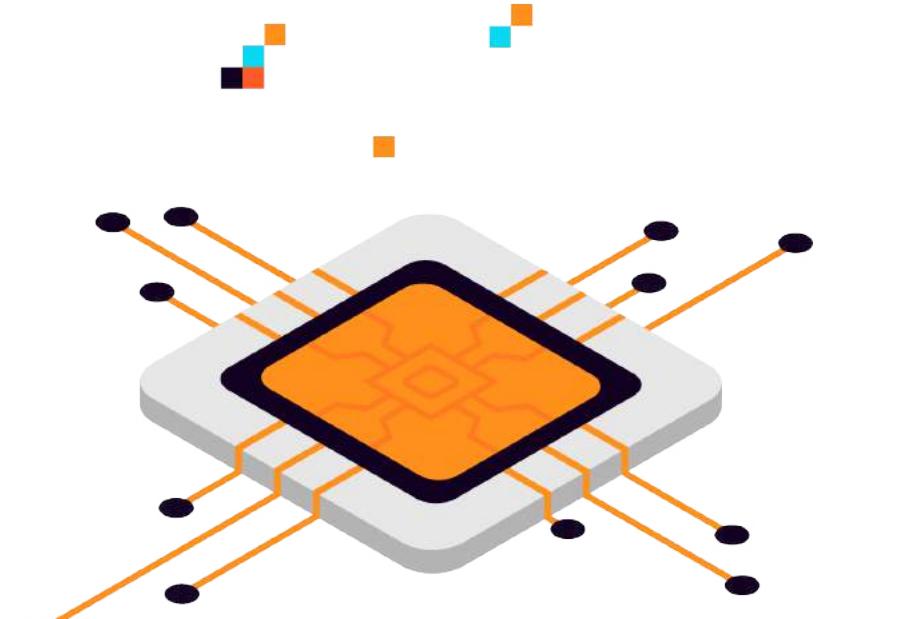
SAGEMAKER LINEAR LEARNER: INPUT/OUTPUT DATA

- Amazon SageMaker linear learner supports the following input data types:
 - RecordIO-wrapped protobuf (*note: only Float32 tensors are supported*)
 - Text/CSV (*note: First column assumed to be the target label*)
 - File or Pipe mode both supported
- For inference, linear learner algorithm supports the application/json, application/x-recordio-protobuf, and text/csv formats.
- For regression (predictor_type='regressor'), the score is the prediction produced by the model.
- For classification (predictor_type='binary_classifier' or predictor_type='multiclass_classifier'), the model returns a score and also a predicted_label. The predicted_label is the class predicted by the model and the score measures the strength of that prediction.



SAGEMAKER LINEAR LEARNER: EC2 INSTANCE

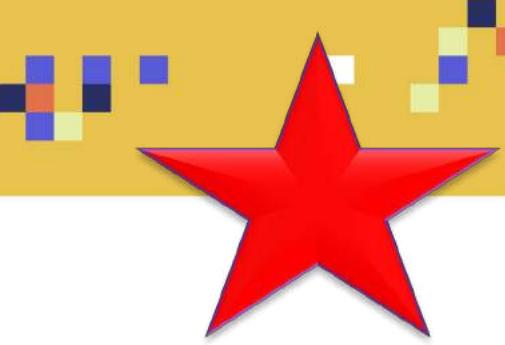
- Linear Learner algorithm could be trained on:
 - Single CPU and GPU instances
 - Multi-Machine CPU and GPU instances
- During testing, multi-GPU computers are not necessary (add cost with no value).



FACTORIZATION MACHINES



FACTORIZATION MACHINES: OVERVIEW



- A factorization machine is a supervised learning algorithm.
- It could be used to perform general purpose classification and regression operations.
- It is an extension of a linear model and works well with highly sparse data.
- An example of high sparse data is (1) click prediction and (2) item recommendation.
- For example, factorization machines can be used to predict the behaviour of customers with an ad is placed by tracking the number/rate of clicks patterns. This is a prime example of sparse dataset.

title	'Til There Was You (1997)	1-900 (1994)	101 Dalmatians (1996)	12 Angry Men (1957)	187 Days in the Valley (1997)	2 Leagues Under the Sea (1954)	20,000 A Space Odyssey (1968)	3 Ninjas: High Noon At Mega Mountain (1998)	39 Steps, The (1935)	...	Yankee Zulu (1994)	Year of the Horse (1997)	You So Crazy (1994)	Frankenstein (1974)	Young Guns (1988)	Young Guns II (1990)	I
user_id	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	NaN	NaN	2.0	5.0	NaN	NaN	3.0	4.0	NaN	NaN	...	NaN	NaN	NaN	5.0	3.0	NaN
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN	2.0	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN
5	NaN	NaN	2.0	NaN	NaN	NaN	NaN	4.0	NaN	NaN	...	NaN	NaN	NaN	4.0	NaN	NaN
6	NaN	NaN	NaN	4.0	NaN	NaN	NaN	5.0	NaN	NaN	...	NaN	NaN	NaN	4.0	NaN	NaN
7	NaN	NaN	NaN	4.0	NaN	NaN	5.0	5.0	NaN	4.0	...	NaN	NaN	NaN	5.0	3.0	NaN
8	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	4.0	...	NaN	NaN	NaN	NaN	NaN	NaN
10	NaN	NaN	NaN	5.0	NaN	NaN	NaN	5.0	NaN	4.0	...	NaN	NaN	NaN	NaN	NaN	NaN
11	NaN	NaN	NaN	NaN	NaN	NaN	NaN	4.0	NaN	NaN	...	NaN	NaN	NaN	4.0	NaN	NaN
12	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN
13	NaN	NaN	2.0	4.0	NaN	NaN	2.0	5.0	1.0	4.0	...	NaN	2.0	NaN	5.0	3.0	NaN
14	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN

**SPARSE DATASET
WHERE USERS DON'T
GENERALLY RATE EVERY
MOVIE OR PRODUCT!**

FACTORIZATION MACHINES: HYPERPARAMETERS

- Full set of hyperparameters:
<https://docs.aws.amazon.com/sagemaker/latest/dg/fact-machines-hyperparameters.html>
- feature_dim: number of features in the input data.
- num_factors: dimensionality of factorization.
- predictor_type: type of predictor, either binary_classifier or regressor.
- bias_init_method: initialization method for bias term.
 - Normal: weights are initialized according to normal distribution.
 - uniform: weights initialized with uniform distribution.
 - constant: weights initialized to scalar values.



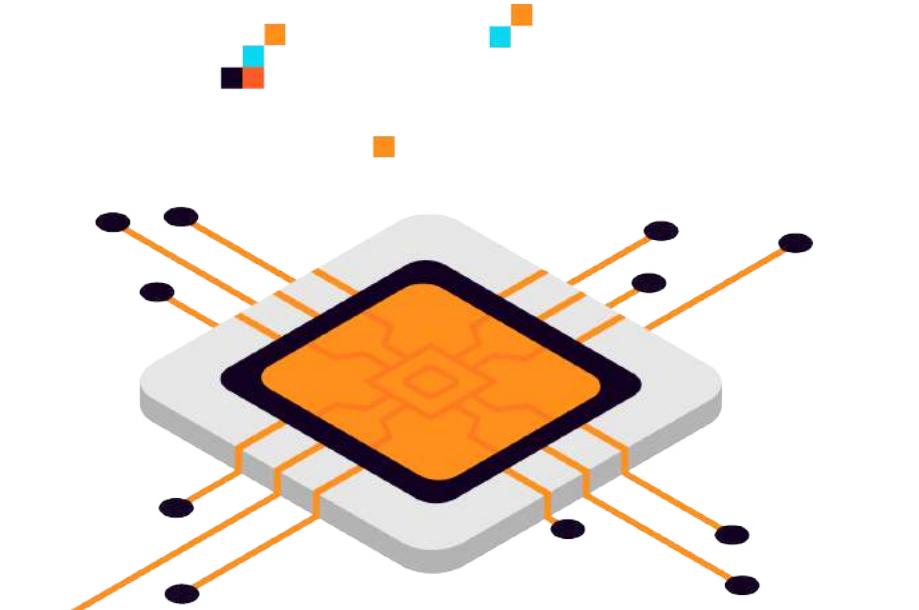
FACTORIZATION MACHINES: INPUT/OUTPUT

- During training:
 - factorization machines expects recordIO-protobuf data format with Float32 tensors.
 - CSV does not work well because data is sparse.
 - Both File and Pipe mode training are supported for recordIO-wrapped protobuf.
- During inference:
 - Factorization machines support the application/json and x-recordio-protobuf formats.

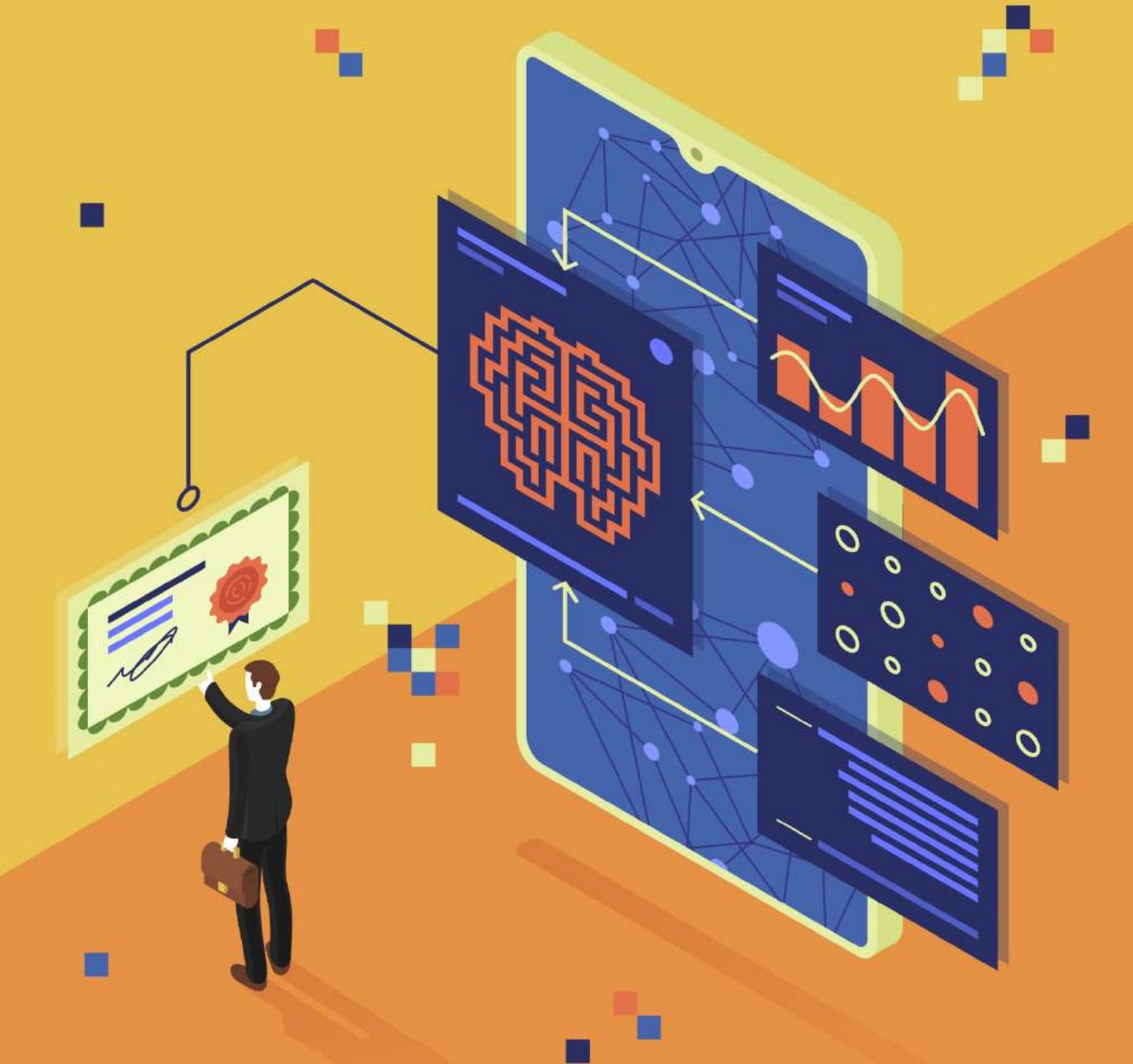


FACTORIZATION MACHINES: INSTANCE TYPES

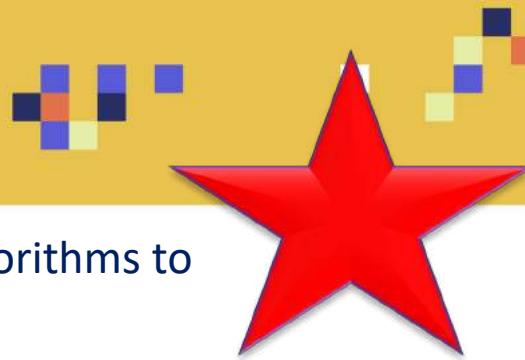
- SageMaker Factorization Machines is highly scalable.
- It could work on multiple distributed instances.
- For Factorization Machines Training and inference:
 - CPU instance is recommended since GPU is preferred with dense data only



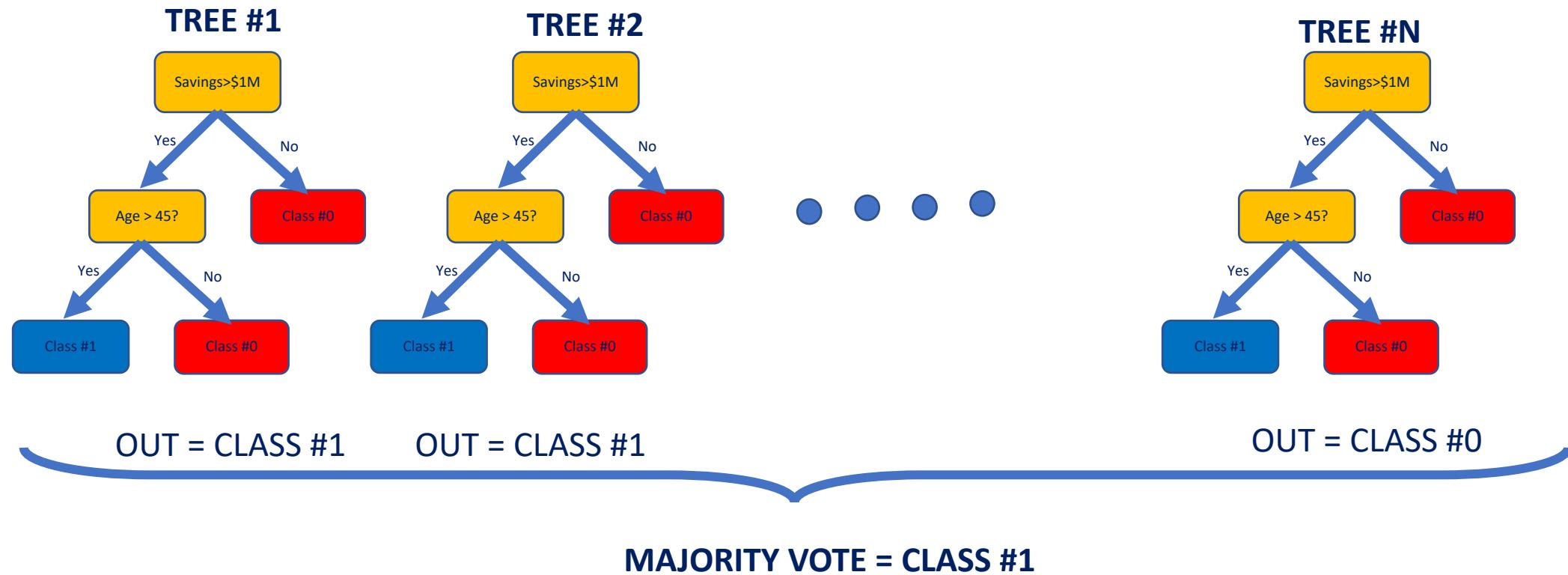
SAGEMAKER XGBOOST



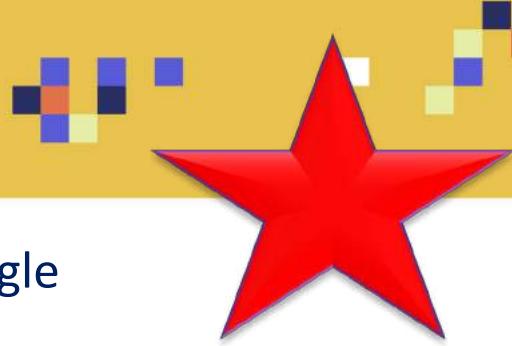
SAGEMAKER XGBOOST: OVERVIEW



- XGBoost or Extreme Gradient Boosting algorithm is one of the most famous and powerful algorithms to perform both regression and classification tasks.
- XGBoost is a supervised learning algorithm and implements gradient boosted trees algorithm.
- The algorithm work by combining an ensemble of predictions from several weak models.



SAGEMAKER XGBOOST: OVERVIEW



- Recently, XGBoost is the go to algorithm for most developers and has won several Kaggle competitions.
- Why does Xgboost work really well?
 - Since the technique is an ensemble algorithm, it is very robust and could work well with several data types and complex distributions.
 - Xgboost has a many tunable hyperparameters that could improve model fitting.
- What are the applications of XGBoost?
 - XGBoost could be used for fraud detection to detect the probability of a fraudulent transactions based on transaction features.

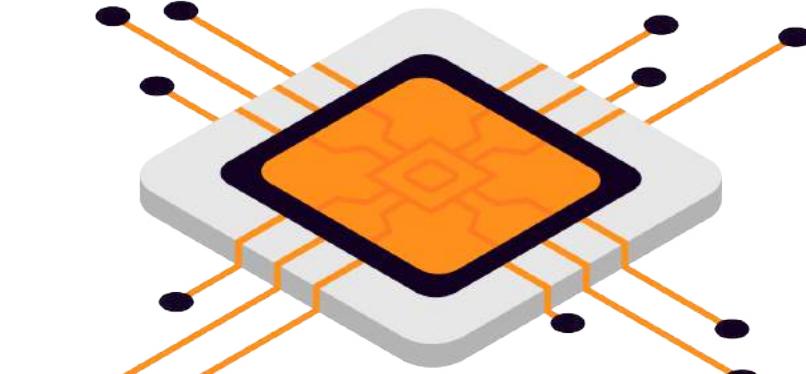
SAGEMAKER XGBOOST: INPUT/OUTPUT DATA

- Gradient boosting uses tabular data for inputs/outputs:
 - Rows represent observations,
 - One column represents the output or target label
 - The rest of the columns represent the inputs (features)
- Amazon SageMaker implementation of XGBoost supports the following file format for training and inference :
 - CSV
 - libsvm
- Xgboost does not support protobuf format (*note: this is unique compared to other Amazon SageMaker algorithms, which use the protobuf training input format*).



SAGEMAKER XGBOOST: EC2 INSTANCE

- XGBoost currently only trains using CPUs.
- XGboost is **memory intensive algorithm** so it does not require much compute.
- M4: General-purpose compute instance is recommended.



SAGEMAKER XGBOOST: HYPERPARAMETERS

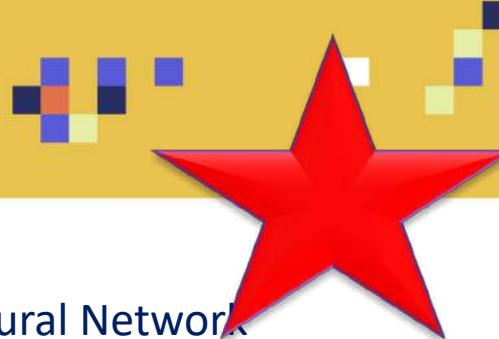
- **Alpha:** L1 regularization term on weights.
- **Lambda:** L2 regularization
- **Booster:** Which booster to use.
- **Eta:** Step size shrinkage used in updates to prevent overfitting.
- **Gamma:** Minimum loss reduction needed to add more partitions to the tree.
- Check out the rest of hyperparameters here:
https://docs.aws.amazon.com/sagemaker/latest/dg/xgboost_hyperparameters.html



SAGEMAKER SEQ2SEQ



SAGEMAKER SEQUENCE-TO-SEQUENCE: OVERVIEW



- Sequence to Sequence is a supervised machine learning algorithm available in SageMaker.
- Under the hood, the algorithm uses Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNN).
- The algorithm takes in sequence of tokens such as audio and generate a sequence of tokens as well.
- Examples:
 - **Machine Translation:**
 - Input: English sentence
 - output: translated French sentence
 - **Text summarization:**
 - Input: Long sentence
 - Output: summarized sentence
 - **Speech to text:**
 - Input: audio clips
 - Output: tokens of words

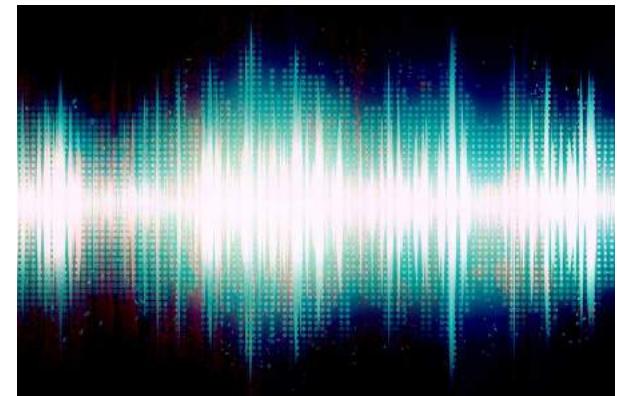


Photo Credit: <https://pixabay.com/illustrations/speech-icon-voice-talking-audio-2797263/>
Photo Credit: <https://pixabay.com/illustrations/sound-audio-waves-equalizer-495859/>

SAGEMAKER SEQUENCE-TO-SEQUENCE: HOW DOES IT WORK?

- Seq2Seq consists of the following layers:
 - **Embedding layer:** the sparse one hot encoded input is being mapped to dense feature layer.
 - **Encoder layer:**
 - Consists of LSTM or GRU
 - It takes input sequence from previous layer and compress the information into a fixed-length feature vector.
 - **Decoder layer:**
 - Consists of RNN and LSTM
 - Decoder layer takes this encoded feature vector and generates output sequence of tokens.

SAGEMAKER SEQUENCE-TO-SEQUENCE: HOW DOES IT WORK?

- Seq2Seq is based on Attention mechanism.
- Attention mechanism has been developed to overcome the issues exist with the traditional encoder-decoder framework.
- As the length of the dataset increases, the performance of the network becomes poor.
- This is because of the limit of how much information the fixed-length encoded feature vector can contain.
- Attention mechanism overcomes these problems by enabling the decoder to locate the most important information in the encoder so it could predict the next token in the sequence.

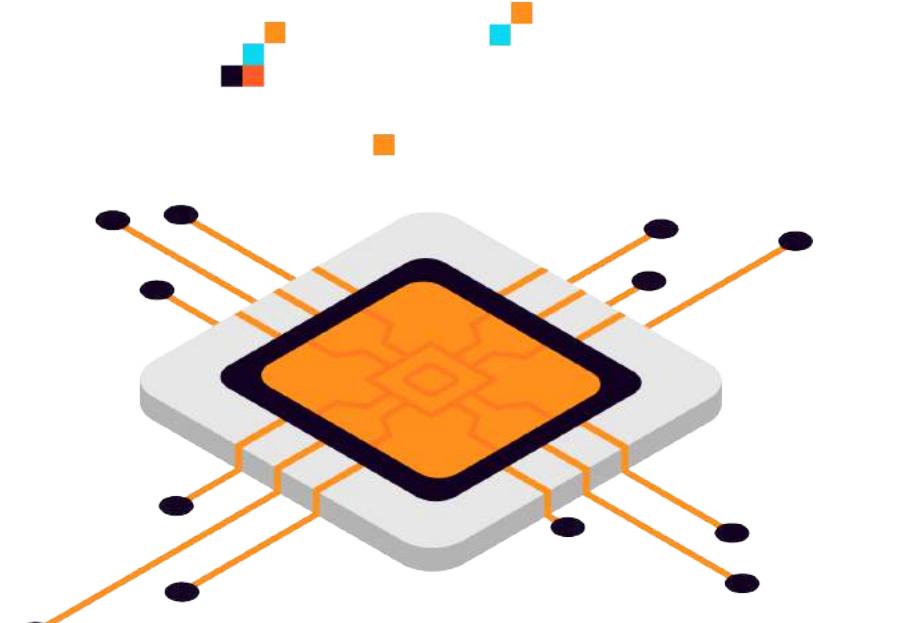
SAGEMAKER SEQUENCE-TO-SEQUENCE: INPUT/OUTPUT

- Seq2seq accepts data in RecordIO-Protobuf format.
- Tokens must be integers format which is an exception from the floating point formatting norm.
- You can use a readily available script by SageMaker that converts from tokenized text files to protobuf.
- After you complete preprocessing stage, the algorithm can be called to kick off the training process.
- The algorithm expects 3 channels as follows:
 - **train**: contains training data (train.rec file generated by preprocessing script).
 - **validation**: contains validation data (val.rec file generated by preprocessing script).
 - **vocab**: contains two vocabulary files (vocab.src.json and vocab.trg.json)



SAGEMAKER SEQUENCE-TO-SEQUENCE: EC2 INSTANCES

- Currently Amazon SageMaker seq2seq is only supported on GPU instance types
- SageMaker is only set up to train on a single machine.
- But it does also offer support for multiple GPUs.



SAGEMAKER SEQUENCE-TO-SEQUENCE: HYPERPARAMETERS

- Check out the full list of parameters here:
<https://docs.aws.amazon.com/sagemaker/latest/dg/seq-2-seq-hyperparameters.html>
- Batch_size: Mini batch size for gradient descent
- cnn_activation_type: CNN activation function type.
- encoder_type: Encoder type. RNN architecture/CNN Architectures
- Optimizer_type (adam, sgd, rmsprop)
- Learning_rate: Initial learning rate.
- Num_layers_encoder: Number of layers for Encoder *rnn* or *cnn*.
- Num_layers_decoder: Number of layers for Decoder *rnn* or *cnn*.



DEEPAR



DEEPAR: OVERVIEW

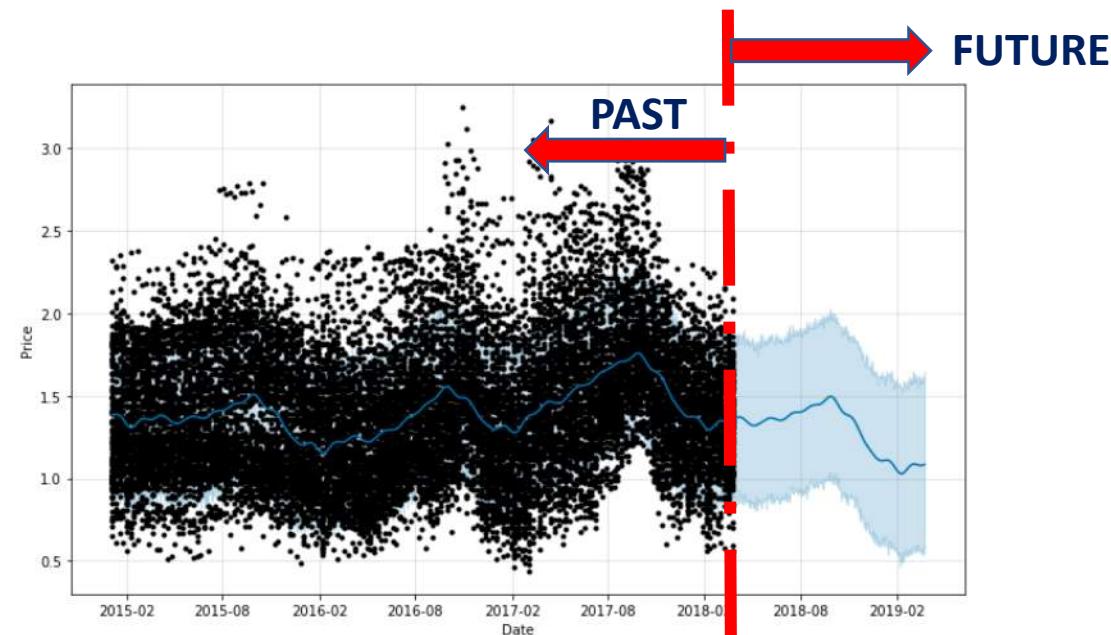
- Amazon SageMaker DeepAR is a one dimensional time series forecasting algorithm
- It works in supervised fashion and uses recurrent neural networks (RNN).
- DeepAR works well with forecasting timeseries that has seasonality.
- Example: avocado price predictions, stock price forecasting.
- DeepAR can work well in challenging problems when corporates are introducing totally new product to the market with not history (Cold Start Problem).

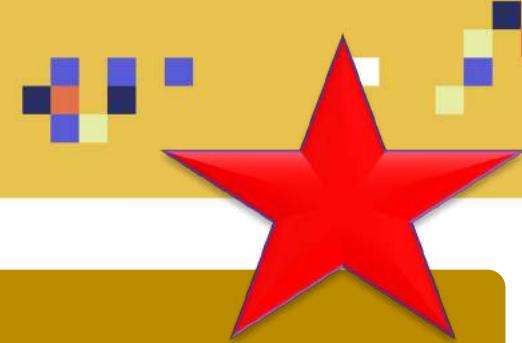


DEEPAR: OVERVIEW



- DeepAR outperforms classical forecasting techniques such as autoregressive integrated moving average (ARIMA) or exponential smoothing (ETS) in cases where multiple time series forecasting is present.
- DeepAR Allows the training of a single model jointly over multiple related time series (Example: several products demand forecast).
- It could be used to generate point forecast (Number of products to be sold next month is 10,000) and probabilistic forecast (Number of products to be sold next month is between 50,000 and 100,000 with 80% probability).





PRODUCT DEMAND PLANNING

- DeepAR could be used to forecast inventory levels.
- By feeding in the algorithm with historical of sales, promotions and outlet locations along with weather, website traffic, the algorithm will train a model to generate accurate product demand forecasts.
- Doing so will empower companies to properly stock inventory in various store locations in anticipation of forecasted demand.

FINANCIAL PLANNING

- DeepAR could be used to accurately predict company's financial information such as revenue, sales, expenses and cash flow.

RESOURCE PLANNING

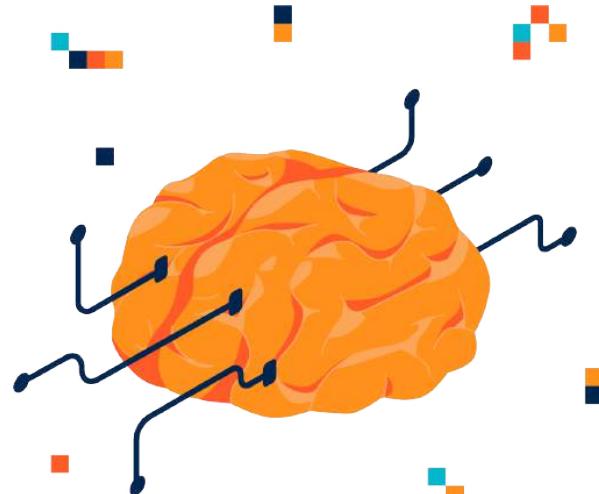
- DeepAR could be used to make predictions related to number of employees, raw materials, advertising, and revenue.



DEEPAR: INPUT/OUTPUT

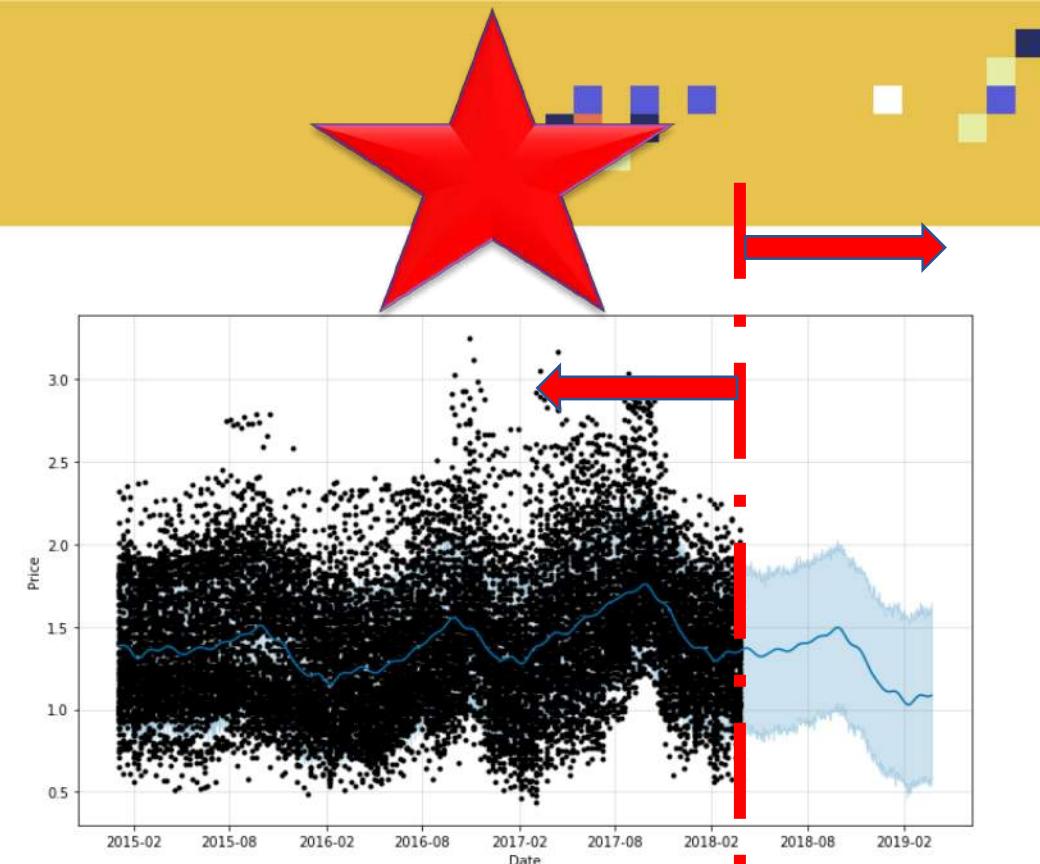
- DeepAR could support two data channels: (1) training and (2) testing
- The testing data channel is optional and could be used for evalution.
- Supported file formats:
 - JSON
 - gzip
 - Parquet
- Data must contain the following:
 - Start: starting time stamp with the format YYYY-MM-DD HH:MM:SS.
 - Target: floating point representing the time series values
 - Dynamic_feat (optional): dynamic features indicates if a promotion was applied to a product or not
 - Cat (optional): if there is categorical features

```
{"start": ..., "target": [1, 5, 10, 2], "dynamic_feat":  
[[0, 1, 1, 0]]}
```



DEEPAR: BEST PRACTICES

- When you apply DeepAR, it is recommended that you provide the entire time series for both training and testing.
- Even during model inference, provide the entire time series.
- Do the following:
 - Use the entire dataset as a testing dataset
 - Remove the last prediction_length points from each time series for training.
- Avoid lengthy prediction time (don't use prediction length>400) which will result in large prediction errors.
- DeepAR works well with multiple time series and starts to outperform the ARIMA and ETS.
- For single time series, ARIMA and ETS works better compared to DeepAR



TRAINING PREDICTION LENGTH

TESTING

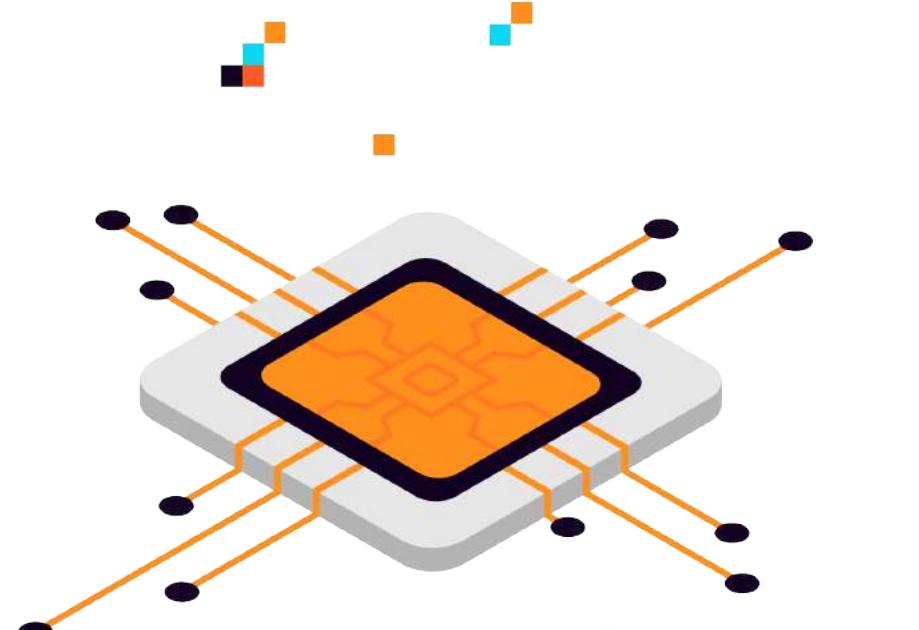
DEEPAR: HYPERPARAMETERS

- Check out the full list of hyperparameters here:
https://docs.aws.amazon.com/sagemaker/latest/dg/deepar_hyperparameters.html
- Context_length: The number of time-points that the model gets to see before making the prediction.
- Prediction_length: The number of time-steps that the model is trained to predict
- dropout_rate: The dropout rate to use during training.
- num_dynamic_feat: The number of dynamic_feat provided in the data.
- Epochs
- mini_batch_size
- Learning_rate
- Num_cells

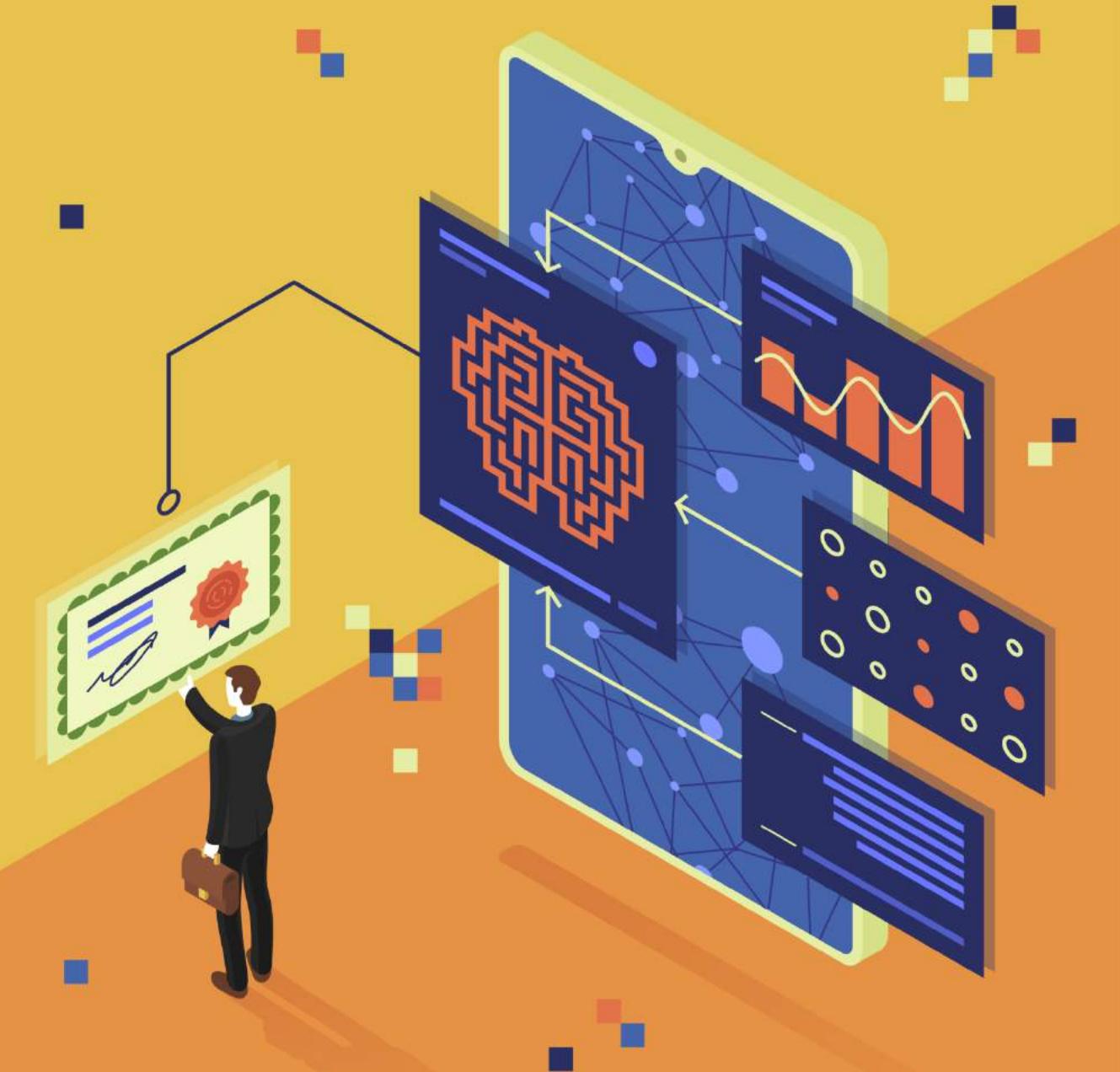


DEEPAR: EC2 INSTANCE TYPES FOR DEEPAR

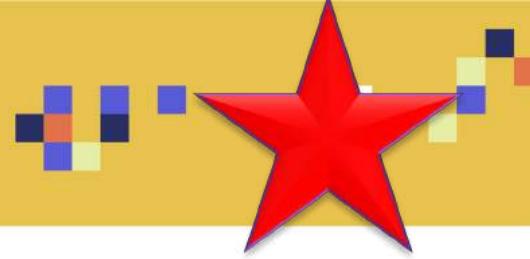
- DeepAR can be trained on both GPU and CPU instances
- DeepAR can be trained in both single and multi-machine settings.
- SageMaker recommends starting out with a single CPU instance (ml.c4.2xlarge or ml.c4.4xlarge)
- Then switch to GPU instances and multiple machines only when:
 - The model is large which occurs when Specifying large values for context_length, prediction_length, num_cells, num_layers, or mini_batch_size
 - During hyperparameters optimization
- DeepAR can only run on CPU instances during inference.



BLAZING TEXT



BLAZINGTEXT: OVERVIEW



- The Amazon SageMaker BlazingText algorithm consists of the following optimized algorithms:

1. Word2vec:

- Word2vec is used to generate a word embedding
- Word2vec algorithm is critical in Natural Language Processing (NLP) applications
- The Word2vec algorithm maps words to distributed vectors
- Words that are semantically similar correspond to vectors that are close together
- word embeddings is capable for capturing the semantic relationships between words
- Skip-gram, batch skip-gram, and continuous bag-of-words (CBOW)
- Capable of training a model on a billion words in minutes

2. Text classification:

- Text classification is critical in: web searches, information retrieval, ranking, and document classification.

This is one of the best courses

I have seen on Udemy!

I really hate this course



1



0

BLAZINGTEXT: WORD2VEC DEEPPDIVE



- Word2vec was introduced in 2013 and made huge leaps in NLP.
- Since computers only deal with numbers and not text, Word2vec creates a word embedding which is an embedded version of the words that could be fed to a computer.
- Word2vec preserves the relationship between words.
- Word2vec is a shallow simple neural network with a single hidden layer that converts words into vectors

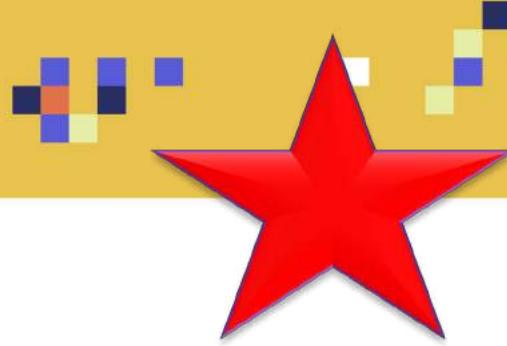


KING-MAN+WOMAN = QUEEN

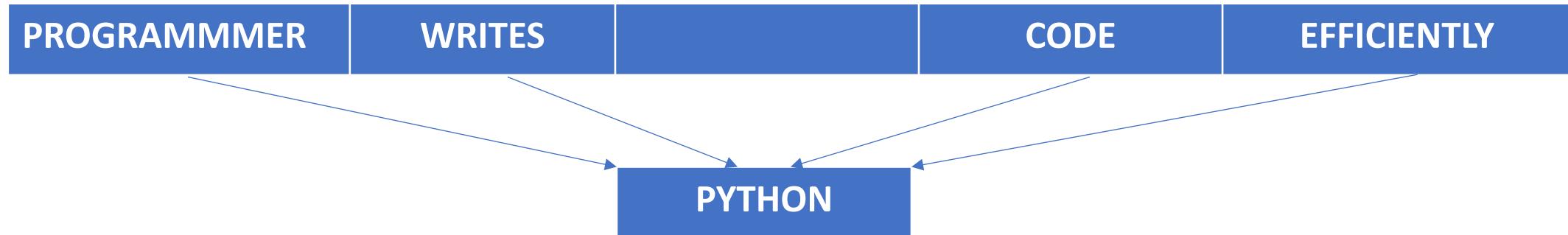
*programmer writes code in python
developer writes code in python*

Programmer and developer should have similar embedding vector

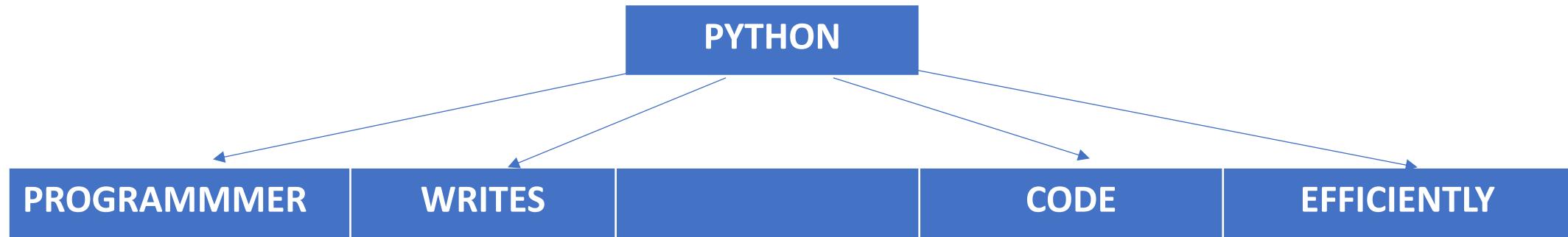
BLAZINGTEXT: WORD2VEC DEEPDIVE – CONTINIOUS BAG OF WORDS (CBOW) Vs. SKIMGRAM



- CBOW: predict the target word from the sentence context



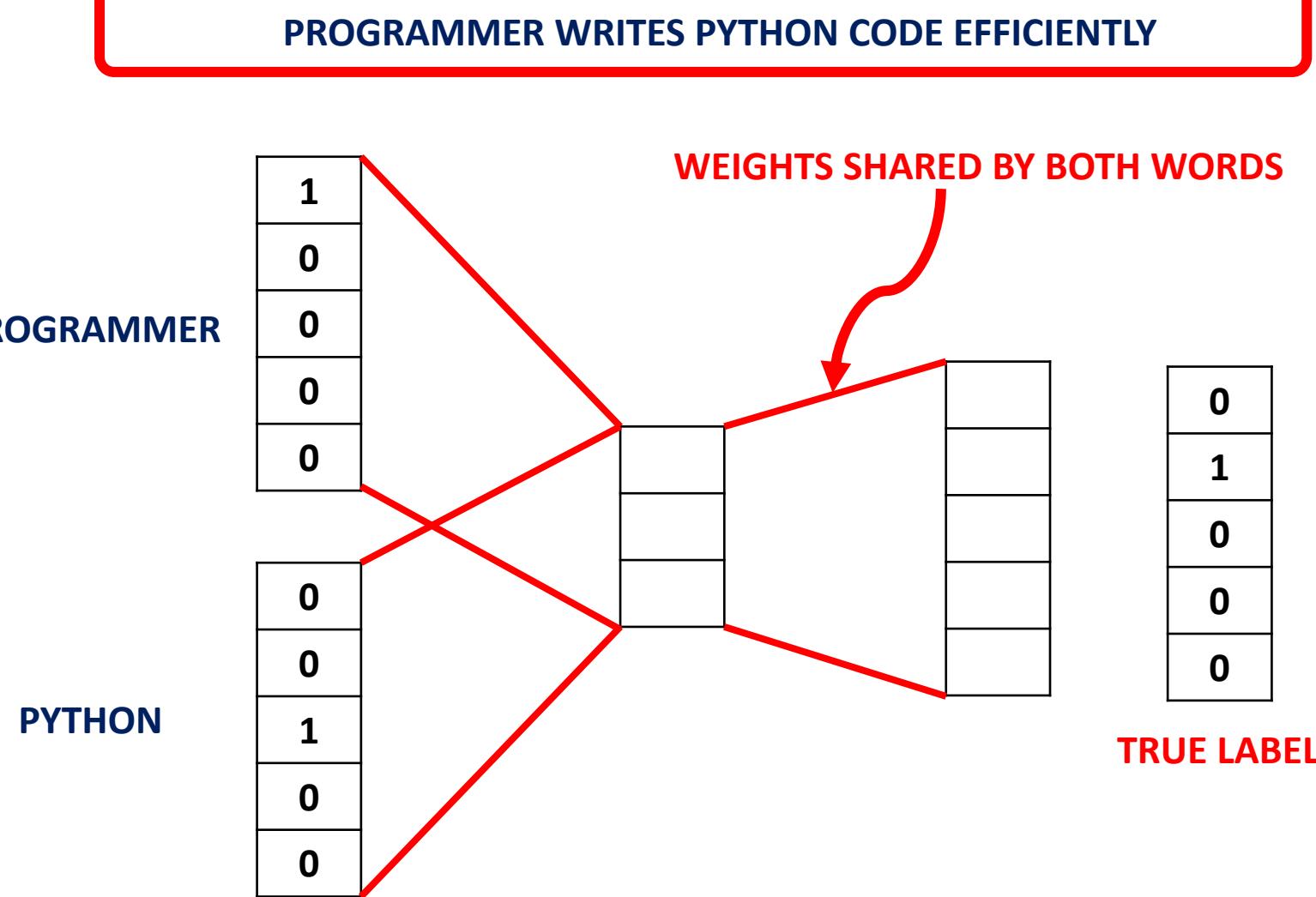
- Skimgram: we try to do the opposite, we try to predict the context from the target word.



BLAZINGTEXT: WORD2VEC DEEPPDIVE – CONTINUOUS BAG OF WORDS



- One hot encoding is performed first.
- Train an artificial neural network where number of inputs represents the total number of words.
- The number of outputs represent the length of the vector.
- The weight matrix represents the set of the vectors.



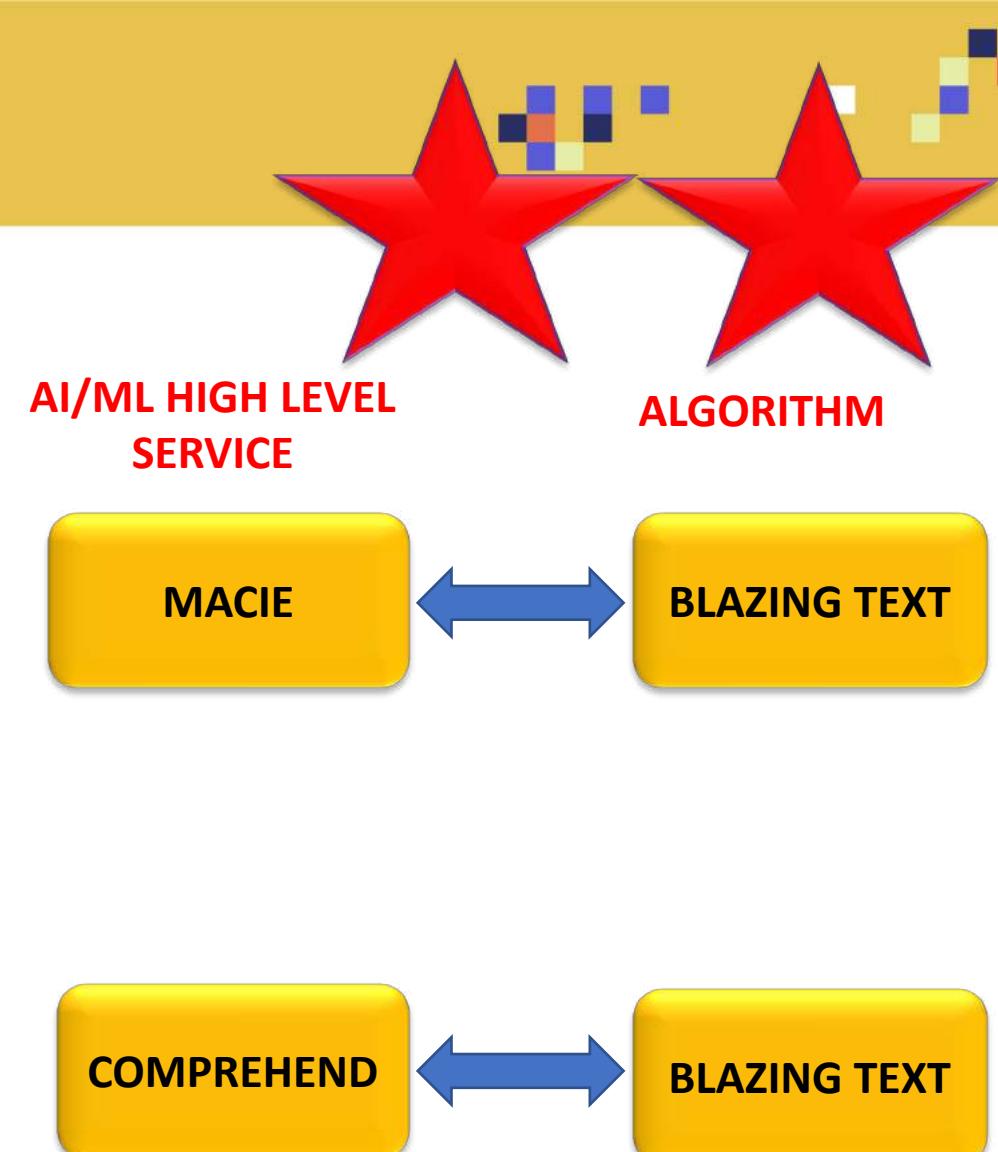
BLAZINGTEXT: USE CASES

- **Document Classification:**

- Scan through a corpus of documents and classify them as follows:
 - (1) Contains sensitive data
 - (2) Does not contain sensitive information.
- Note that Amazon provides a high level service called Macie that could do this.
- **Amazon Macie** is a security service that automatically discovers and classifies sensitive data.

- **Sentiment Analysis:**

- Blazing text could be used to perform sentiment analysis by scanning through customer tweets, Facebook posts, and reviews and assess whether customers are happy or not.
- Note that Amazon provides a high level service called Amazon Comprehend that could provide the same service.



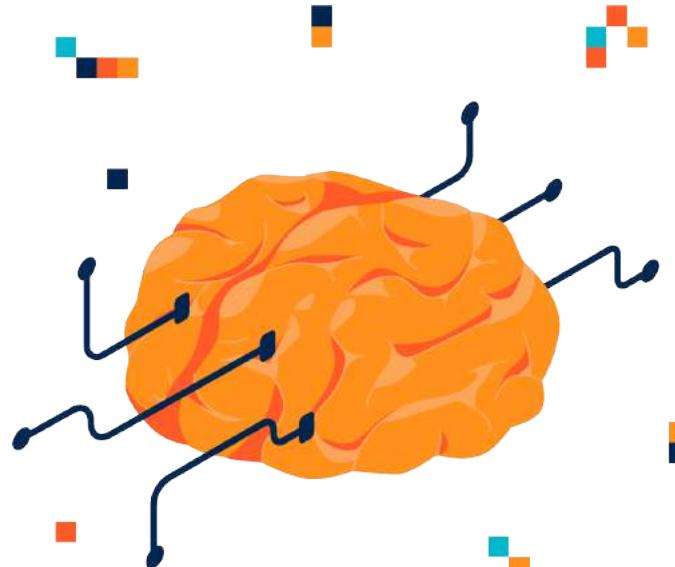
BLAZINGTEXT: INPUT/OUTPUT

- The BlazingText algorithm takes in a text file with space-separated tokens.
- In the text file, each line contains a single sentence.
- For supervised mode (text classification):
 - Files should contain a training sentence per line along with the labels.
 - Labels are prefixed by the string `_label_`.

`_label_4 linux ready for prime time , intel says , despite all the linux hype , the open-source movement has yet to make a huge splash in the desktop market . that may be about to change , thanks to chipmaking giant intel corp .`

`_label_2 bowled by the slower one again , kolkata , november 14 the past caught up with sourav ganguly as the indian skipper's return to international cricket was short lived .`

- For unsupervised Blazingtext (Word2vec), it expects a text file with one training sentence per line and no label.



BLAZINGTEXT: HYPERPARAMETERS

For the full list of hyperparameters, check this out:

https://docs.aws.amazon.com/sagemaker/latest/dg/blazingtext_hyperparameters.html

Word2Vec Hyperparameters:

- Mode: The Word2vec architecture used for training such as: batch_skipgram, skipgram, or cbow
- batch_size: The size of each batch when mode is set to batch_skipgram.
- Epochs
- Learning rate

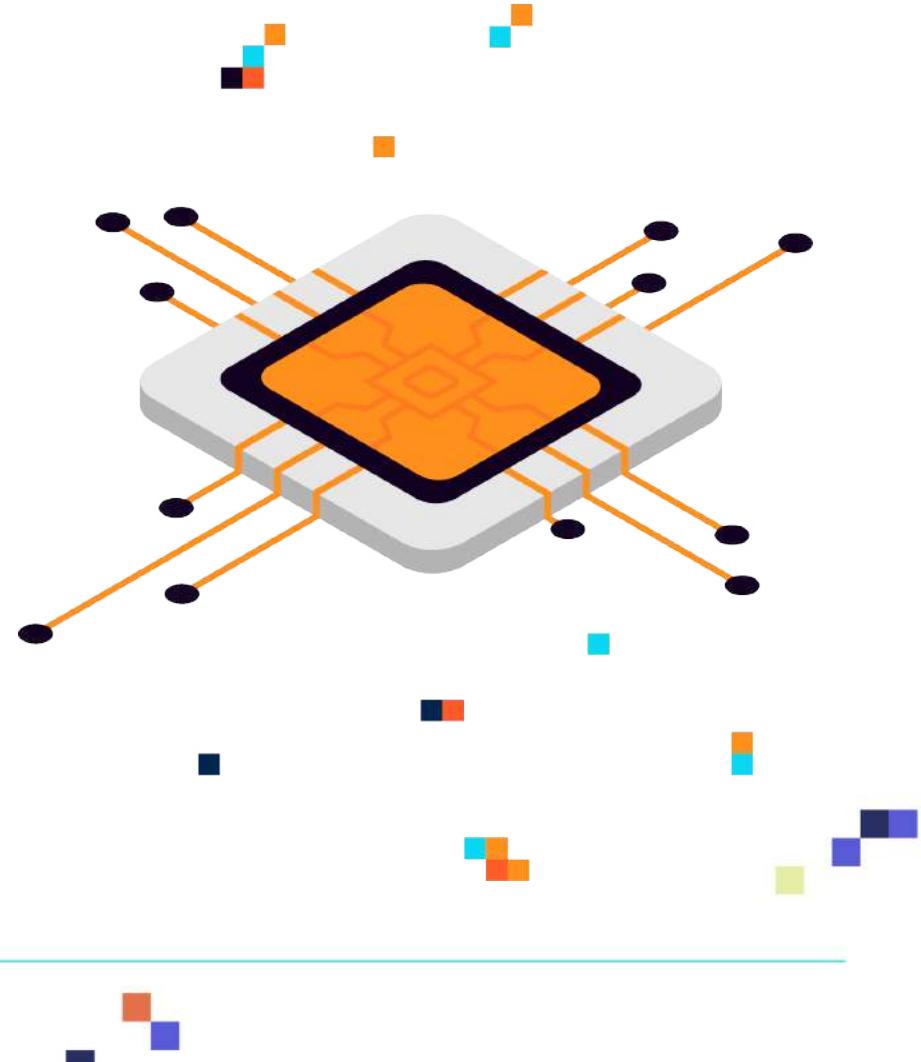
Text Classification Hyperparameters:

- Mode: The training mode, valid values: supervised
- early_stopping: Whether to stop training if validation accuracy doesn't improve after a patience number of epochs.
- Epochs
- Learning rate
- vector_dim: The dimension of the embedding layer
- word_ngrams: The number of word n-gram features to use.

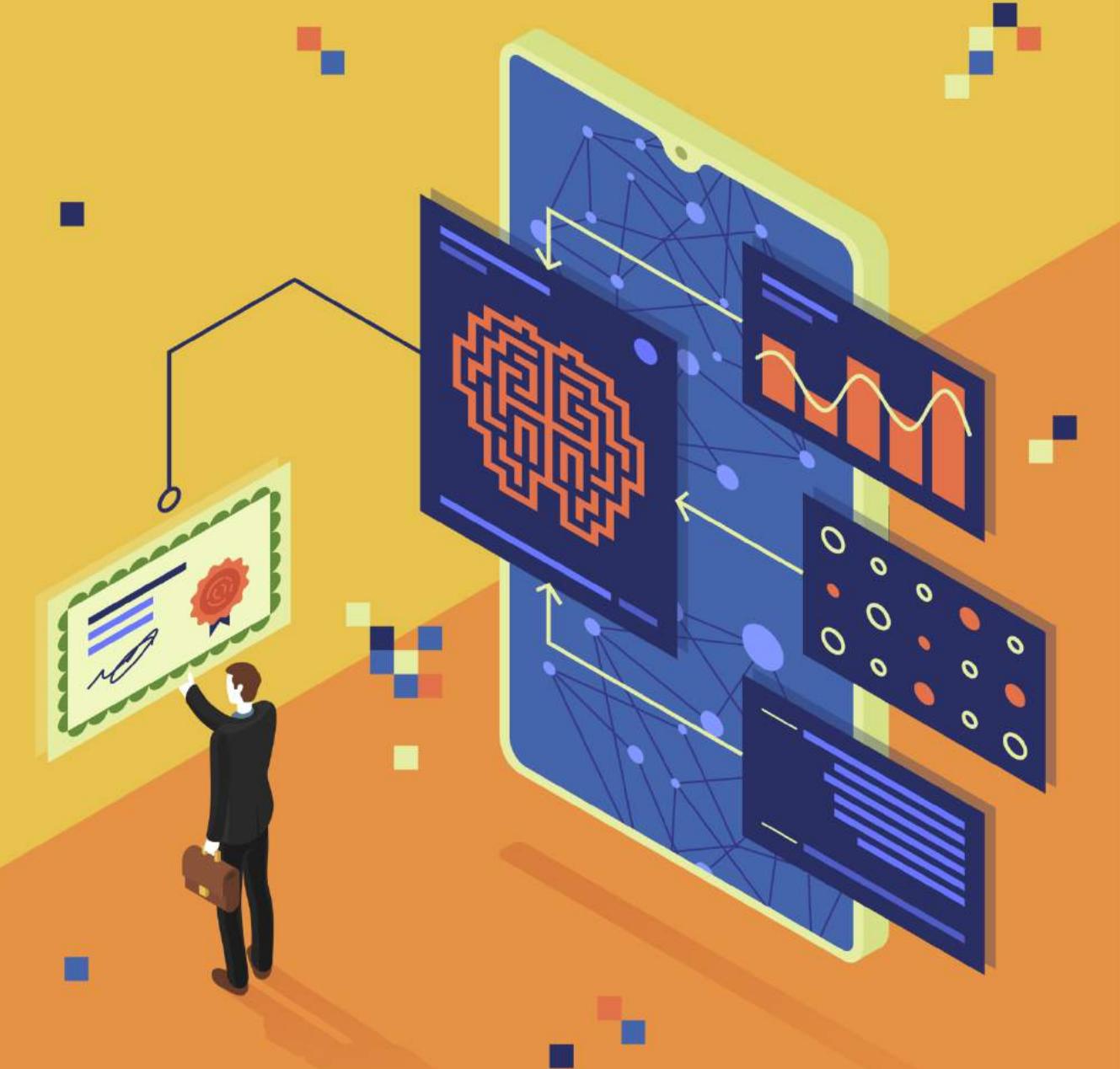


BLAZINGTEXT: EC2 INSTANCES

- For cbow and skipgram modes, BlazingText supports single CPU and single GPU instances.
- ml.p3.2xlarge instance is recommended.
- For batch_skipgram mode, BlazingText supports single or multiple CPU instances.
- For supervised text classification, C5 instance is recommended if the training dataset is less than 2 GB.
- For larger datasets, use single GPU (ml.p2.xlarge or ml.p3.2xlarge).



AWS MACHINE LEARNING CERTIFICATION



DOMAIN #3: MODELING (36% EXAM)



AWS ML CERTIFICATION EXAM DOMAINS



Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #3 OVERVIEW:

SECTION #8: MACHINE AND DEEP LEARNING BASICS – PART #1

- Artificial Neural Networks Basics: Single Neuron Model
- Activation Functions
- Multi-Layer Perceptron Model
- How do Artificial Neural Networks Train?
- ANN Parameters Tuning – Learning rate and batch size
- Tensorflow playground
- Gradient Descent and Backpropagation
- Overfitting and Under fitting
- How to overcome overfitting?
- Bias Variance Trade-off
- L1 Regularization
- L2 Regularization

SECTION #9: MACHINE AND DEEP LEARNING BASICS – PART #2

- Artificial Neural Networks Architectures
- Convolutional Neural Networks
- Recurrent Neural Networks
- Vanishing Gradient Problem
- LSTM Networks
- Model Performance Assessment – Confusion Matrix
- Model Performance Assessment – Precision, recall, F1-score
- Model Performance Assessment – ROC, AUC, Heatmap, and RMSE
- K-Fold Cross validation
- Transfer Learning
- Ensemble Learning – Bagging and Boosting

DOMAIN #3 OVERVIEW:

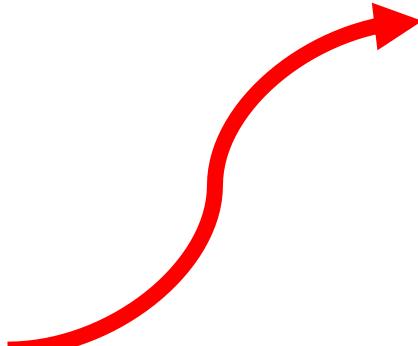


SECTION #10: MACHINE AND DEEP LEARNING IN AWS – BUILT-IN ALGORITHMS PART #1

- AWS SageMaker
- Deep Learning on AWS
- SageMaker Built-in algorithms
- Object Detection
- Image Classification
- Semantic Segmentation
- SageMaker Linear Learner
- Factorization Machines
- XG-Boost
- SageMaker Seq2Seq
- SageMaker DeepAR
- SageMaker Blazing Text

SECTION #11: MACHINE AND DEEP LEARNING IN AWS – BUILT-IN ALGORITHMS PART #2

- Random Cut Forest
- Neural Topic Model
- LDA
- K-Nearest Neighbours (KNN)
- K Means
- Principal Component Analysis (PCA)
- IP insights
- Reinforcement Learning
- Object2Vec
- Automatic Model Tuning



WE ARE HERE!

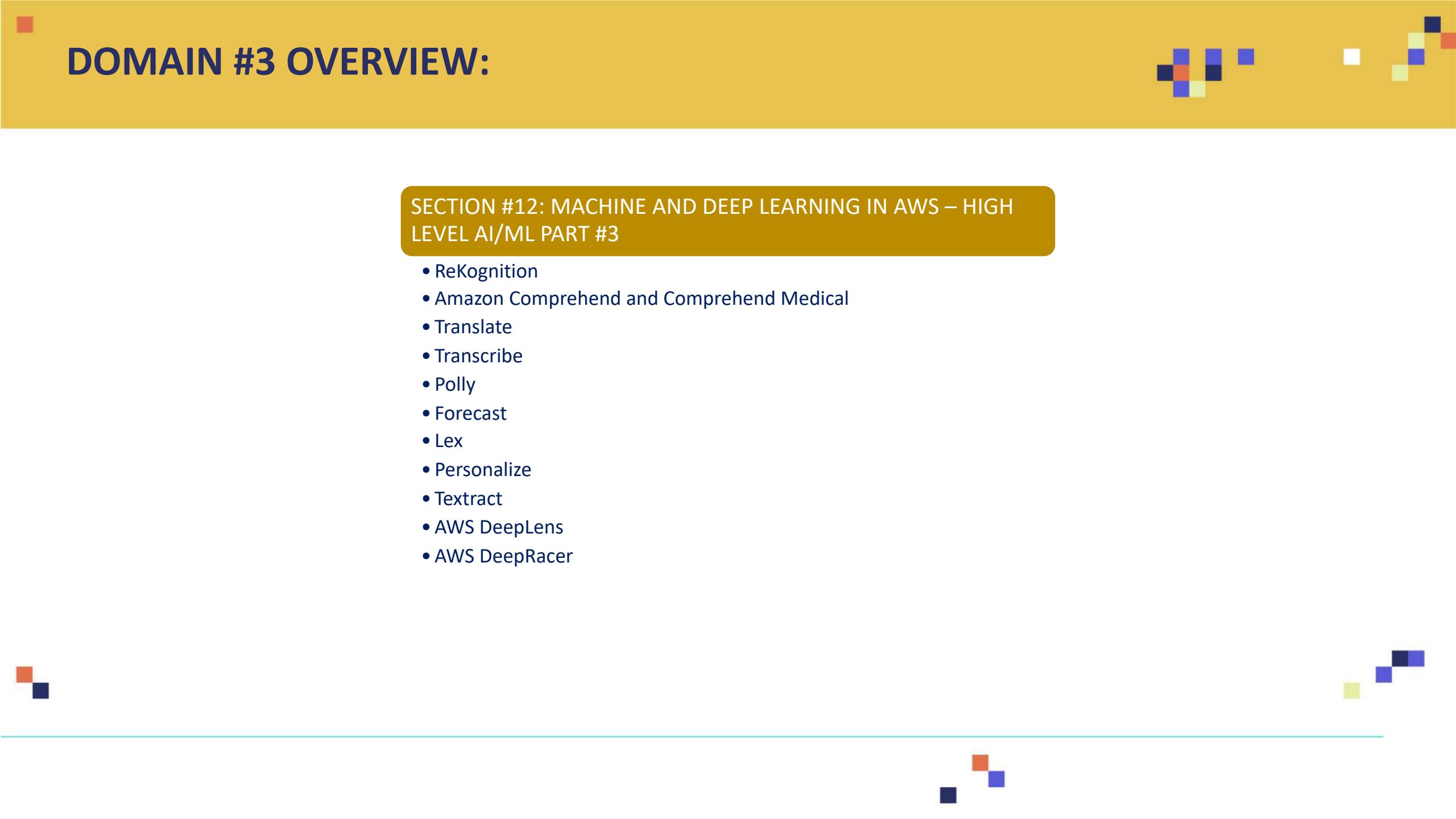


DOMAIN #3 OVERVIEW:



SECTION #12: MACHINE AND DEEP LEARNING IN AWS – HIGH LEVEL AI/ML PART #3

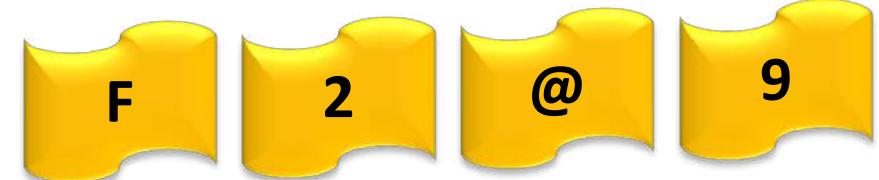
- ReKognition
- Amazon Comprehend and Comprehend Medical
- Translate
- Transcribe
- Polly
- Forecast
- Lex
- Personalize
- Textract
- AWS DeepLens
- AWS DeepRacer



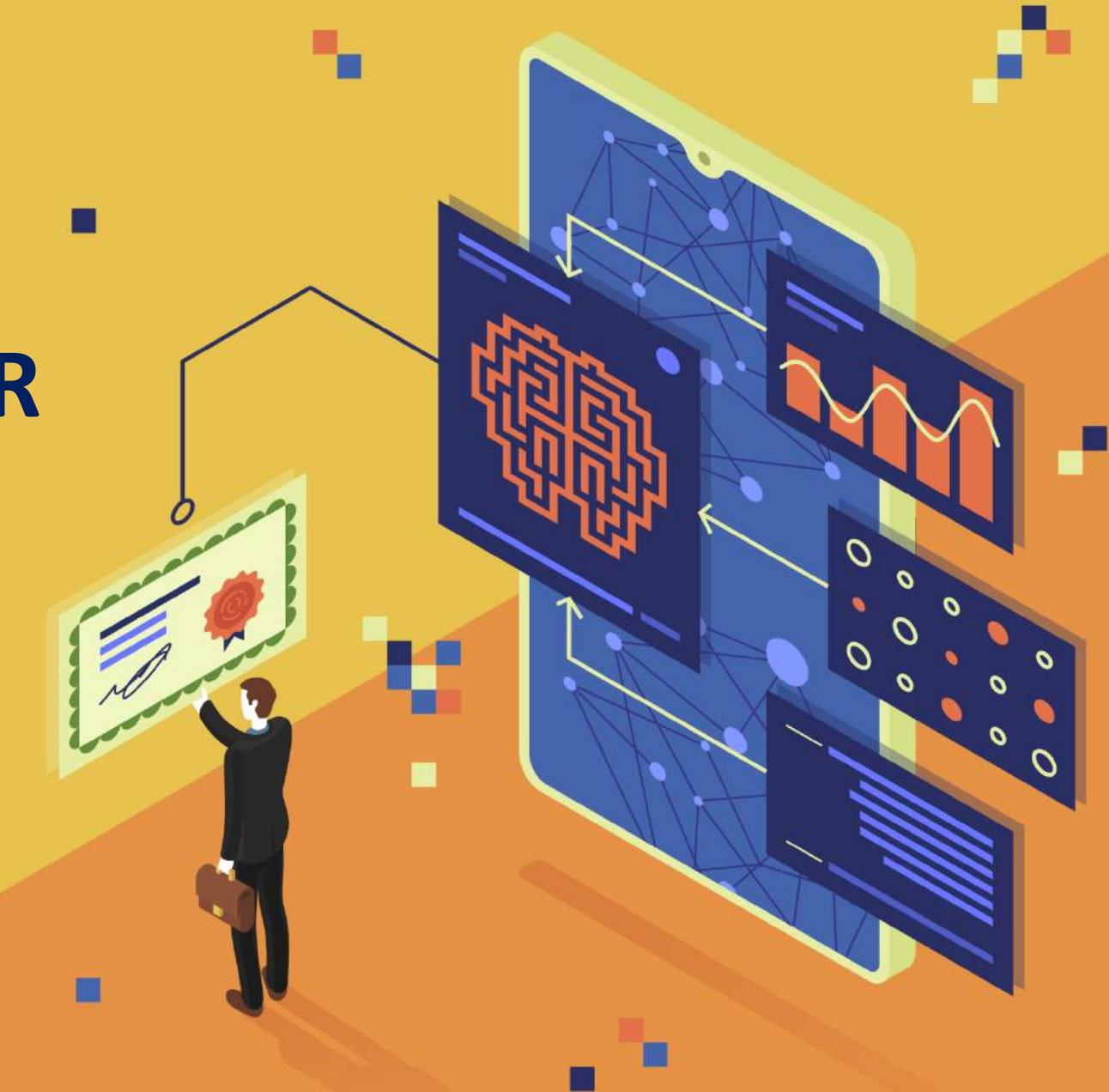
RECALL OUR MINI CHALLENGE AND PRIZE!



- For those of you who will successfully complete the entire section and watch the videos till the end, they will receive a valuable prize!



AWS SAGEMAKER



AWS SAGEMAKER BUILT-IN ALGORITHMS



SAGEMAKER AVAILABLE ALGORITHMS



BlazingText Word2Vec	BlazingText implementation of the Word2Vec algorithm for scaling and accelerating the generation of word embeddings from a large number of documents.
DeepAR	An algorithm that generates accurate forecasts by learning patterns from many related time-series using recurrent neural networks (RNN).
Factorization Machines	A model with the ability to estimate all of the interactions between features even with a very small amount of data.
Gradient Boosted Trees (XGBoost)	Short for “Extreme Gradient Boosting”, XGBoost is an optimized distributed gradient boosting library.
Image Classification (ResNet)	A popular neural network for developing image classification systems.
IP Insights	An algorithm to detect malicious users or learn usage patterns of IP addresses.
K-Means Clustering	One of the simplest ML algorithms. It's used to find groups within unlabeled data.
K-Nearest Neighbor (k-NN)	An index based algorithm to address classification and regression based problems.
Latent Dirichlet Allocation (LDA)	A model that is well suited to automatically discovering the main topics present in a set of text files.
Linear Learner (Classification)	Linear classification uses an object's characteristics to identify the appropriate group that it belongs to.
Linear Learner (Regression)	Linear regression is used to predict the linear relationship between two variables.
Neural Topic Modelling (NTM)	A neural network based approach for learning topics from text and image datasets.
Object2Vec	A neural-embedding algorithm to compute nearest neighbors and to visualize natural clusters.
Object Detection	Detects, classifies, and places bounding boxes around multiple objects in an image.
Principal Component Analysis (PCA)	Often used in data pre-processing, this algorithm takes a table or matrix of many features and reduces it to a smaller number of representative features.
Random Cut Forest	An unsupervised machine learning algorithm for anomaly detection.
Semantic Segmentation	Partitions an image to identify places of interest by assigning a label to the individual pixels of the image.
Sequence2Sequence	A general-purpose encoder-decoder for text that is often used for machine translation, text summarization, etc.

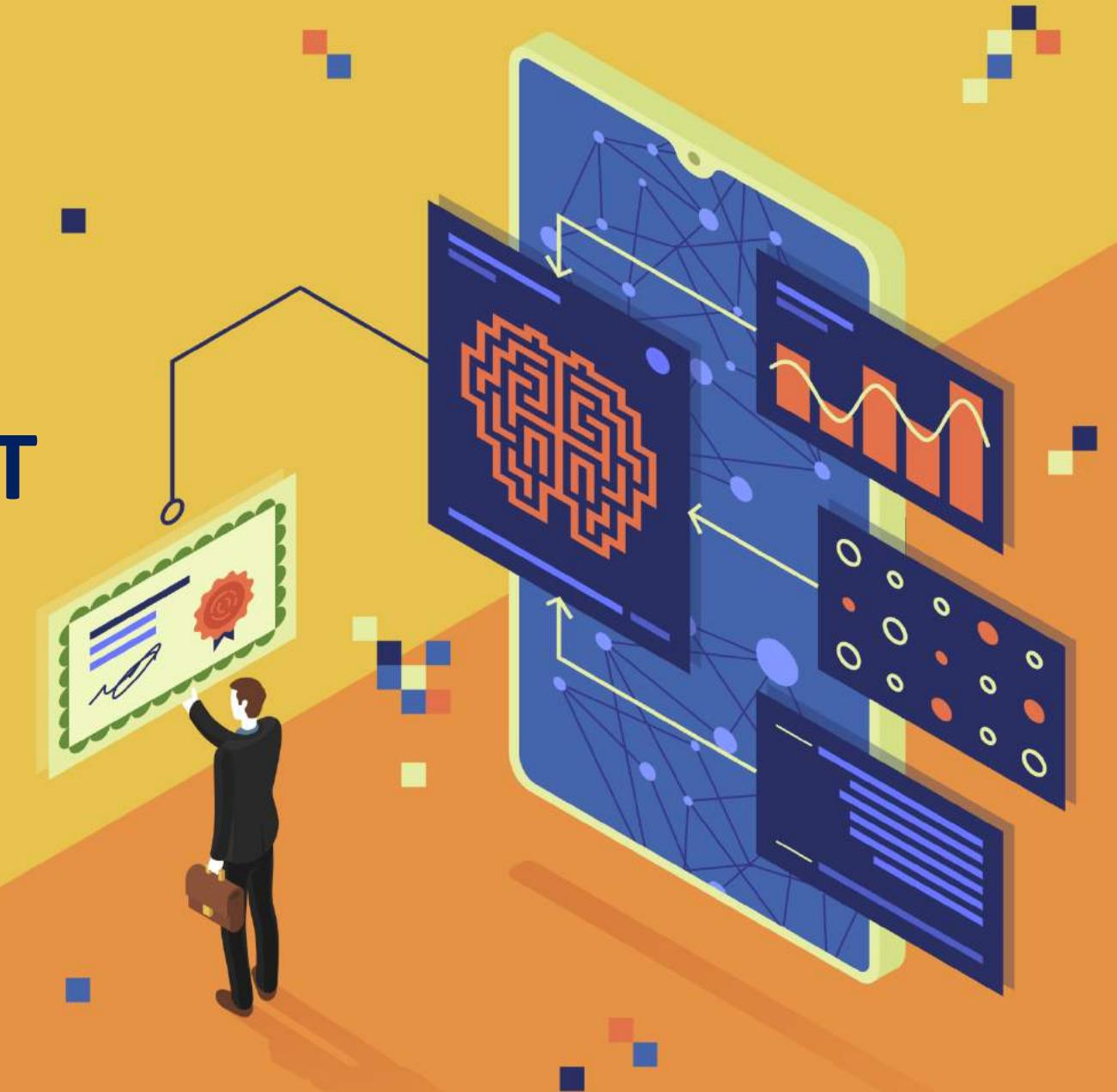
■ RECALL WHAT IS AN ALGORITHM? AND A HEURISTIC?



An Algorithm consists of multiple well defined steps to solve a given problem. An important feature of an algorithm is that the outcome should be repeatable across multiple runs.

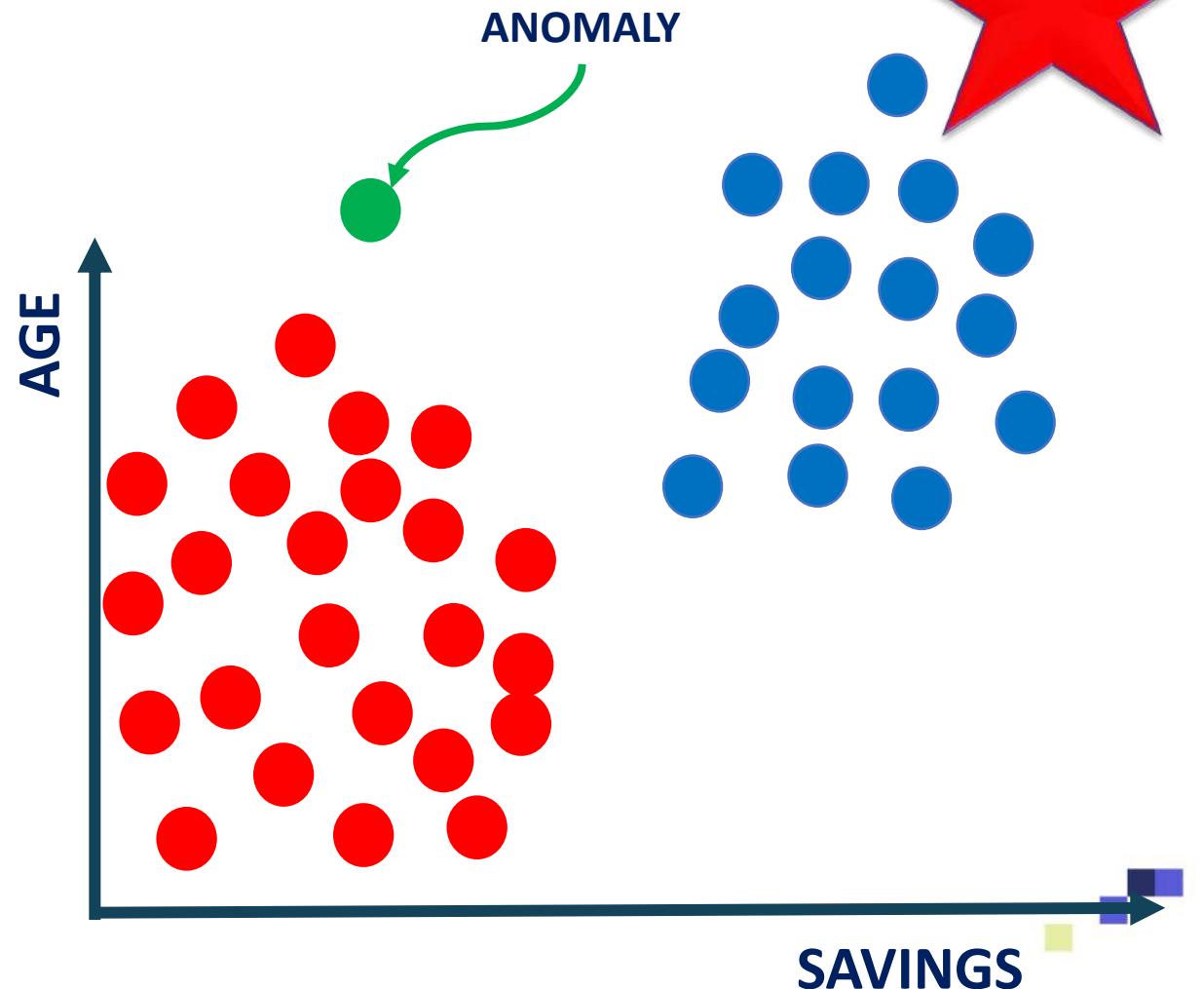
A heuristic: is an “educated guess”, it represents an acceptable solutions that’s generated much faster by applying “rule of thumb”. Heuristics do not guarantee a consistent outcome.

RANDOM CUT FOREST



RANDOM CUT FOREST (RCF): OVERVIEW

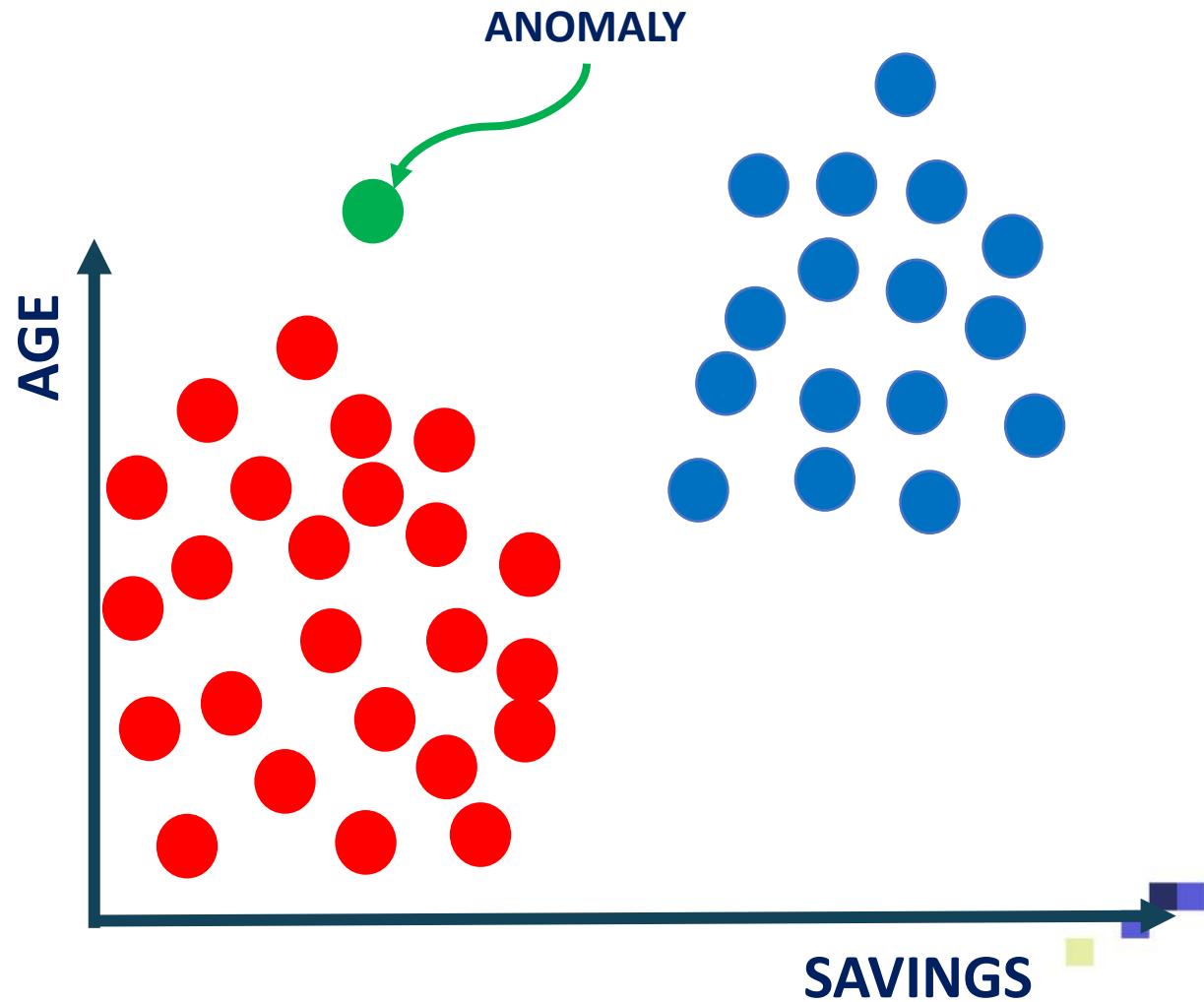
- Random Cut Forest algorithm will be on the exam!!
- Amazon SageMaker Random Cut Forest (RCF) is an unsupervised algorithm
- It could be used for anomaly detection.
- Anomalies are data points that diverge from the rest of the properly structured data.
- Anomalies could be spikes or breaks in the dataset and could be detected from the regular structured dataset.
- Anomaly detection and removal is crucial in machine learning because adding anomalies will unnecessarily increase the complexity of the model.



RANDOM CUT FOREST (RCF): OVERVIEW



- RCF algorithm assigns an anomaly score to each and every point in the dataset.
- If the score is more than 3 standard deviations, the anomaly score is considered high.
- The algorithm can work well with one and multi dimensional time series data.
- The algorithm works by sampling data randomly.
- The algorithm works by creating a forest of trees where each tree is a partition of the training data.
- The algorithms then examines the expected change in complexity of the tree after a point has been added to it.



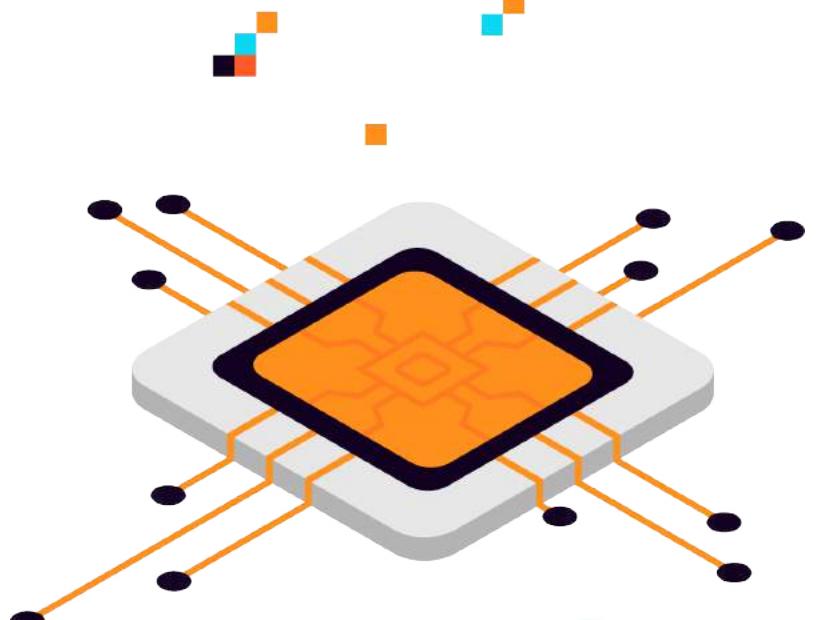
RANDOM CUT FOREST: INPUT/OUTPUT

- Amazon SageMaker Random Cut Forest require training channel and optional testing channels.
- Metrics such as accuracy, precision, recall, and F1-score can be calculated based on the testing dataset provided in the testing channel.
- The algorithm supports both File and Pipe modes
- Train and test data content types are:
 - application/x-recordio-protobuf
 - text/csv formats.
- During inference, RCF supports application/x-recordio-protobuf, text/csv and application/json.



RANDOM CUT FOREST: EC2 INSTANCE

- For RCF algorithm training, the following instances are recommended:
 - For training, ml.m4, ml.c4, and ml.c5 instances.
 - For inference, ml.c5.xl instance which will lead to maximum performance at an optimized cost.
- RCF could run on GPU instance types but it is not necessary.



RANDOM CUT FOREST: HYPERPARAMETERS

- feature_dim: number of features in the data set.
- eval_metrics: A list of metrics used to score a labeled test data set such as accuracy, positive and negative precision, recall, and F1-scores.
- num_samples_per_tree: Number of random samples given to each tree from the training data set.
- num_trees: Number of trees in the forest.

NOTES:

- num_trees and num_samples_per_tree are the most important parameters in the RCF algorithm.
- SageMaker recommends using 100 trees, this will strike a good balance between model complexity and noise.
- As the number of num_trees increases, the noise observed in anomaly scores is reduced because the final outcome is the average of all the output from all the trees.
- num_samples_per_tree should be chosen such that $1/\text{num_samples_per_tree}$ equals to the ratio of anomalous to normal data.
- Example: if in each tree, there is 256 samples then the data will contain $1/256$ (0.4%) anomalies.



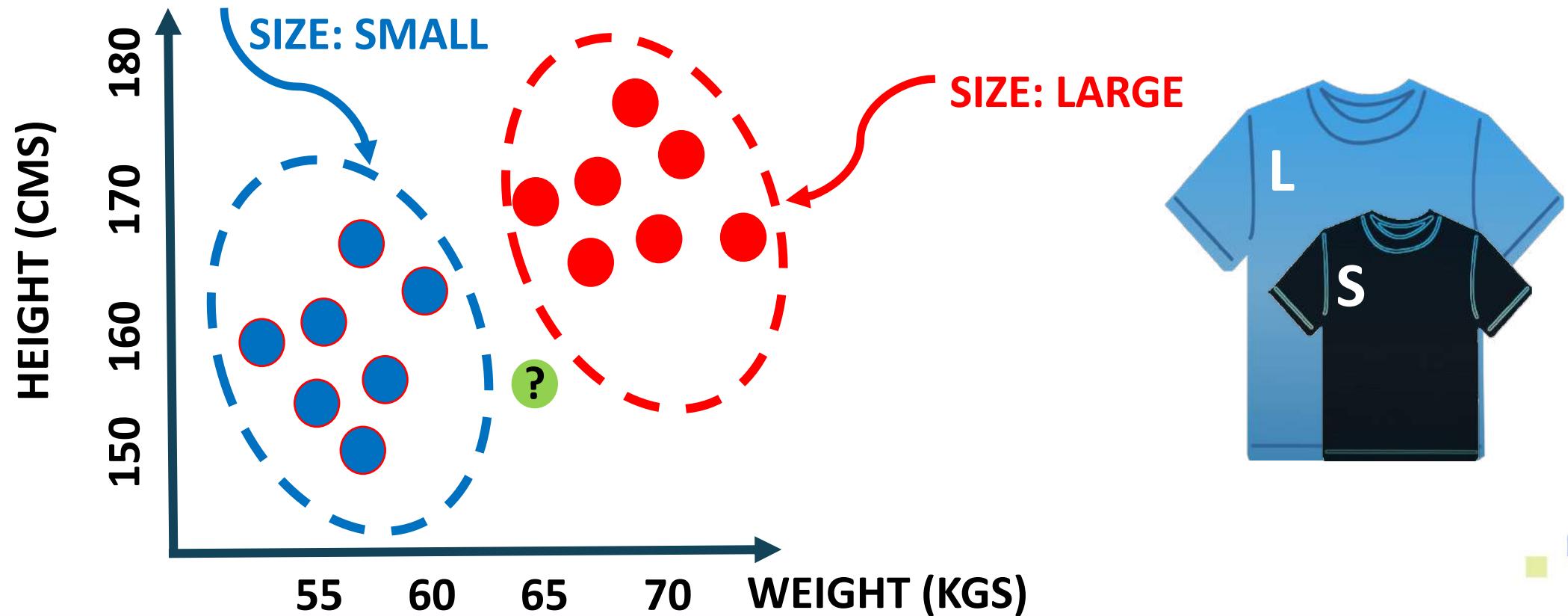
K NEAREST NEIGHBORS (KNN)



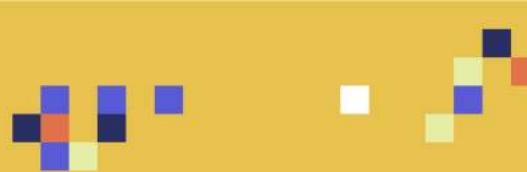
K NEAREST NEIGHBORS (KNN): INTUITION



- k-nearest neighbors algorithm (KNN) works by finding the **most similar** data points in the training data, and attempt to make an **educated guess** based on their classifications.

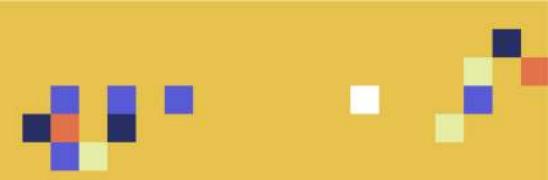


K NEAREST NEIGHBORS (KNN): ALGORITHM STEPS

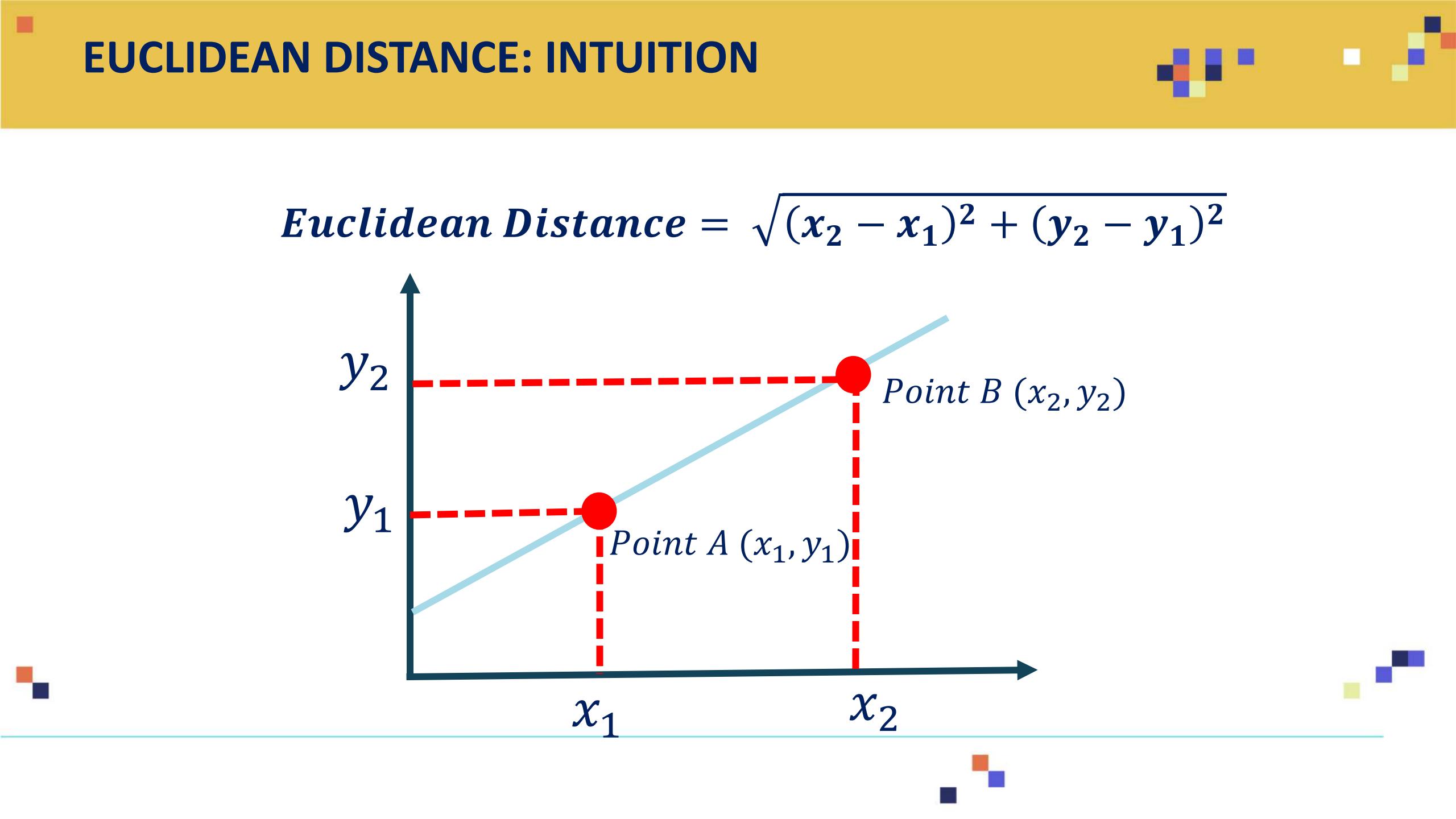
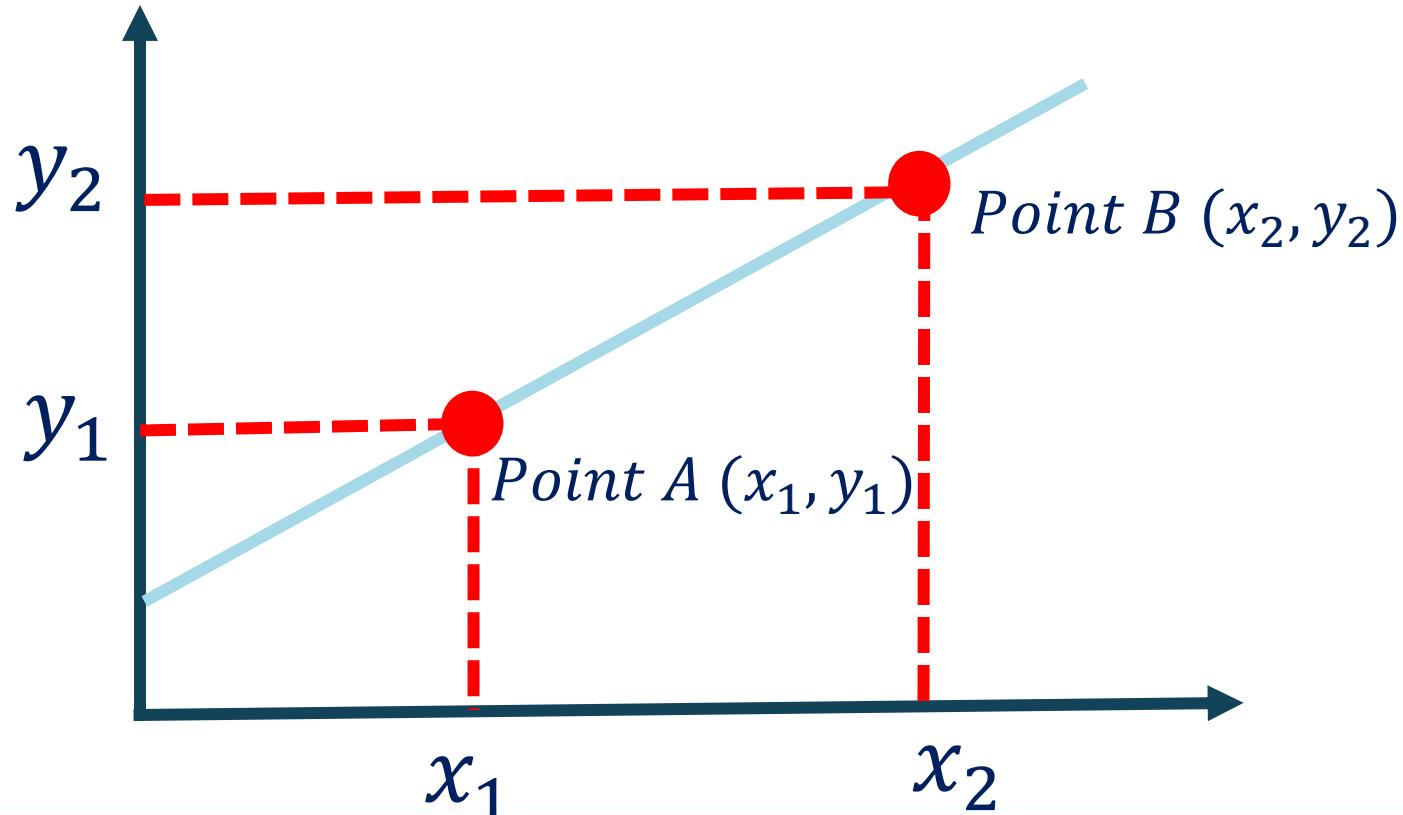


- Select a value for k (e.g.: 1, 2, 3, 10..)
- Calculate the Euclidian distance between the point to be classified and every other point in the training data-set
- Pick the k closest data points (points with the k smallest distances)
- Run a majority vote among selected data points, the dominating classification is the winner! Point is classified based on the dominant class.
- Repeat!

EUCLIDEAN DISTANCE: INTUITION



$$\textit{Euclidean Distance} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$



K NEAREST NEIGHBORS (KNN): EXAMPLE



- KNN will look for the 5 data points that are closest to the new customer data point
- The algorithm will determine which category (class) are these 5 points in.
- Since 4 points had class ‘Small’ and 1 had ‘Large’, then new customer shall be assigned Small size.



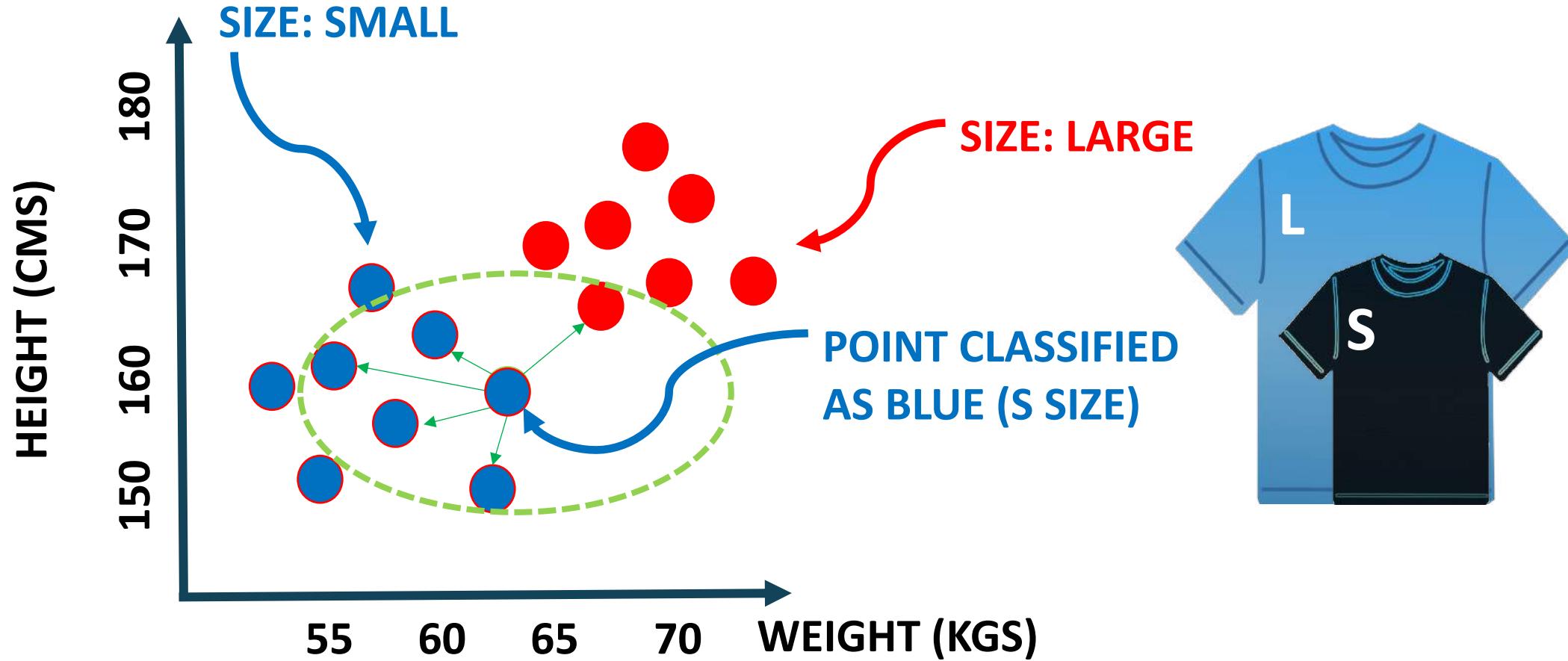
Height (in cms)	Weight (in kgs)	T Shirt Size	Eucledian Distance	Vote
158	58	S	4.242640687	
158	59	S	3.605551275	
158	63	S	3.605551275	
160	59	S	2.236067977	3
160	60	S	1.414213562	1
163	60	S	2.236067977	3
163	61	S	2	2
160	64	L	3.16227766	5
163	64	L	3.605551275	
165	61	L	4	
165	62	L	4.123105626	
165	65	L	5.656854249	
168	62	L	7.071067812	
168	63	L	7.280109889	
168	66	L	8.602325267	
170	63	L	9.219544457	
170	64	L	9.486832981	
170	68	L	11.40175425	
New Customer Information				
161	61			
Assume k = 5				



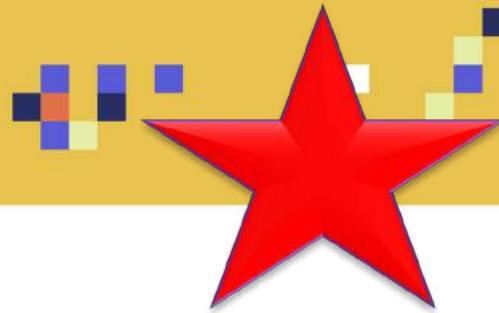
K NEAREST NEIGHBORS (KNN): EXAMPLE



- Let's understand this example visually!



K NEAREST NEIGHBORS (KNN) IN SAGEMAKER



- KNN in SageMaker could be used to perform simple classification or regression
 - Classification: algorithm finds the K-closest points to a given sample point and return the most frequent label
 - Regression: algorithm finds K-closest points to a given sample point and return the average value.
- KNN is a lazy algorithm, it does not try to generalize the model for the entire training dataset but it relies on neighbouring data points.
- KNN is used for grouping some customers based on their credit risk or perform recommendations.
- Training with the KNN algorithm has three steps:
 - Sampling
 - Dimension reduction
 - Index building
- Sampling is used to minimize the size of dataset to optimize memory.
- Dimensionality reduction is performed to:
 - Decrease the feature dimension of the data to reduce the footprint of the k-NN model in memory and inference latency and avoids the “curse of dimensionality”

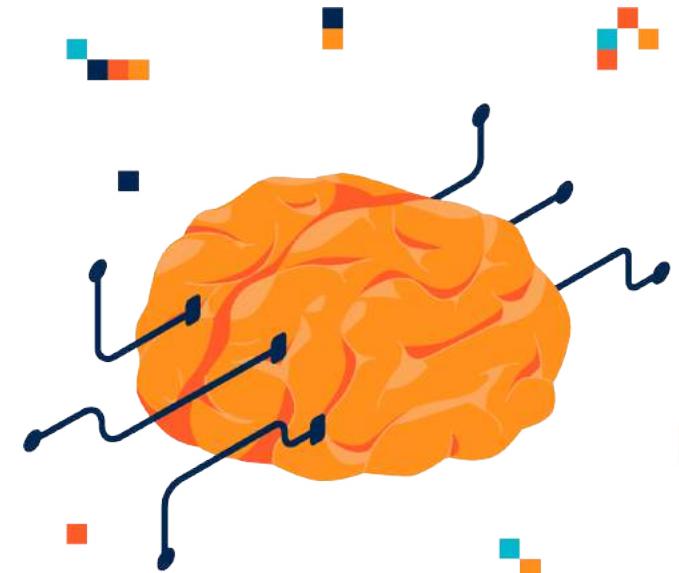
K NEAREST NEIGHBORS (KNN): HYPERPARAMETERS

- Full set of hyperparameters:
https://docs.aws.amazon.com/sagemaker/latest/dg/kNN_hyperparameters.html
- K: The number of nearest neighbors
- Sample_size: The number of data points to be sampled from the training data set.
- feature_dim: The number of features in the input data.
- predictor_type: The type of inference to use on the data labels.
- dimension_reduction_target: The target dimension to reduce to.



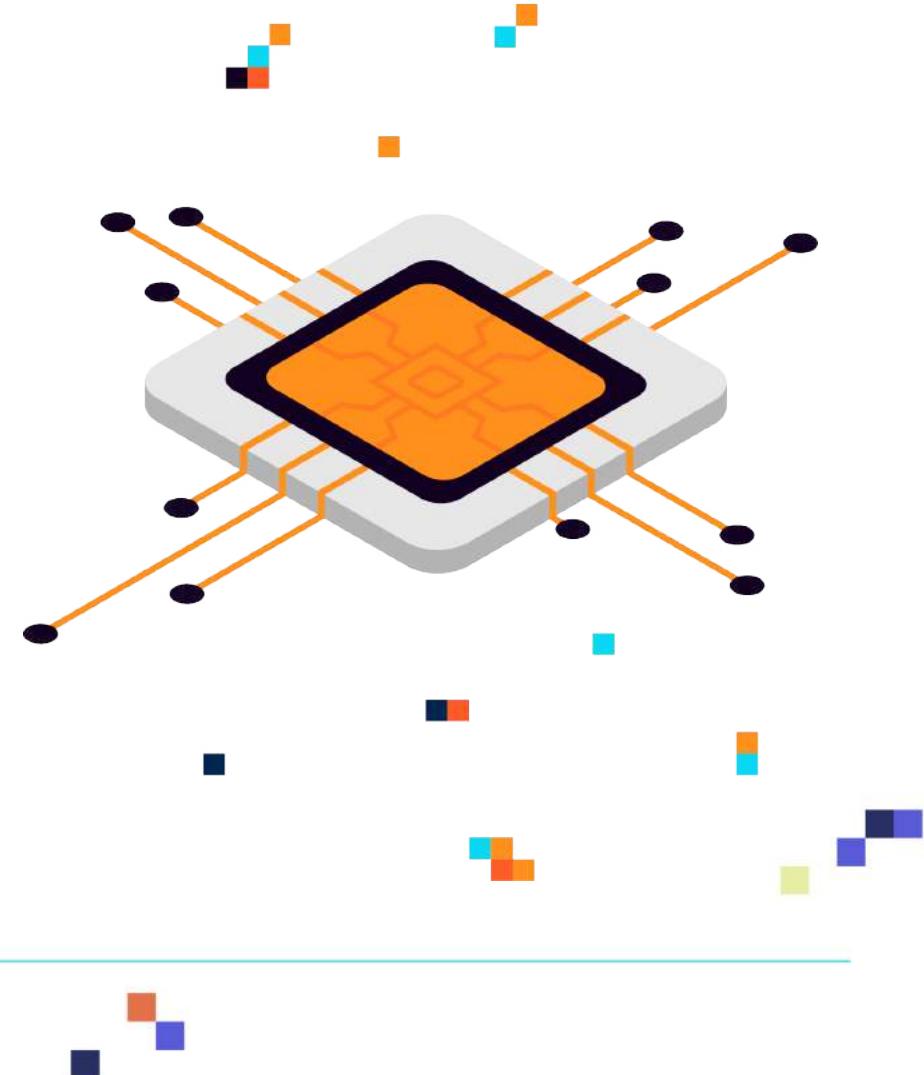
K NEAREST NEIGHBORS (KNN): INPUT/OUTPUT

- KNN supports two channels:
 - Train channel contains training data
 - Test channel to provide test scores such as accuracy for classifier or MSE for regressor
- SageMaker KNN algorithm supports recordIO-protobuf or CSV formats
- KNN can be used in both File or pipe mode



K NEAREST NEIGHBORS (KNN): INSTANCE TYPES

- For KNN Training:
 - CPU such as Ml.m5.2xlarge
 - GPU such as Ml.p2.xlarge
- For Inference:
 - GPU for higher throughput on large batches
 - CPU generally provides lower latency

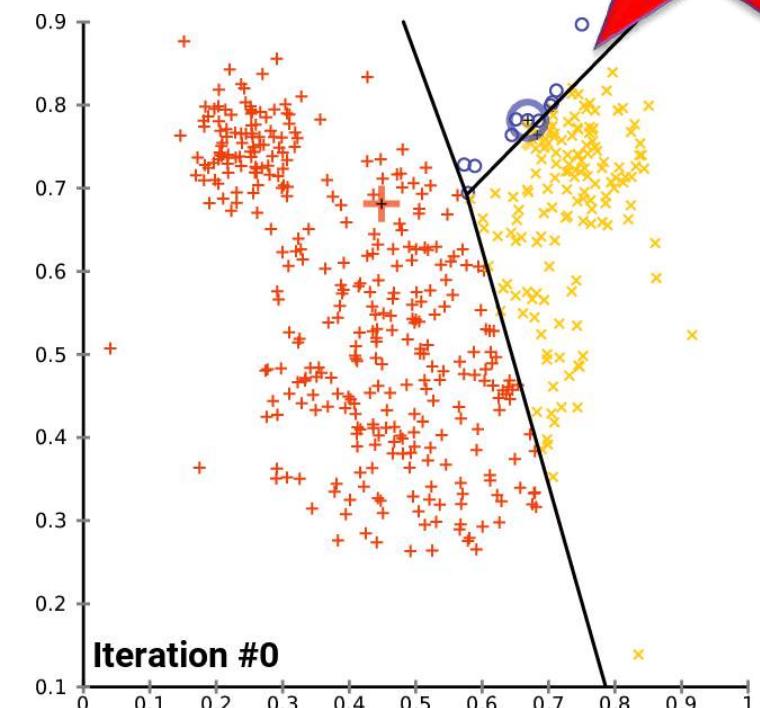


K MEANS



K MEANS: OVERVIEW

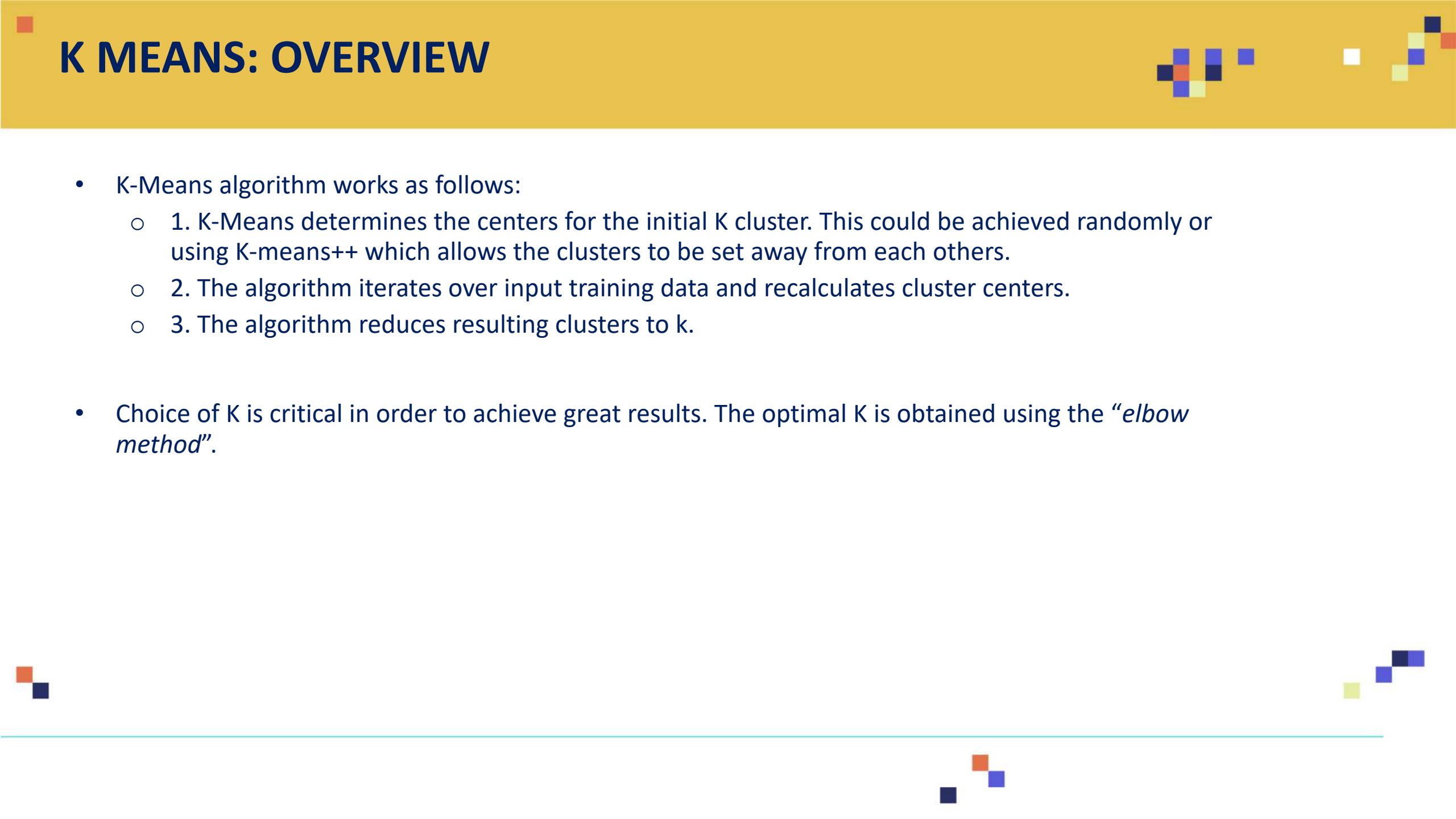
- K-means is an unsupervised learning algorithm (clustering).
- K-means works by grouping some data points together (clustering) in an unsupervised fashion.
- Amazon SageMaker K-means algorithm scales for large amount of data and offers great computational efficiency.
- Amazon SageMaker offers these advantages by streaming mini-batches (small, random subsets) of the training data.
- The k-means algorithm relies on tabular data:
 - observations are put in rows
 - Attributes of the observations are put in columns. So in every row, these attributes demonstrates a single point in n -dimensional space.
- The algorithm groups observations with similar attribute values together by measuring the Euclidian distance between points.



K MEANS: OVERVIEW



- K-Means algorithm works as follows:
 - 1. K-Means determines the centers for the initial K cluster. This could be achieved randomly or using K-means++ which allows the clusters to be set away from each others.
 - 2. The algorithm iterates over input training data and recalculates cluster centers.
 - 3. The algorithm reduces resulting clusters to k.
- Choice of K is critical in order to achieve great results. The optimal K is obtained using the “*elbow method*”.



K MEANS: HYPERPARAMETERS

- Full set of hyperparameters:
<https://docs.aws.amazon.com/sagemaker/latest/dg/k-means-api-config.html>
- feature_dim: The number of features in the input data.
- K: The number of clusters
- init_method: hyperparameter that determines how the K-mean algorithm will choose the initial cluster centers.
- eval_metrics: metrics to assess the performance of the model
- extra_center_factor



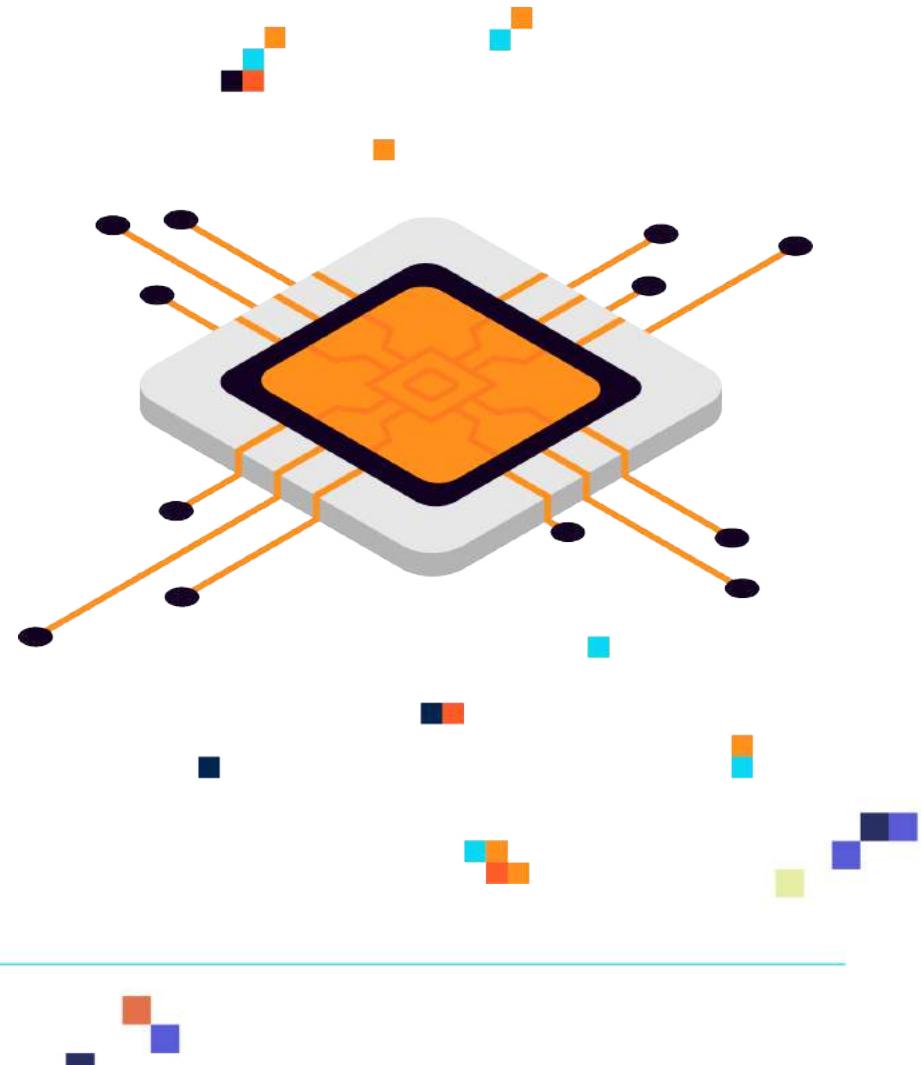
K MEANS: INPUT/OUTPUT

- K-Means supports two channels:
 - Train channel contains training data
 - Optional Test channel
- SageMaker K-Means algorithm supports recordIO-protobuf or CSV formats
- K-Means can be used in both File or pipe mode



K MEANS: EC2 INSTANCE

- For K-Means Training:
 - CPU instance is recommended
 - Technique could work on GPU as well but only one GPU per instance used on GPU
 - op*.xlarge recommended in case of using GPU



PRINCIPAL COMPONENT ANALYSIS (PCA)



PRINCIPAL COMPONENT ANALYSIS: OVERVIEW

- PCA is an unsupervised machine learning algorithm.
- PCA performs dimensionality reductions while attempting at keeping the original information unchanged.
- PCA works by trying to find a new set of features called components.
- Components are composites of the uncorrelated given input features.
- The first component accounts for largest data variability followed by second component and so on.

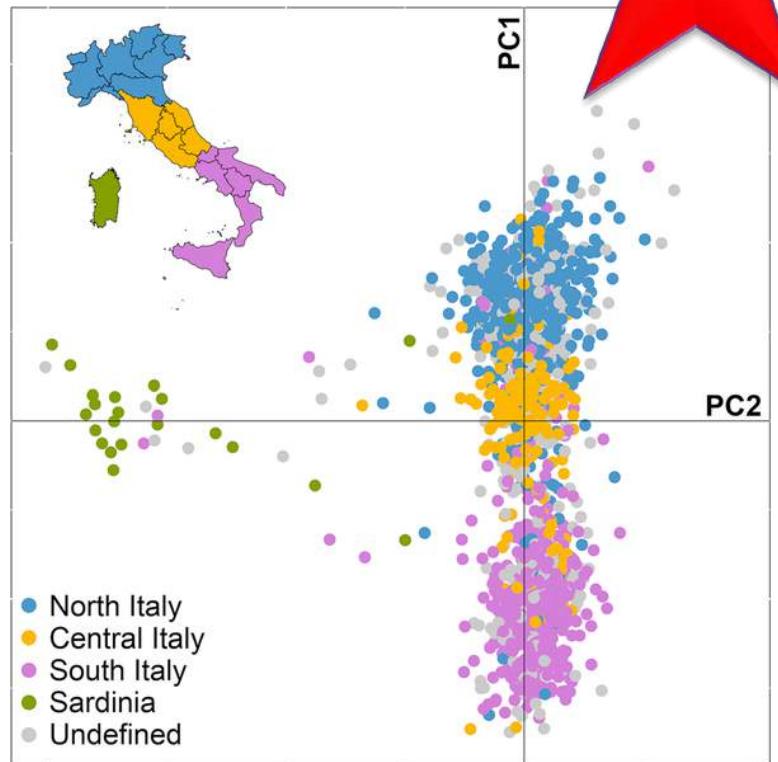
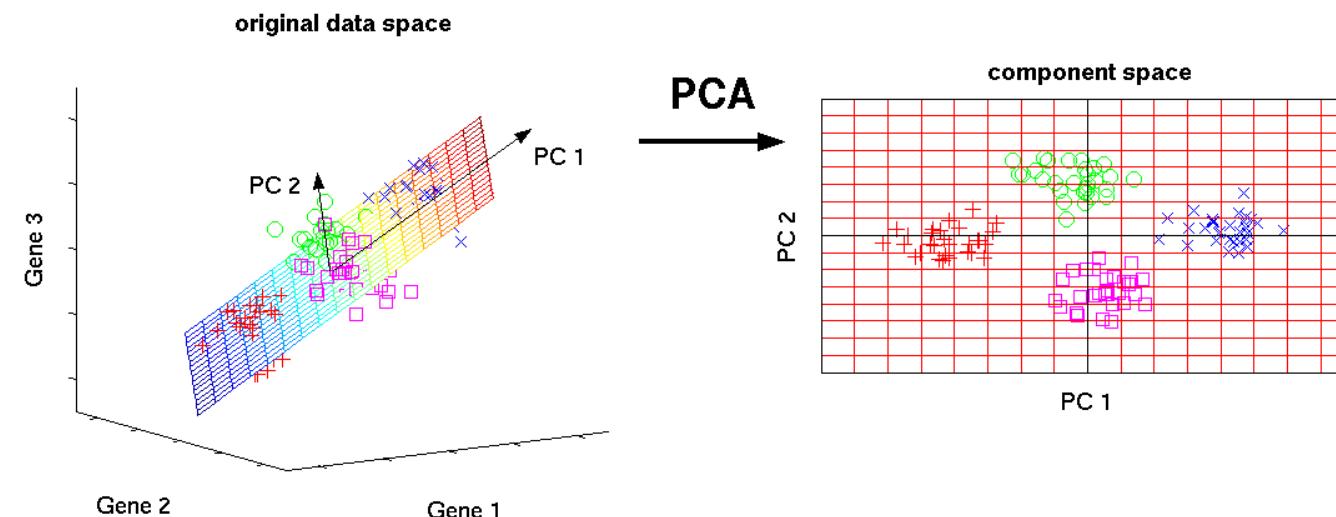


Photo Credit: https://commons.wikimedia.org/wiki/File:Principal_Component_Analysis_of_the_Italian_population.png

PRINCIPAL COMPONENT ANALYSIS: OVERVIEW

- PCA (Principal Component Analysis) performs dimensionality reduction as shown below.
- The algorithm generates the principal components by calculating the covariance matrix first and then doing singular value decomposition.
- In Amazon SageMaker PCA operates in two modes:
 - Regular: works well with sparse data small (manageable) number of observations/features.
 - Randomized: works well with large number of observations/features.



PRINCIPAL COMPONENT ANALYSIS: HYPERPARAMETERS

- Full set of hyperparameters:
<https://docs.aws.amazon.com/sagemaker/latest/dg/PCA-reference.html>
- feature_dim: number of features in the input data.
- num_components: number of principal components to compute.
- algorithm_mode: Mode for computing the principal components, choose between regular or randomized
- extra_components: As extra components go up, more accurate results are achieved at the cost of increased memory/computation consumption.
- subtract_mean: True or false Boolean to dictate whether data should be unbiased or not.



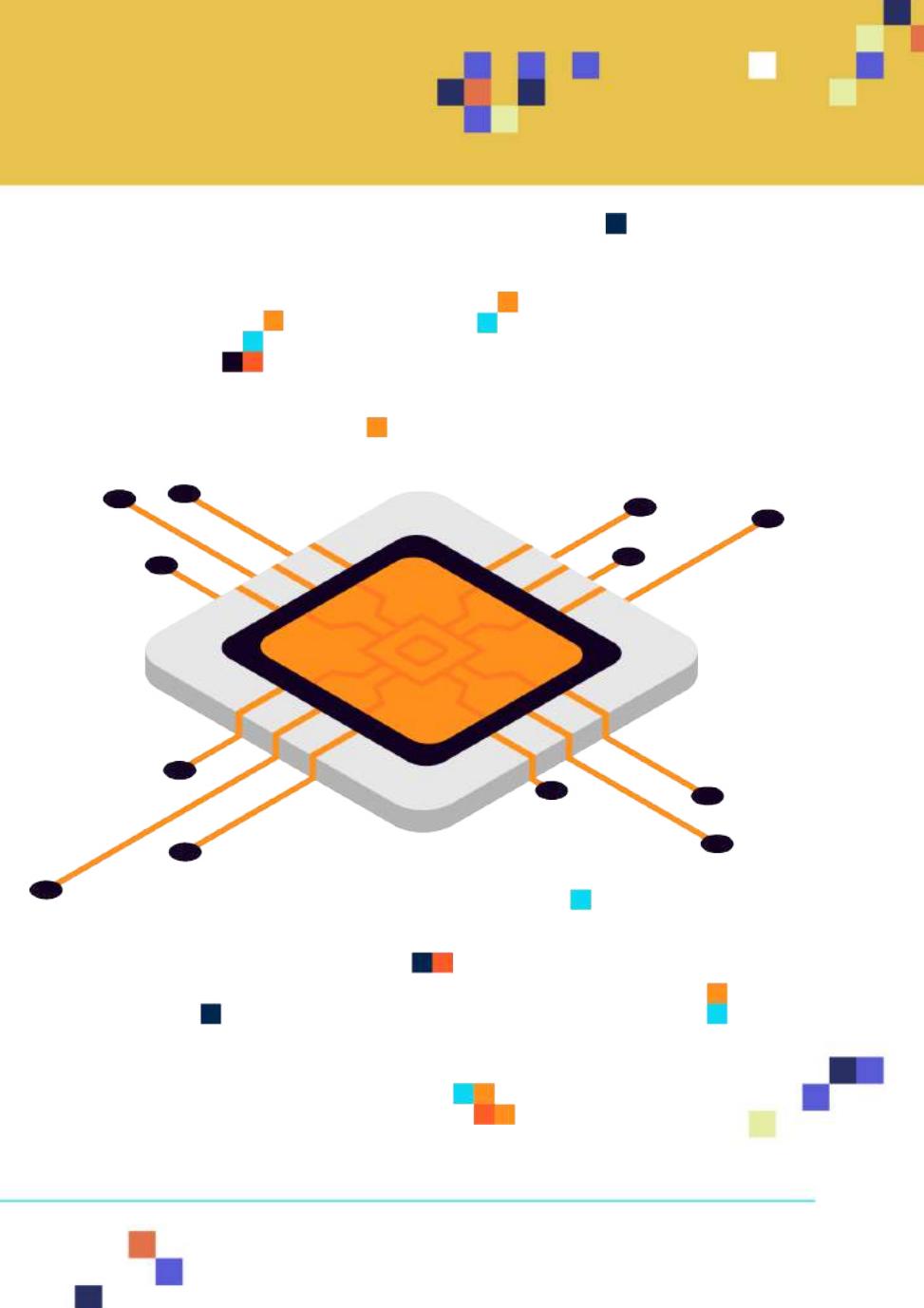
PRINCIPAL COMPONENT ANALYSIS: INPUT/OUTPUT

- SageMaker PCA algorithm supports recordIO-protobuf or CSV formats
- PCA can be used in both File or pipe mode



■ PRINCIPAL COMPONENT ANALYSIS: INSTANCE TYPES

- For PCA Training:
 - CPU instance or GPU are recommended



IP INSIGHTS



IP INSIGHTS: OVERVIEW

- Amazon SageMaker IP Insights is an unsupervised learning algorithm.
- The algorithm works by learning the usage patterns for various IPv4 addresses.
- The algorithm learns the relationship between various entities and IPv4 addresses.
- IP insights is useful for fraud detection to detect anomalous logins from foreign (unusual) IPv4 addresses.
- Trained IP Insight models can be hosted at an endpoint and works in real-time.

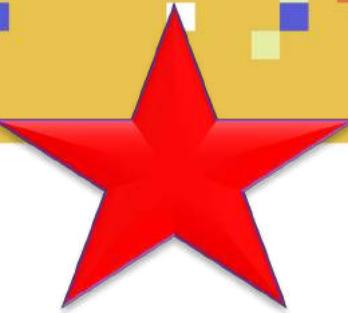
IPv4 address in dotted-decimal notation

172 . 16 . 254 . 1



IP INSIGHTS: OVERVIEW

- When you send an (entity, IPv4 Address) event to a trained IP insights model, it provides a score that indicates if the event is ‘anomalous’ or ‘safe’.
- If the event is anomalous, a second factor authentication might be triggered to verify user identity.
- More advanced security systems could be developed by combining IP insight’s generated scores with other features to rank the findings of another security system (Amazon GuardDuty).
- IP Insights is capable of learning embeddings which consists of vector representations of IP addresses.
- Those embeddings can be used as features and being fed into another machine learning model.
- Under the hood, IP insights uses a neural network to learn latent vector representations of both IPv4 address and entities.
- IP insights work by hashing and embedding Entities
- In order to increase datasets during training, the algorithm generates negative samples by randomly pairing entities and IP’s



IP INSIGHTS: HYPERPARAMETERS

- Full set of hyperparameters:
[https://docs.aws.amazon.com/sagemaker/latest/dg
/ip-insights-hyperparameters.html](https://docs.aws.amazon.com/sagemaker/latest/dg/ip-insights-hyperparameters.html)
- num_entity_vectors: indicates the number of entity vector representations to train.
- vector_dim: size of embedding vectors to represent entities and IP addresses.
- learning_rate
- Epochs
- num_ip_encoder_layers: The number of fully connected layers used to encode the IP address embedding.



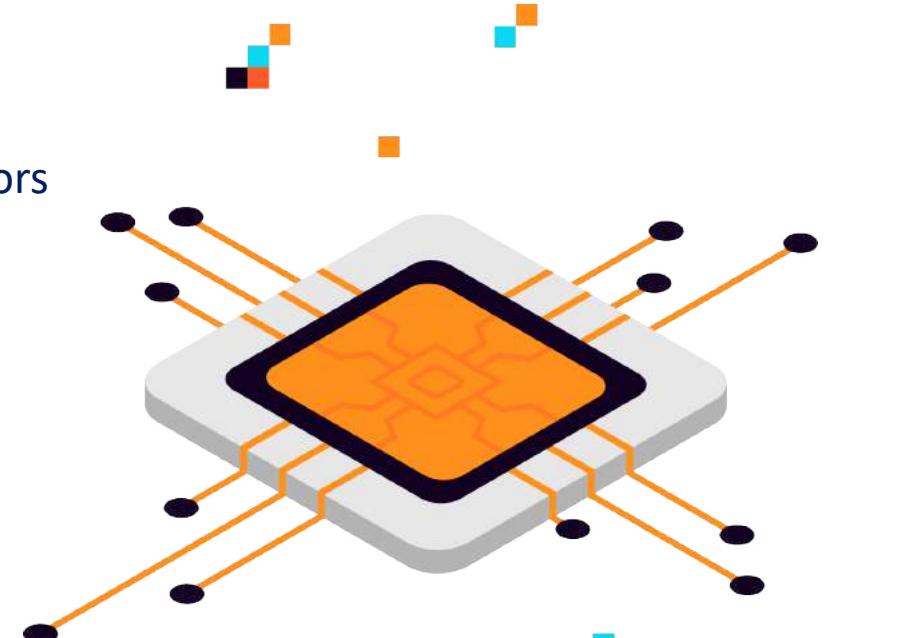
IP INSIGHTS : INPUT/OUTPUT

- IP Insights expects two channels: (1) training and (2) validation (optional).
- Training and validation channels must be in text/csv format.
 - CSV File Column #1: holds a unique identifier for the entity.
 - CSV File Column #2: holds the IPv4 address.
- The validation channel is used to assess the performance of the algorithm and see how good is it in discriminating between positive and negative samples.
- An important metric for performance assessment is the area-under-the-curve (AUC) score on a predefined negative sampling strategy.
- IP Insights currently supports only File mode.
- For Inference, IP Insights supports text/csv, application/json, and application/jsonlines data content types.

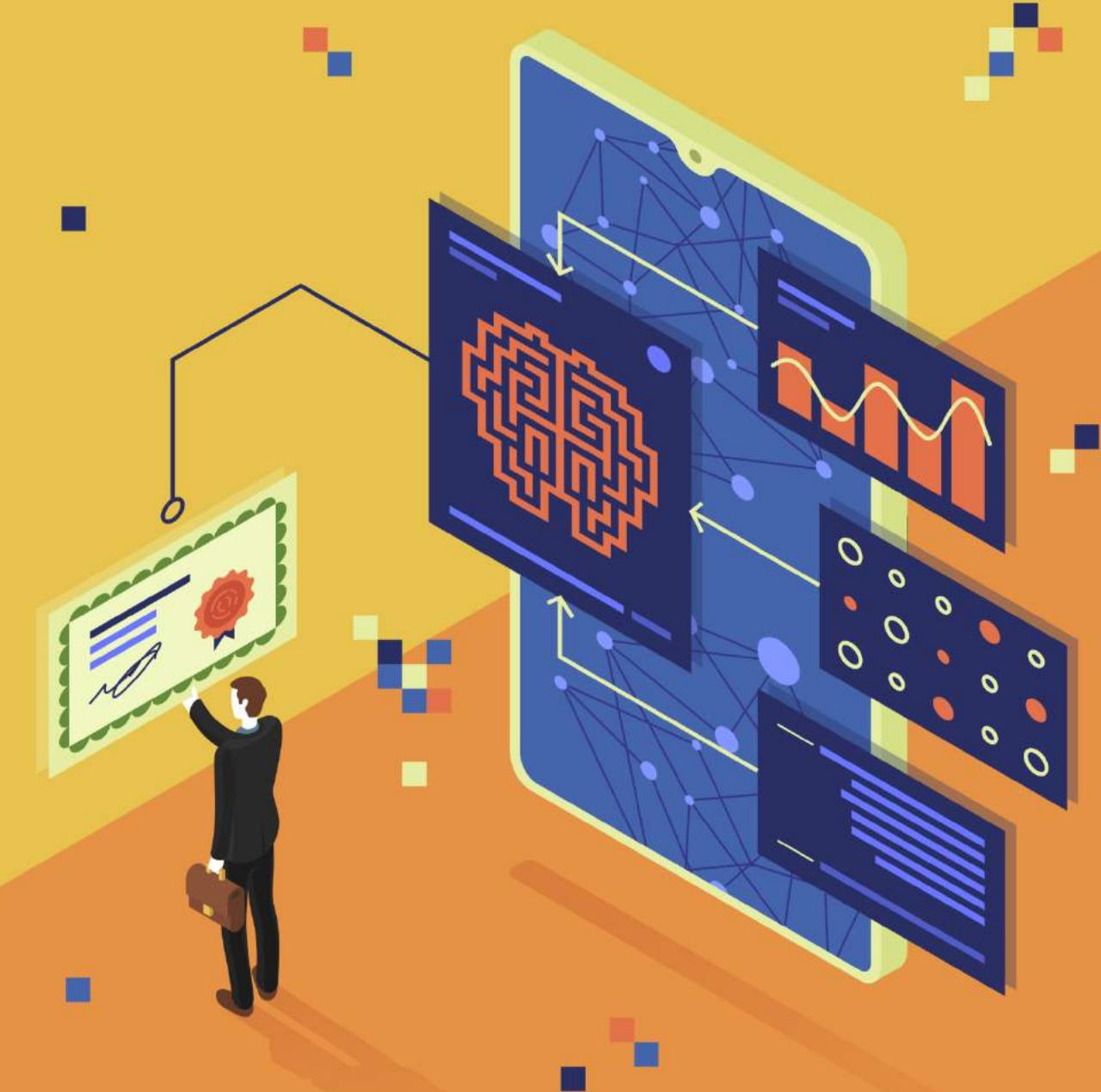


IP INSIGHTS: INSTANCE TYPES

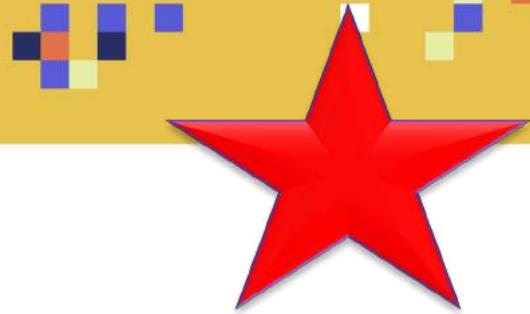
- In order to train the IP insights, a GPU instance is recommended
 - MI.p3.2xlarge or higher
 - Size of CPU instance depends on vector_dim and num_entity_vectors
 - IP insights can rely on multiple GPUs



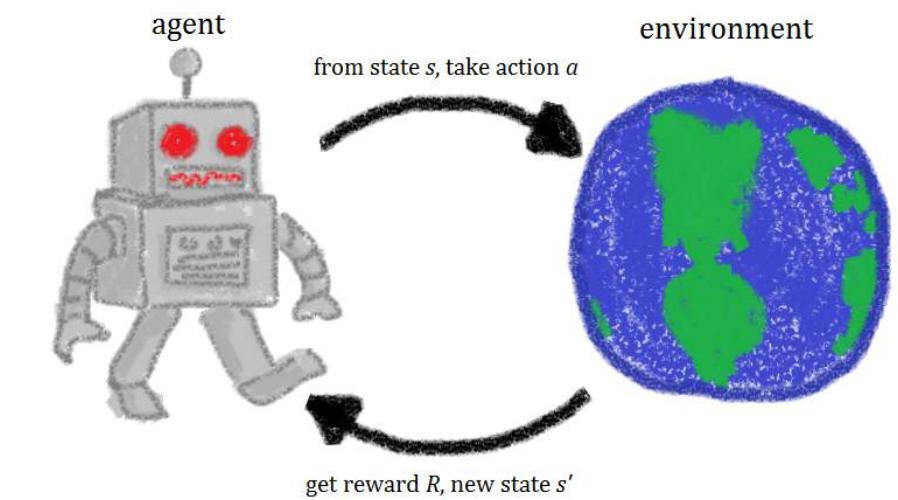
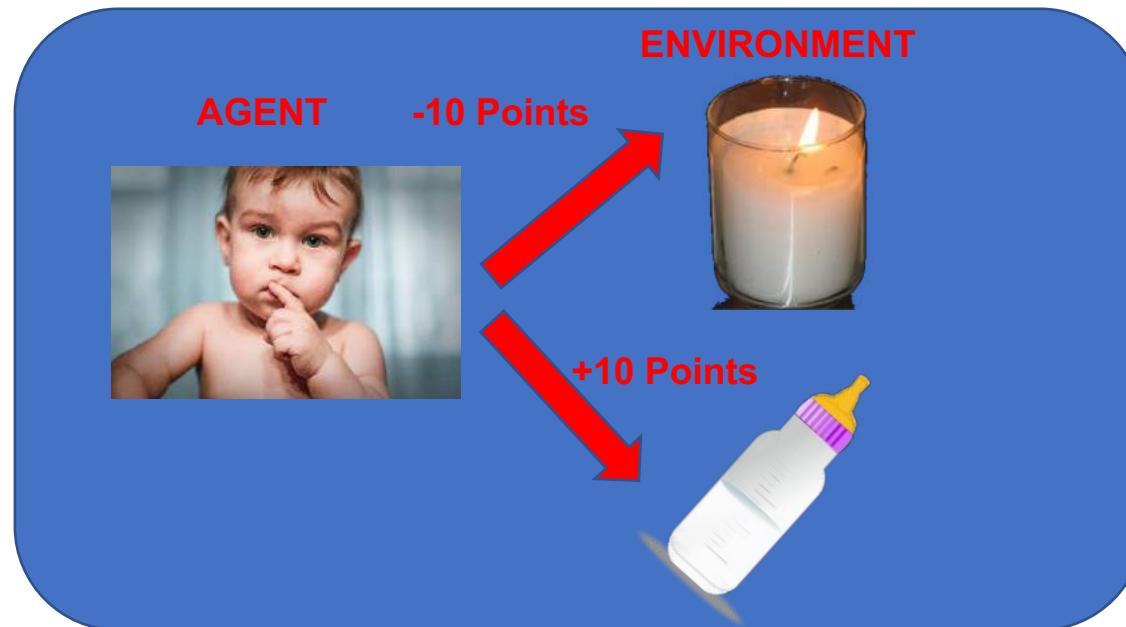
REINFORCEMENT LEARNING



MACHINE LEARNING: REINFORCEMENT LEARNING



- Reinforcement learning allows machines take actions to maximize cumulative reward.
- Reinforcement algorithms learn by trial and error through reward and penalty.
- Two elements: **environment** and **learning agent**.
- The environment rewards the agent for correct actions.
- Based on the reward or penalty, agent improves its environment knowledge to make better decision.



https://commons.wikimedia.org/wiki/File:RL_agent.png

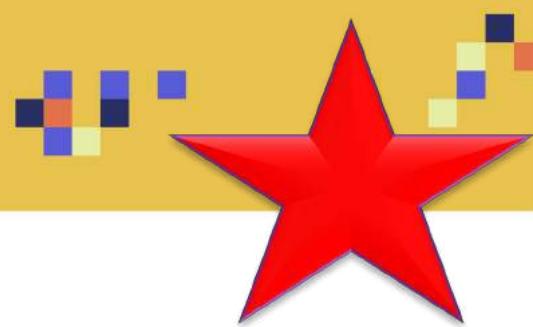
REINFORCEMENT LEARNING: OVERVIEW

- Reinforcement learning (RL) is a machine learning technique that works by learning a policy or strategy through trial and error.
- In RL, an agent is present in an environment and its objective is to maximize cumulative reward.
- In RL, the following steps are performed by the agent:
 - Takes an action
 - Observes the state of the environment
 - Receives a reward or penalty
- The goal is to maximize the cumulative long-term reward.
- Example:
 - Agent: Robot
 - Environment: maze
 - Objective: Navigate the maze in the shortest time possible
- After the robot is trained, it can make decisions on its own

REINFORCEMENT LEARNING: WHY IS REINFORCEMENT LEARNING IMPORTANT?

- RL works well with complex large scale tasks such as self driving cars, supply chain management, game artificial intelligence, industrial robotics.
- Since RL models learn by trying to maximize the reward and reduce the penalty, it can be trained to make decisions in dynamic environments and under extreme uncertainty.

REINFORCEMENT LEARNING : MARKOV DECISION PROCESS (MDP)

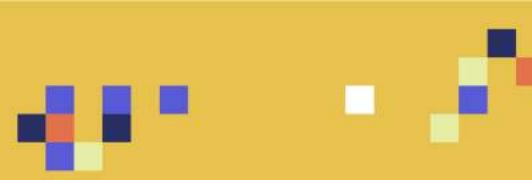


- Reinforcement learning implements Markov Decision Processes (MDPs) model.
- RL training consists of a series of time steps in an MDP starting from an initial state until final state.
- MDP works through a series of time steps that include the following:
 - **Environment:** includes the space in which the RL model operates which could be simulation or real world environments.
 - **State:** indicates the data related to the past actions that the RL model has taken. The state is important in determining the actions that the RL model will take in the future. For example: in case of autonomous cars, the RL model state is the position of the vehicle.
 - **Action:** action taken by the agent such as autonomous vehicle making a right turn.
 - **Reward:** the number that the agent takes after taking the last action. Remember that the agent is trying to maximize the cumulative reward. The RL Model is trying to find the optimal policy or strategy to maximize the reward and avoid penalties.
 - **Observation:** data related to the environment state. It could be visible or partially visible to the agent.
 - ❖ Example #1 chess: full state of the board is visible to agent.
 - ❖ Example #2: robot in a maze can only view a small portion of the maze.

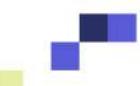
REINFORCEMENT LEARNING: KEY FEATURES OF AMAZON SAGEMAKER RL

- The following components are available in RL SageMaker:
 - TensorFlow and Apache MXNet deep learning framework frameworks.
 - An RL toolkit:
 - ❖ Includes state of the art RL algorithms and allows for the proper communication between agent and environment.
 - ❖ Amazon SageMaker supports the Intel Coach and Ray RLLib toolkits.
 - RL environment: open source, custom or commercial environments are available.

REINFORCEMENT LEARNING: RL ENVIRONMENTS

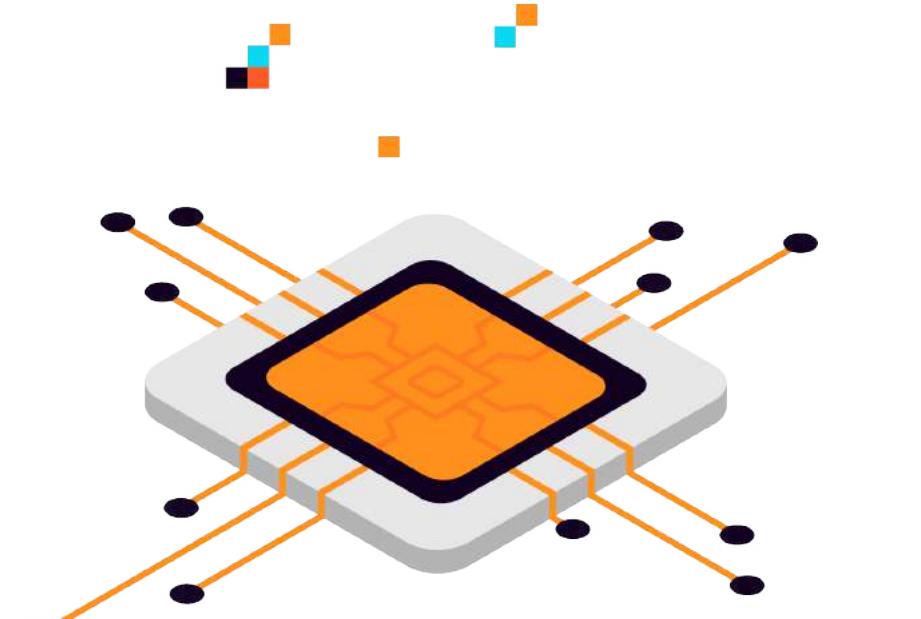


- Amazon SageMaker RL offers environments that emulates real world.
- Simulators are crucial in some cases where driving (self driving cars) or flying (drones) in real life is dangerous
- The simulation environment consists of an agent and a simulator.
- Here are a list of options available:
 - Use OpenAI Gym Interface for Environments in Amazon SageMaker RL:
 - ❖ Gym is a toolkit for developing and comparing reinforcement learning algorithms. Check out the website here: <https://gym.openai.com/docs/>
 - Use Open Source Environments: EnergyPlus and RoboSchool, in Amazon SageMaker RL by building your own container.
 - Use Commercial Environments: MATLAB and Simulink.
- Check this out for the full diagram of RL components supported by SageMaker:
<https://docs.aws.amazon.com/sagemaker/latest/dg/reinforcement-learning.html>



REINFORCEMENT LEARNING: EC2 INSTANCES

- GPUs are recommended
- You can run RL on:
 - Multi-core and multi-instance
 - Can distribute training and/or environment



REINFORCEMENT LEARNING: HYPERPARAMETERS

- Hyperparameter tuning job is required in order to optimize hyperparameters for Amazon SageMaker RL.
- Check out this example for abstracting parameters to tune:
<https://github.com/awslabs/amazon-sagemaker-examples/tree/master/reinforcement-learning>



NEURAL TOPIC MODEL



NEURAL TOPIC MODEL (NTM): OVERVIEW

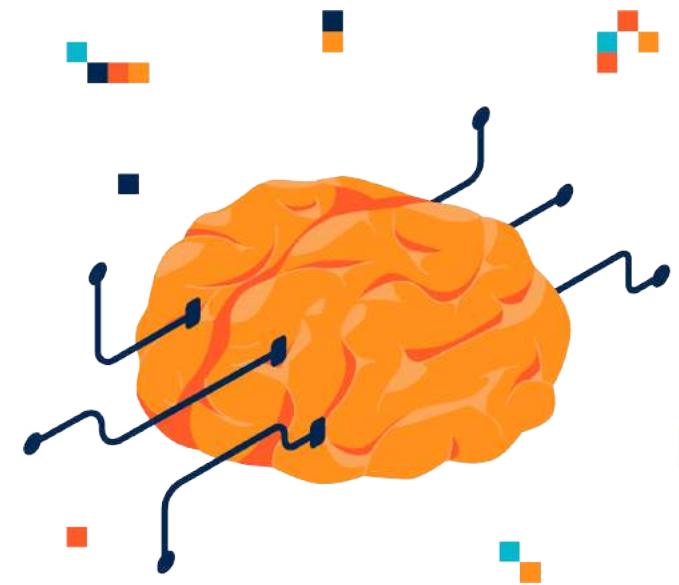
- Neural Topic Model is an unsupervised learning algorithm
- NTM organizes a corpus of documents into topics.
- For example, the words hungry, bananas, chocolate, watermelons share the same topic of “food”
- Topic modeling can be used to classify/summarize documents based on the topics detected.
- Amazon SageMaker offers two algorithms (1) NTM and (2) LDA (to be presented shortly) to perform topic modeling. They could generate totally different results.
- Users of the algorithm need to specify how many topics they'd like to have. Topics are a latent representation based on top ranking words.



Photo Credit: <http://www.peakpx.com/465710/vinyl-records-and-file-documents>

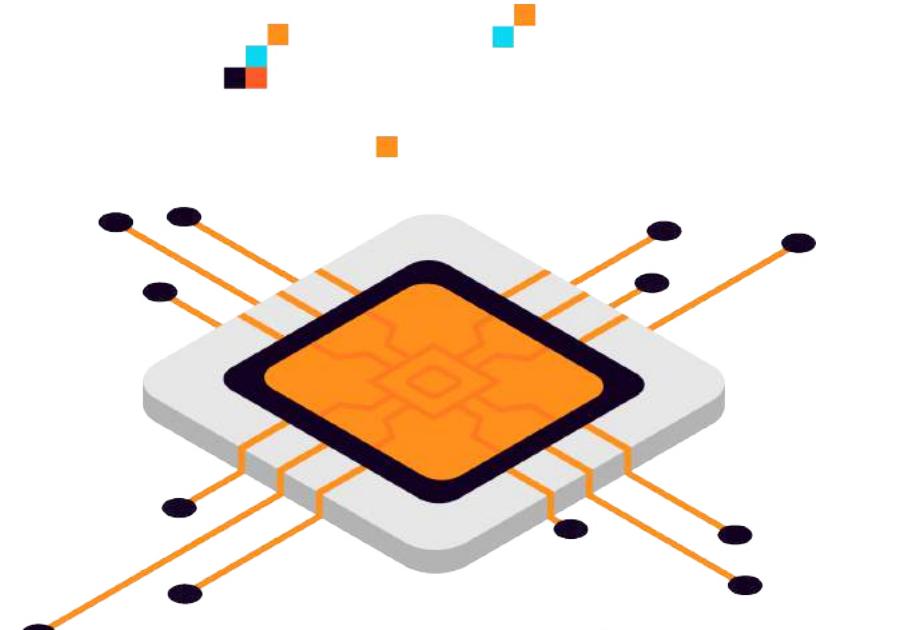
NEURAL TOPIC MODEL: INPUT/OUTPUT

- Amazon SageMaker Neural Topic Model supports four data channels as follows:
 - Train
 - Validation (optional)
 - Test (optional)
 - Auxiliary (optional)
- Supports the following formats:
 - recordIO-wrapped-protobuf (dense and sparse)
 - CSV file formats.
- File mode or Pipe mode could be used to train models on data in recordIO-wrapped-protobuf or CSV formats.



NEURAL TOPIC MODEL: EC2 INSTANCE

- NTM training supports both GPU and CPU instances.
 - GPU instances are recommended
 - For inference, CPU instances are sufficient.

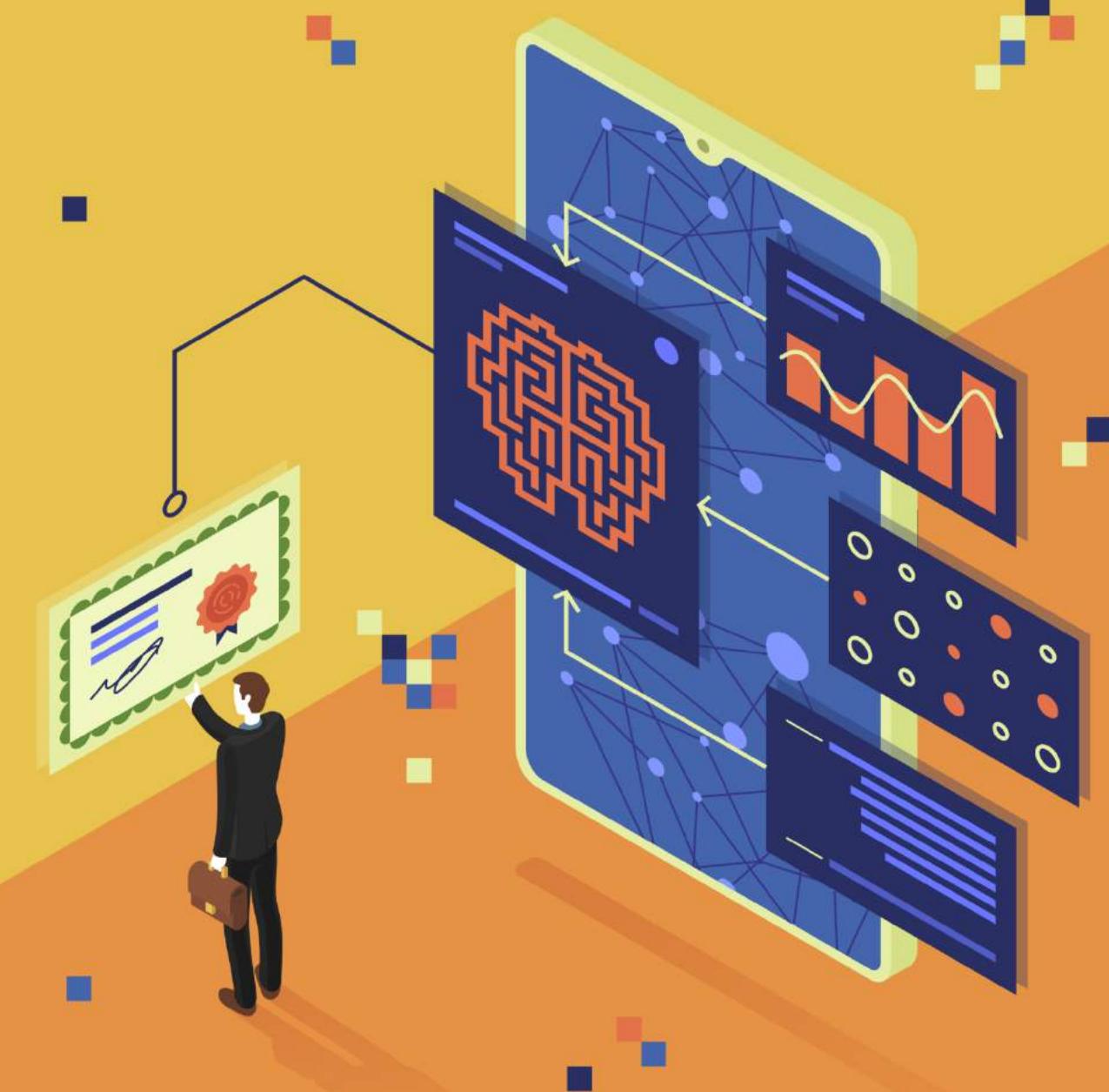


NEURAL TOPIC MODEL: HYPERPARAMETERS

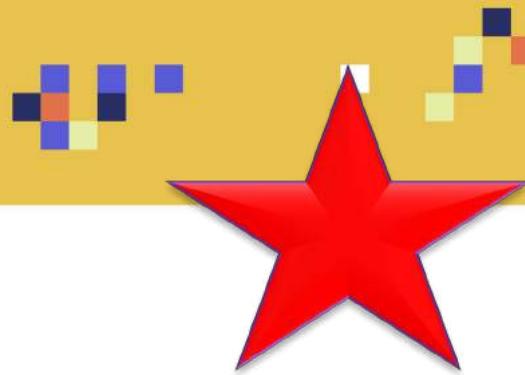
- **For full list of parameters:**
https://docs.aws.amazon.com/sagemaker/latest/dg/ntm_hyperparameters.html
- **feature_dim:** vocabulary size of the dataset
- **num_topics:** number of required topics
- **encoder_layers:** number of layers in the encoder and the output size of each layer
- **encoder_layers_activation:** activation function to use in the encoder layers
- **sub_sample:** fraction of training data to sample for training per epoch
- **Epochs**
- **learning_rate**
- **Optimizer**



LDA



LDA: OVERVIEW



- The Amazon SageMaker LDA stands for Latent Dirichlet Allocation.
- The algorithm is unsupervised learning algorithm.
- LDA describes a set of observations as a mixture of distinct categories.
- LDA performs the same function as the Neural Topic algorithm but does not rely on deep learning so it's not computationally expensive (CPU instance would be enough).
- LDA takes in a text corpus and group them into user-specified number of topics



Photo Credit: <http://www.peakpx.com/465710/vinyl-records-and-file-documents>



LDA: OVERVIEW



- LDA is an unsupervised algorithm so topics won't necessarily align with what humans could categorize them.
- Example: Let's assume that we have a set of documents and the only words that occur within them are: *eat, sleep, play, meow, and bark*
- LDA will generate the following table:

Topic	eat	sleep	play	meow	bark
Topic 1	0.1	0.3	0.2	0.4	0.0
Topic 2	0.2	0.1	0.4	0.0	0.3

- Topic 1 is probably about cats who meow and sleep
- Topic 2 is probably about dogs play and bark
- These topics can be found even though the words dog and cat never appear in any of the texts.



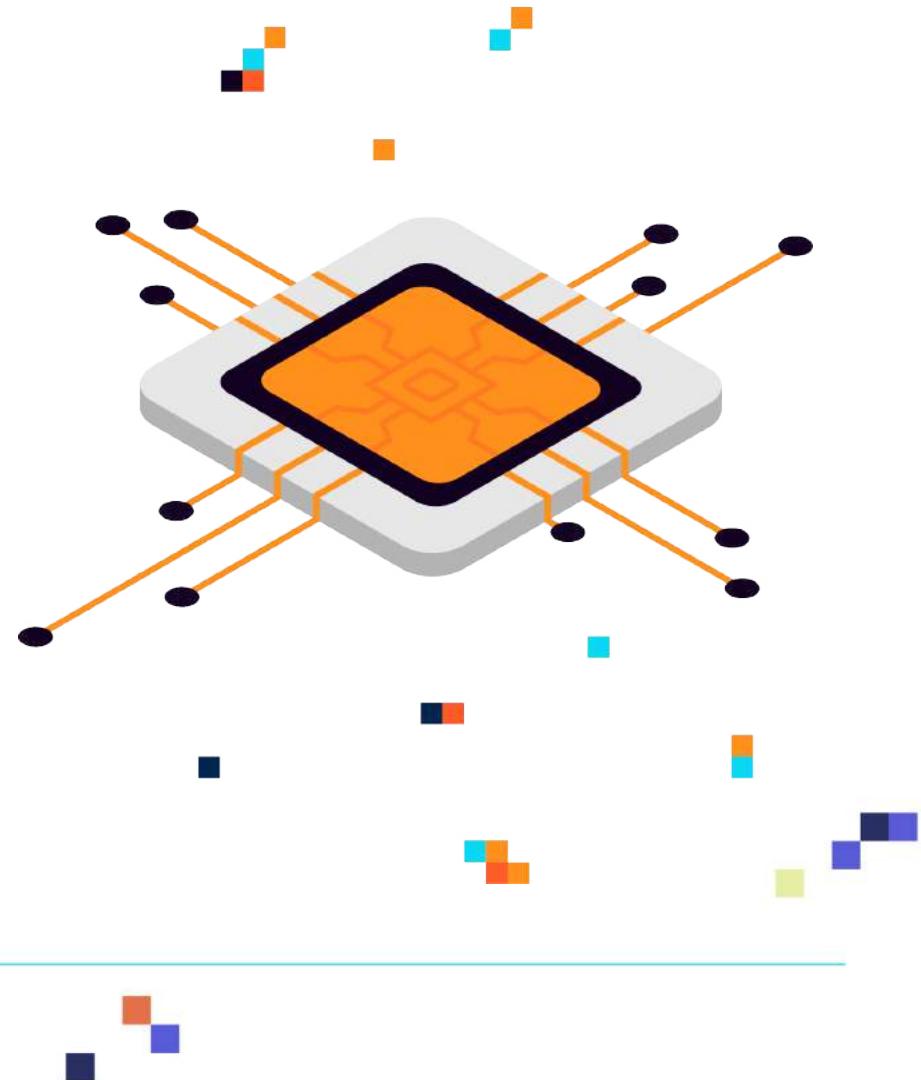
LDA: INPUT/OUTPUT

- Amazon SageMaker LDA supports:
 - Train channel
 - Test channel (optional)
- Supports the following formats:
 - recordIO-protobuf
 - CSV file formats.
- Pipe mode could be used to train models on data in recordIO format.



LDA: EC2 INSTANCE

- One CPU instances is sufficient for training and inference.



LDA: HYPERPARAMETERS

- **For full list of parameters:**
https://docs.aws.amazon.com/sagemaker/latest/dg/lda_hyperparameters.html
- **feature_dim:** vocabulary size of the dataset
- **num_topics:** number of required topics
- **Alpha0:** Initial guess for the concentration parameter, smaller value generates sparse topic mixtures while Large values (>1.0) produce more uniform mixtures
- **Max_iterations**
- **Max_restarts**



OBJECT2VEC



OBJECT2VEC: OVERVIEW

- SageMaker Object2Vec is an unsupervised algorithm that generalizes the Word2Vec embedding technique introduced before (blazingtext algorithm).
- It is a general-purpose customizable neural embedding algorithm which can be used to embed sentence, movies and products.
- Object2vec can learn low-dimensional dense embeddings of high-dimensional objects.
- The embeddings keeps the relationship between pairs of objects.
- Object2vec can be used to: (1) compute nearest neighbors of objects, (2) visualize clusters of related objects.
- Using these embedding, you can feed this to perform classification or regression.
- Example: you can apply object2vec to learn the embeddings of customers. This can be achieved by creating training data from a sequence of transactions/purchases paired with the ID of the customer.

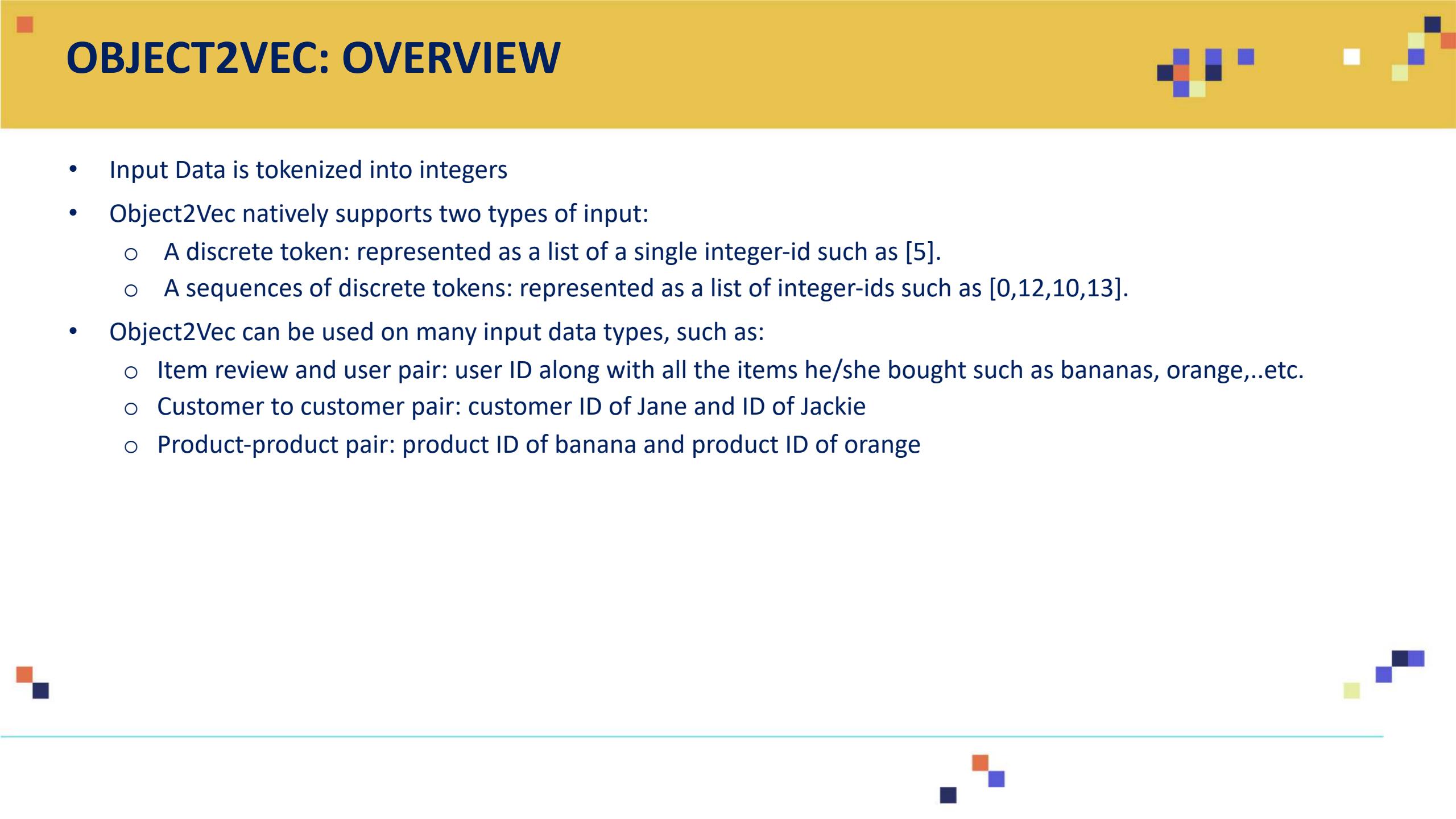


Photo Credit: <https://pixabay.com/photos/groceries-fruit-vegan-soy-food-1343141/>

OBJECT2VEC: OVERVIEW

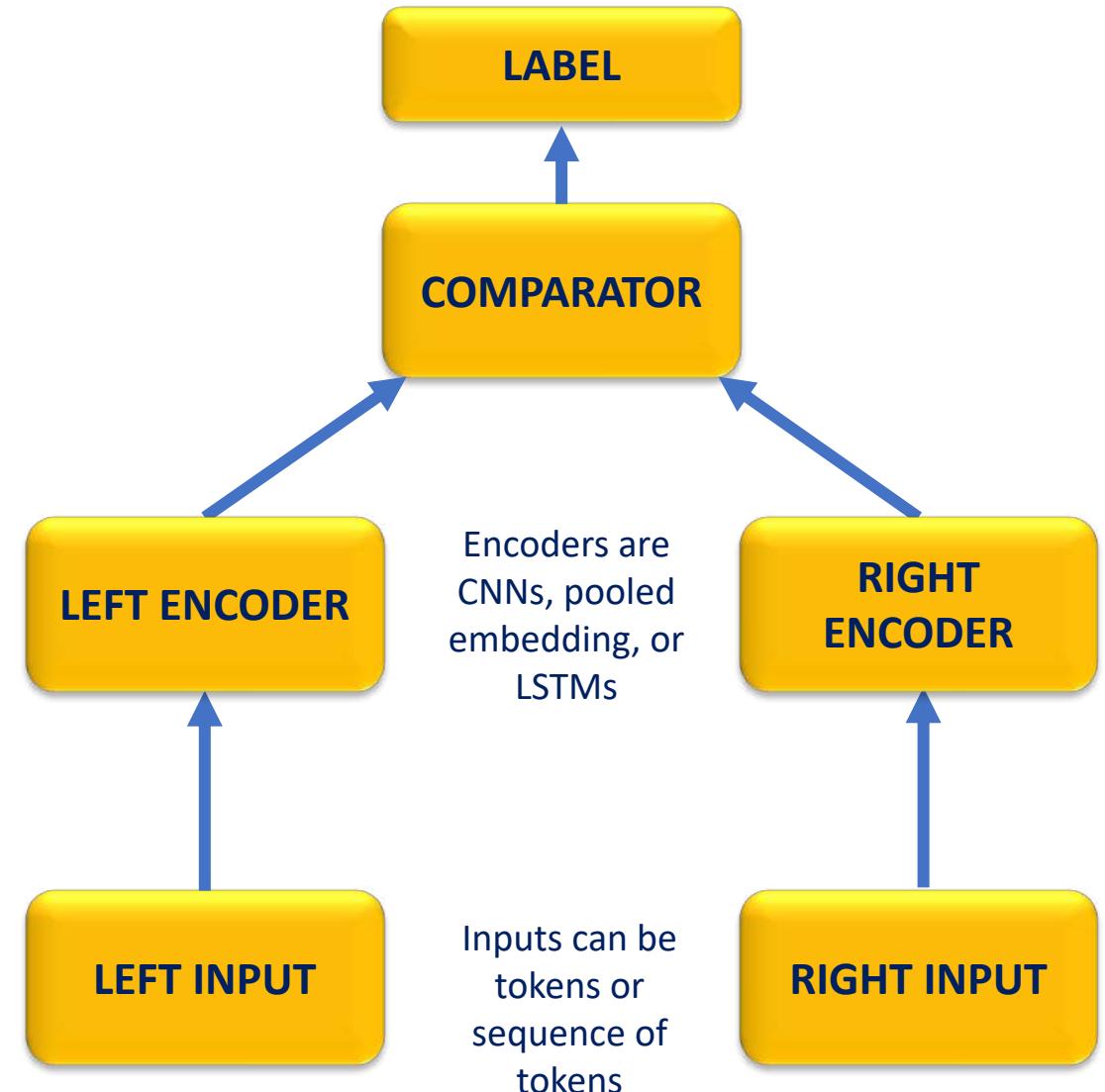


- Input Data is tokenized into integers
- Object2Vec natively supports two types of input:
 - A discrete token: represented as a list of a single integer-id such as [5].
 - A sequences of discrete tokens: represented as a list of integer-ids such as [0,12,10,13].
- Object2Vec can be used on many input data types, such as:
 - Item review and user pair: user ID along with all the items he/she bought such as bananas, orange,..etc.
 - Customer to customer pair: customer ID of Jane and ID of Jackie
 - Product-product pair: product ID of banana and product ID of orange



OBJECT2VEC: OVERVIEW

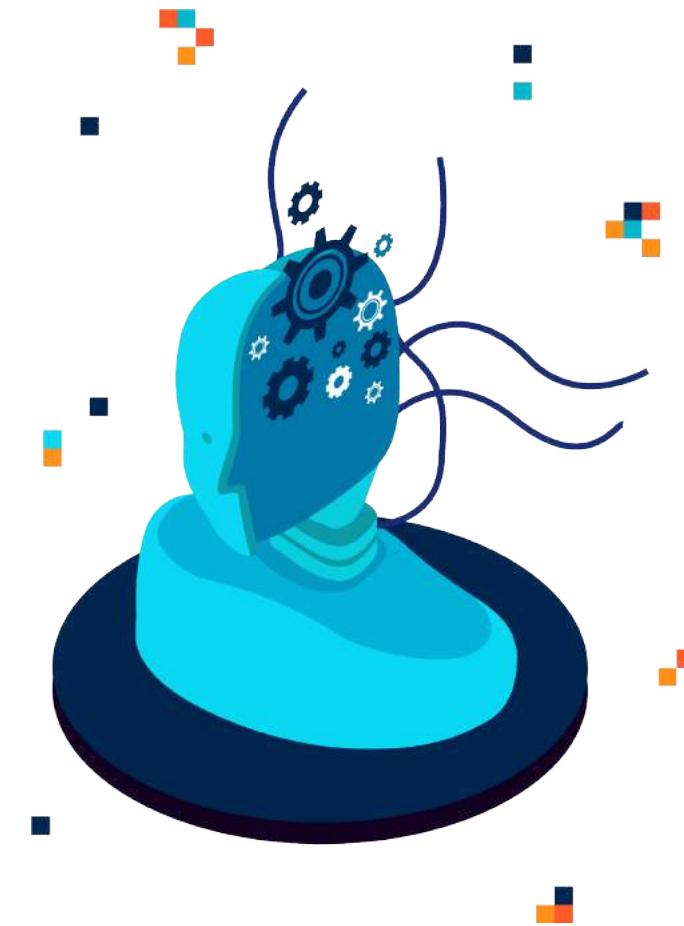
- For object2vec, pairs can as follows:
 - (token, sequence)
 - (token, token)
 - (sequence, sequence)
- Choose the proper encoder:
 - Average-pooled embeddings
 - convolutional neural networks (CNNs)
 - Multi-layered bidirectional long short-term memory (BiLSTMs)



<https://aws.amazon.com/blogs/machine-learning/introduction-to-amazon-sagemaker-object2vec/>

OBJECT2VEC: HYPERPARAMETERS

- For the full list of parameters:
<https://docs.aws.amazon.com/sagemaker/latest/dg/object2vec-hyperparameters.html>
 - enc0_max_seq_len: The maximum sequence length for the enc0 encoder.
 - enc0_vocab_size: The vocabulary size of enc0 tokens.
 - comparator_list: A list used to customize the way in which two embeddings are compared.
 - Dropout
 - early_stopping_patience: The number of consecutive epochs without improvement allowed before early stopping is applied.
 - enc_dim: The dimension of the output of the embedding layer.
 - enc0_cnn_filter_width: The filter width of the convolutional neural network (CNN) enc0 encoder.
 - enc0_token_embedding_dim: The output dimension of the enc0 token embedding layer.



OBJECT2VEC: EC2 INSTANCES

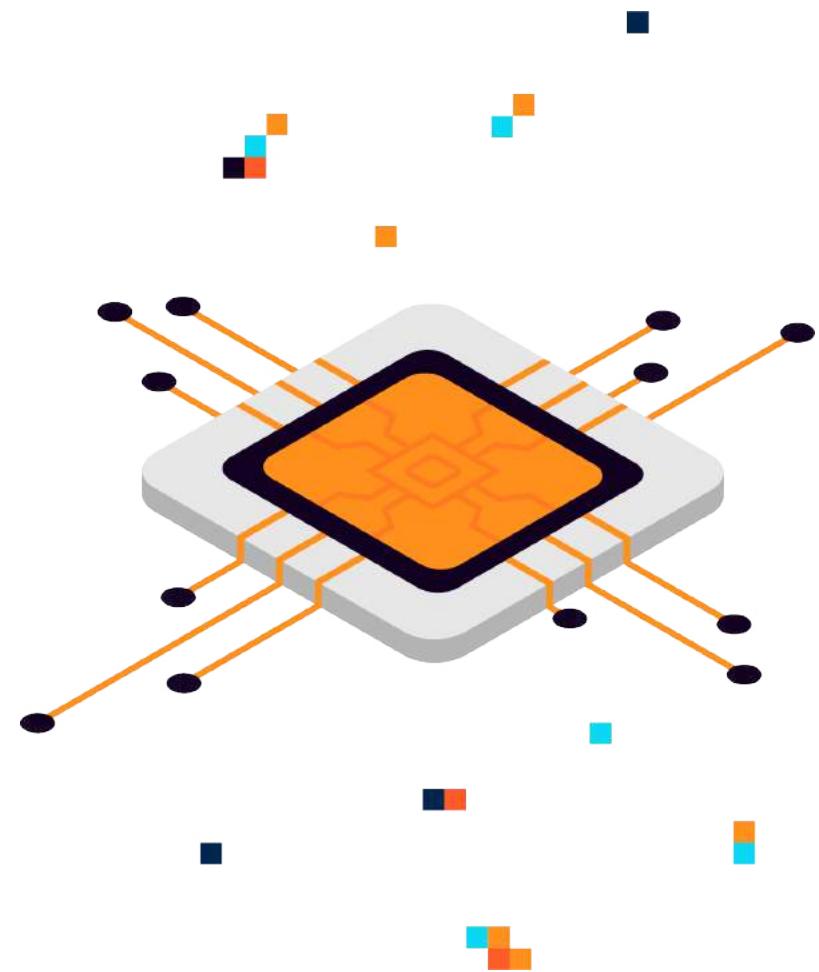


Training:

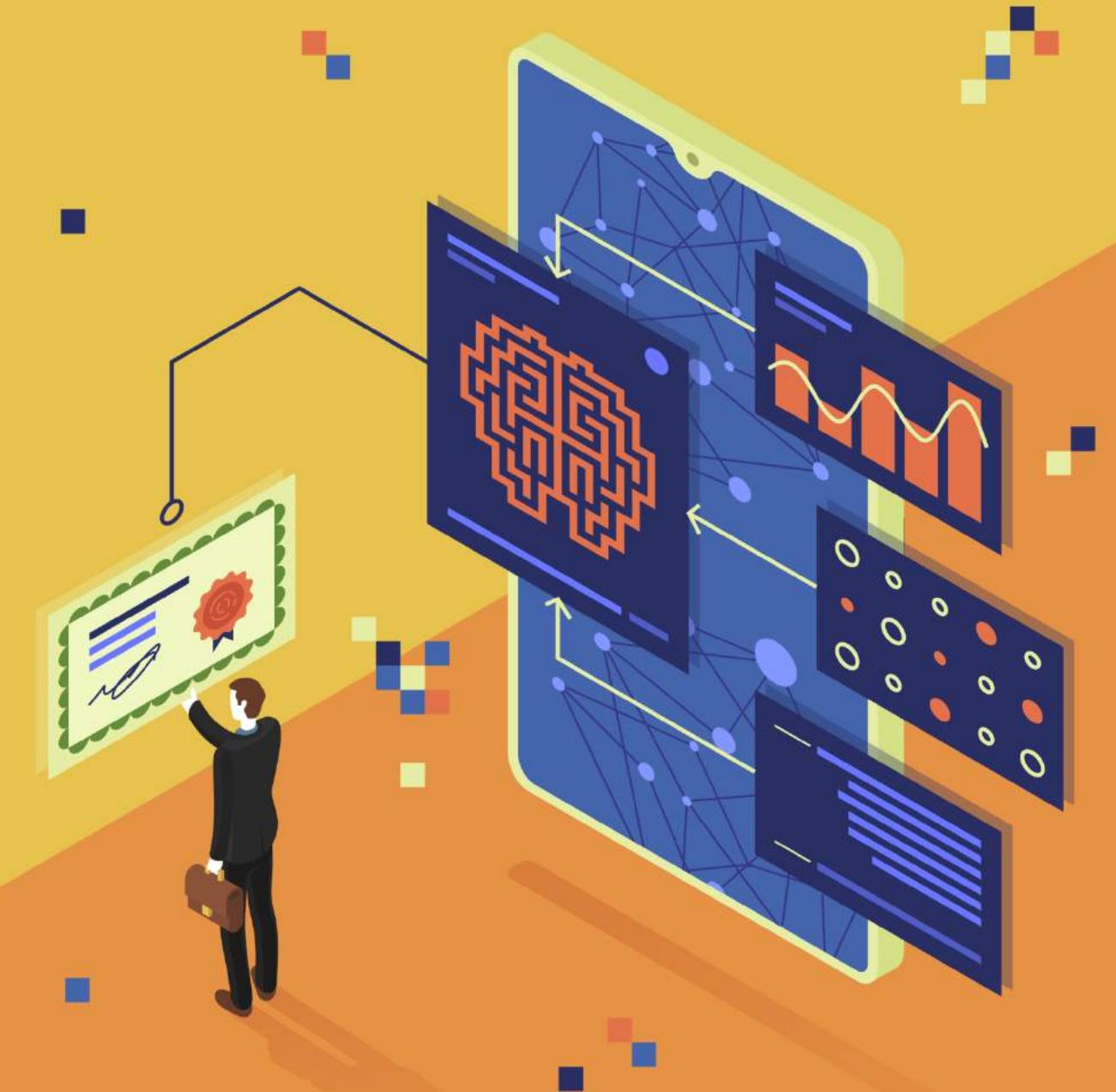
- You can train Object2vec on CPU and GPU
- For CPU, start with an ml.m5.2xlarge instance.
- For GPU, start with an ml.p2.xlarge instance.
- Larger instance is recommended such as an ml.m5.4xlarge or an ml.m5.12xlarge instance if training takes longer time.
- Object2Vec algorithm can train only on a single machine but it does offer support for multiple GPUs.

Inference:

- ml.p3.2xlarge GPU instance is recommended of deep network is utilized.
- You can use `INFERENCE_PREFERRED_MODE` environment variable due to GPU memory scarcity. GPU optimization can be selected: (Classification or Regression) or GPU optimization (Encoder Embeddings).



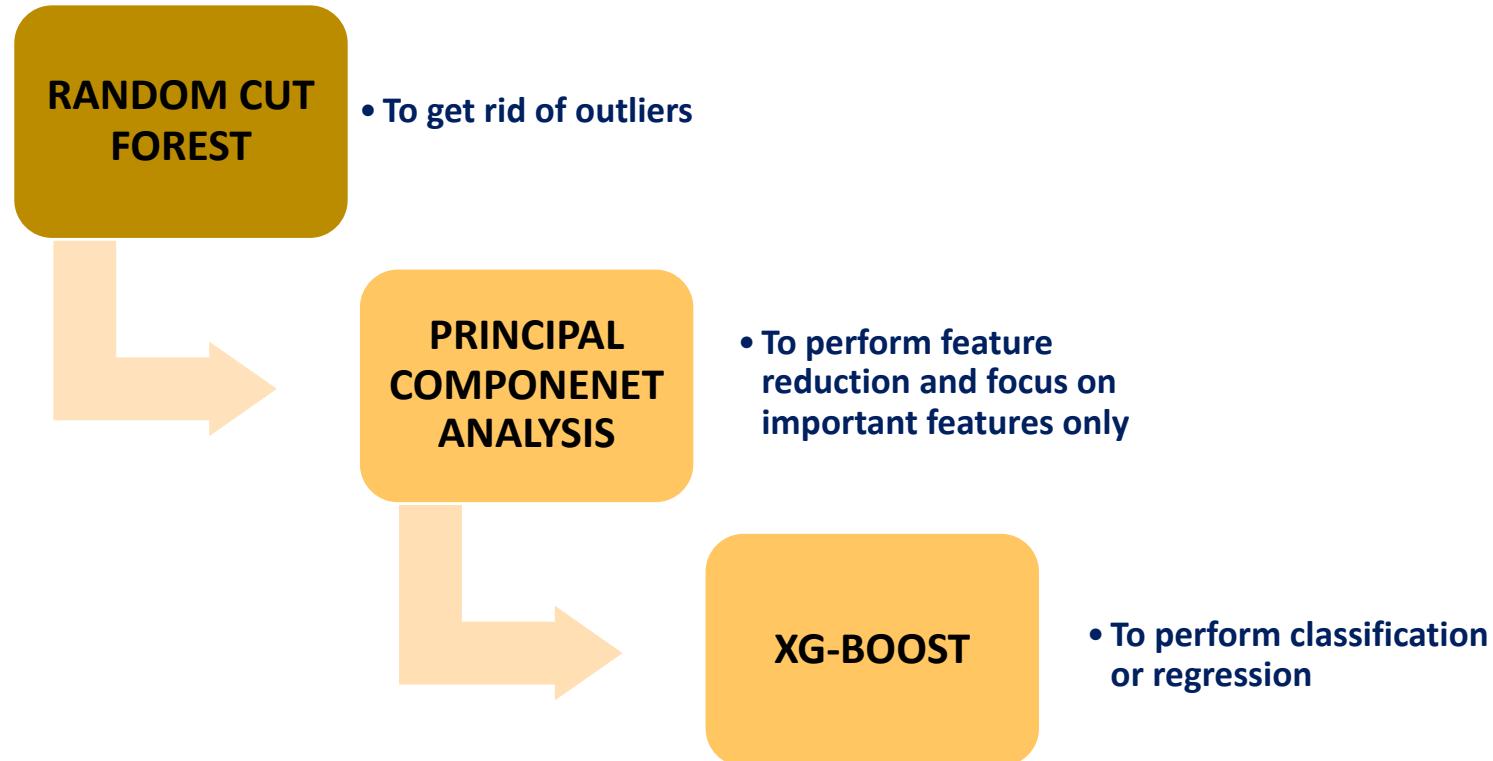
MANY LAYERED
MODELS TOGETHER



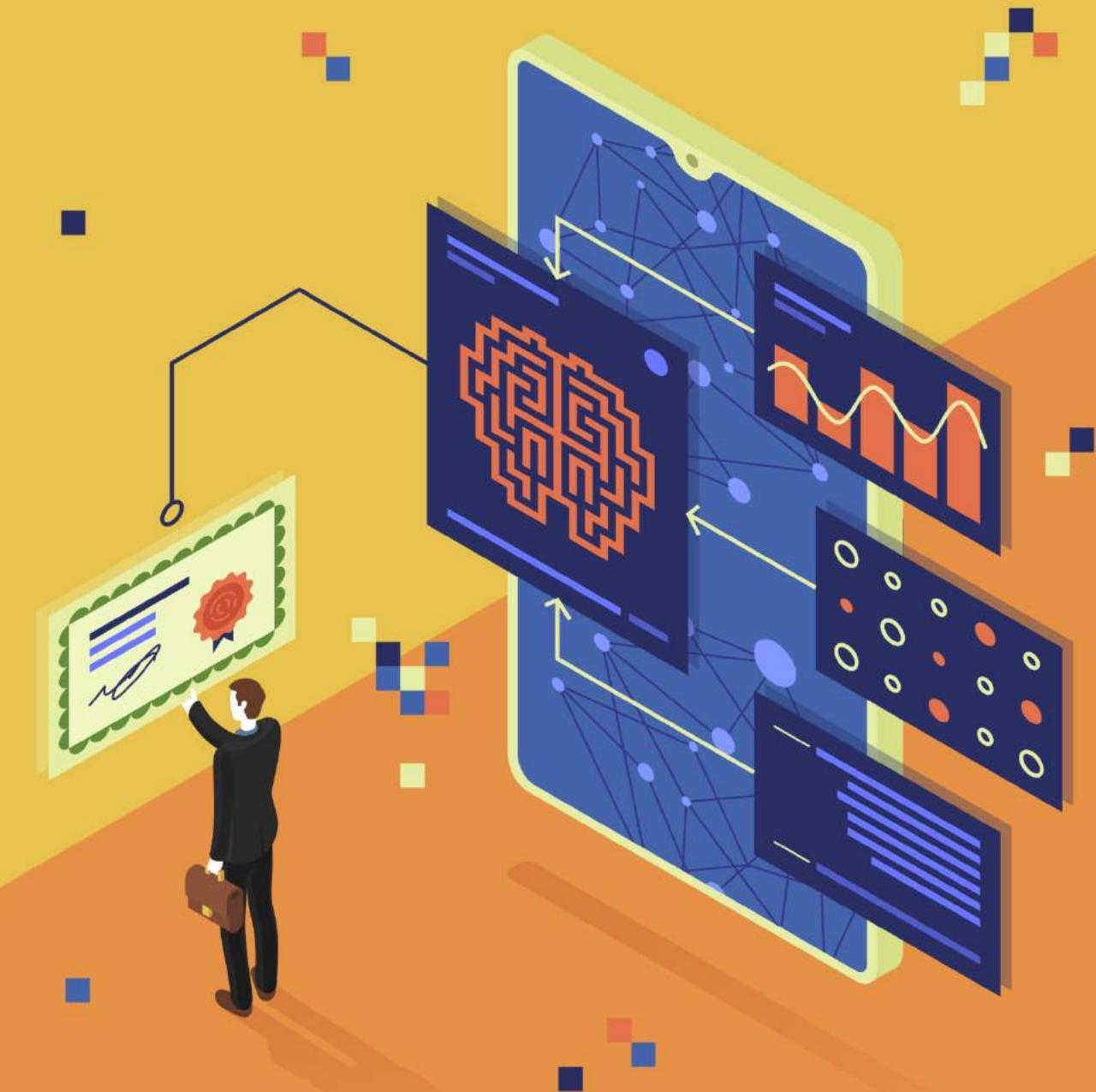
LAYERED ALGORITHMS



- Sometimes we might need to apply many algorithms in a sequential fashion as follows:



AWS SAGEMAKER AUTOMATIC MODEL TUNING

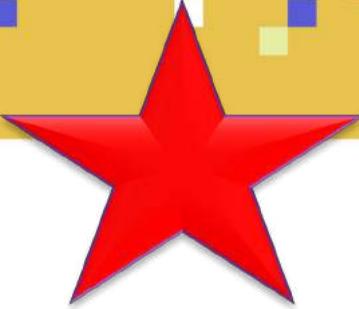


AWS SAGEMAKER AUTOMATIC MODEL TUNING

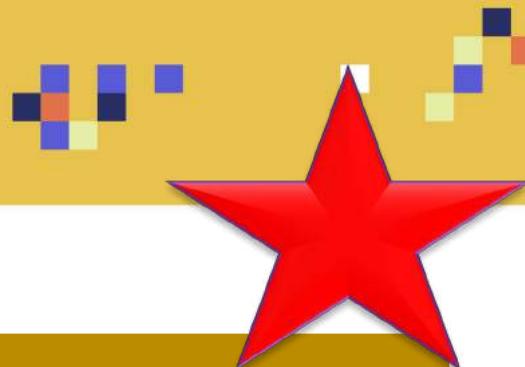
- Let's first cover the difference between parameter and hyper parameter:
 - **Hyper parameter:** values set prior to the training process such as number of neurons, layers, learning rate..etc
 - **Parameter:** values that are obtained by the training process such as network weights.
- Hyperparameter optimization is an extremely challenging task.
- Luckily, Amazon SageMaker automatic model tuning could perform hyperparameter optimization in an easy way.
- Automatic model tuning works by searching for the best version of a model.
- The user will select (1) the algorithm and (2) a range of hyperparameters (upper and lower bound) and Automatic model tuning will conduct numerous training jobs to find the optimal set of parameters.
- Automatic model tuning will then choose the final set of hyperparameter values that result in the best model.

AWS SAGEMAKER AUTOMATIC MODEL TUNING

- Example: A user can select the following:
 - **(1) Metric:** Users can select the metric to optimize for such as accuracy, precision, recall and AUC.
 - **(2) Training Algorithm:** User can select Xgboost algorithm to perform classification.
 - **(3) Range of hyperparameters:** User set the hyperparameters range such as eta, gamma, and lambda.
- Automatic model tuning will launch multiple training jobs using the range of hyperparameter values specific by the user and return the best set of parameters that optimizes for accuracy, recall or AUC (whatever metric you chose).
- The model is adaptive and learns from experience so it doesn't have to try every possible hyperparameters combination.



AWS SAGEMAKER AUTOMATIC MODEL TUNING



RANDOM SEARCH

- Random search hyper parameter optimization works by selecting random numbers from the range selected by the user.
- Parameter selection does not learn from experience (past runs).

BAYESIAN SEARCH

- Bayesian-based hyper parameter optimization works by learning from its past mistakes!
- The algorithm guesses the next combination to choose based on the previous performance.
- It treats the tuning process as a regression problem.
- Hyper parameter tuning uses an Amazon SageMaker implementation of Bayesian optimization.
- The algorithm works by applying exploration/exploitation.

AWS SAGEMAKER AUTOMATIC MODEL TUNING: BEST PRACTICES



- In order to get the best results when you run the Automatic model tuning, you need to do the following:

USE A SMALL RANGE FOR EACH OF THE HYPER PARAMETER

- For example: we know from experience that battery resistance is between 0 - 0.5 ohms, then use this range and avoid adding negative values.

AVOID RUNNING MULTIPLE TRAINING JOBS IN PARALLEL, THE AUTOMATIC MODEL TUNING LEARNS FROM EXPERIENCE

- So it's recommended that you run one tuning job first and then run another one and so on...the algorithm learns from experience.

ONLY EXPERIMENT WITH A SMALL NUMBER OF HYPER PARAMETERS

- Avoid optimizing many of them at once.
- The maximum per optimization job is capped at 20.

USE LOGARITHMIC SCALES

- Amazon SageMaker automatically tries to know if the hyperparameters are log-scaled or linear-scaled. If you know beforehand that the parameters are log-scaled, it's better to set that initially.

AWS MACHINE LEARNING CERTIFICATION



DOMAIN #3: MODELING (36% EXAM)



AWS ML CERTIFICATION EXAM DOMAINS



Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #3 OVERVIEW:

SECTION #8: MACHINE AND DEEP LEARNING BASICS – PART #1

- Artificial Neural Networks Basics: Single Neuron Model
- Activation Functions
- Multi-Layer Perceptron Model
- How do Artificial Neural Networks Train?
- ANN Parameters Tuning – Learning rate and batch size
- Tensorflow playground
- Gradient Descent and Backpropagation
- Overfitting and Under fitting
- How to overcome overfitting?
- Bias Variance Trade-off
- L1 Regularization
- L2 Regularization

SECTION #9: MACHINE AND DEEP LEARNING BASICS – PART #2

- Artificial Neural Networks Architectures
- Convolutional Neural Networks
- Recurrent Neural Networks
- Vanishing Gradient Problem
- LSTM Networks
- Model Performance Assessment – Confusion Matrix
- Model Performance Assessment – Precision, recall, F1-score
- Model Performance Assessment – ROC, AUC, Heatmap, and RMSE
- K-Fold Cross validation
- Transfer Learning
- Ensemble Learning – Bagging and Boosting

DOMAIN #3 OVERVIEW:

SECTION #10: MACHINE AND DEEP LEARNING IN AWS – BUILT-IN ALGORITHMS PART #1

- AWS SageMaker
- Deep Learning on AWS
- SageMaker Built-in algorithms
- Object Detection
- Image Classification
- Semantic Segmentation
- SageMaker Linear Learner
- Factorization Machines
- XG-Boost
- SageMaker Seq2Seq
- SageMaker DeepAR
- SageMaker Blazing Text

SECTION #11: MACHINE AND DEEP LEARNING IN AWS – BUILT-IN ALGORITHMS PART #2

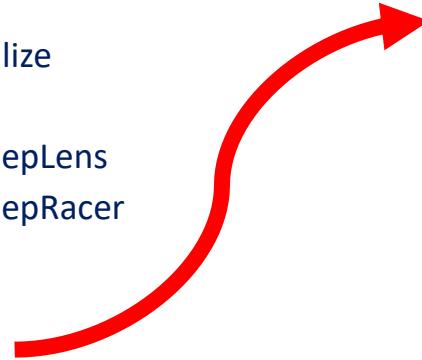
- Random Cut Forest
- Neural Topic Model
- LDA
- K-Nearest Neighbours (KNN)
- K Means
- Principal Component Analysis (PCA)
- IP insights
- Reinforcement Learning
- Object2Vec
- Automatic Model Tuning
- SageMaker and Spark

DOMAIN #3 OVERVIEW:



SECTION #12: MACHINE AND DEEP LEARNING IN AWS – HIGH LEVEL AI/ML PART #3

- ReKognition
- Amazon Comprehend and Comprehend Medical
- Translate
- Transcribe
- Polly
- Forecast
- Lex
- Personalize
- Textract
- AWS DeepLens
- AWS DeepRacer

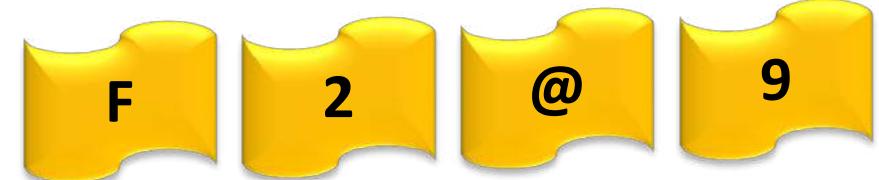


WE ARE HERE!

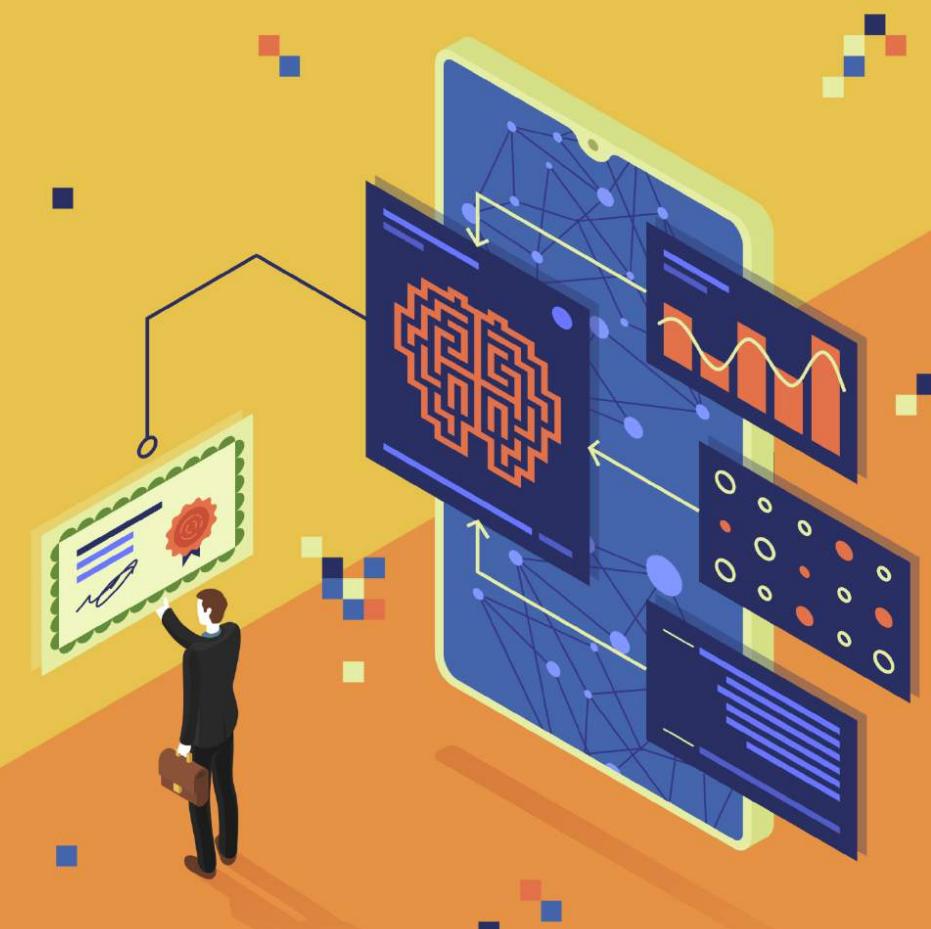
RECALL OUR MINI CHALLENGE AND PRIZE!



- For those of you who will successfully complete the entire section and watch the videos till the end, they will receive a valuable prize!



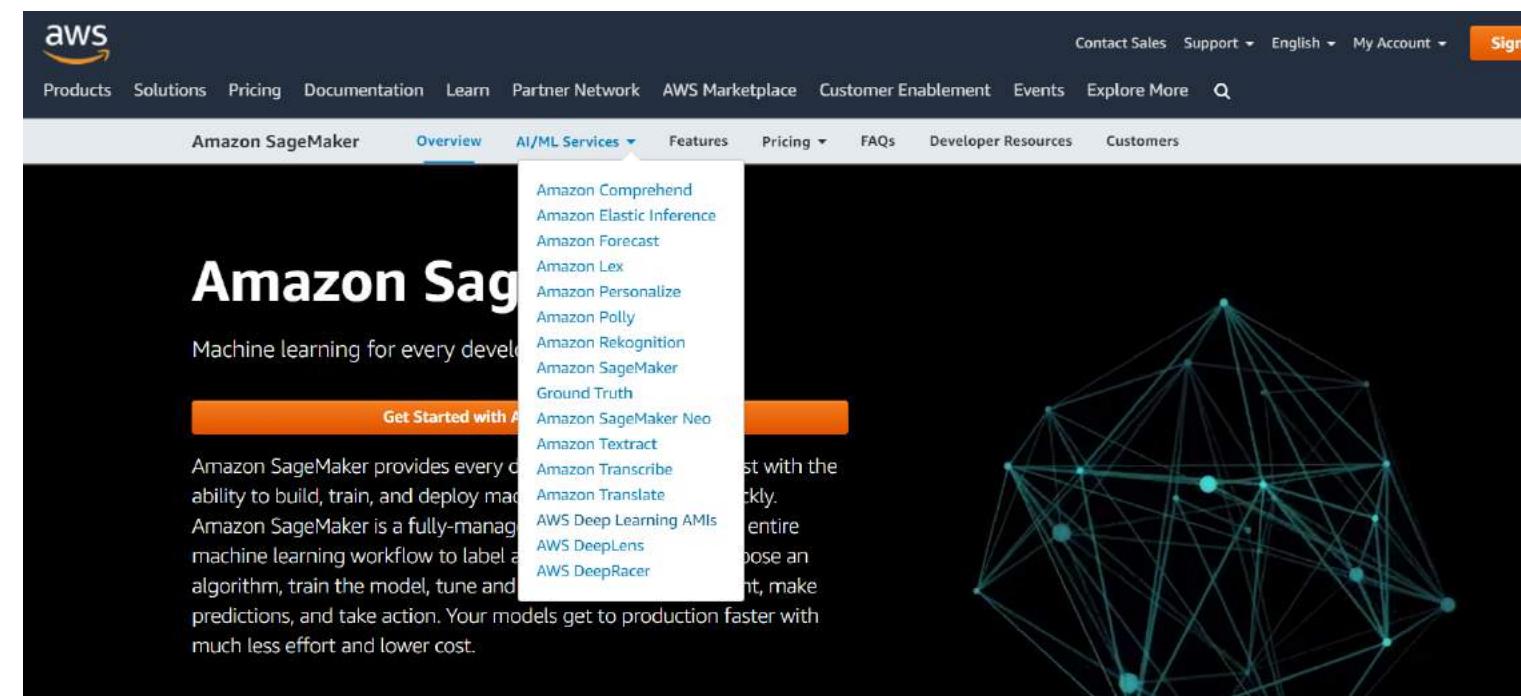
AMAZON AI/ML SERVICES



AMAZON AI/ML SERVICES



- SageMaker Algorithms presented so far require some level of programming skills to build, train and deploy AI/ML models.
- Besides that, Amazon offers a list of AI/ML services that DO NOT require any programming skills at all!!
- These are pre-trained AI services that could do the following:
 - Computer vision
 - Natural Language
 - Recommendations
 - Forecasting



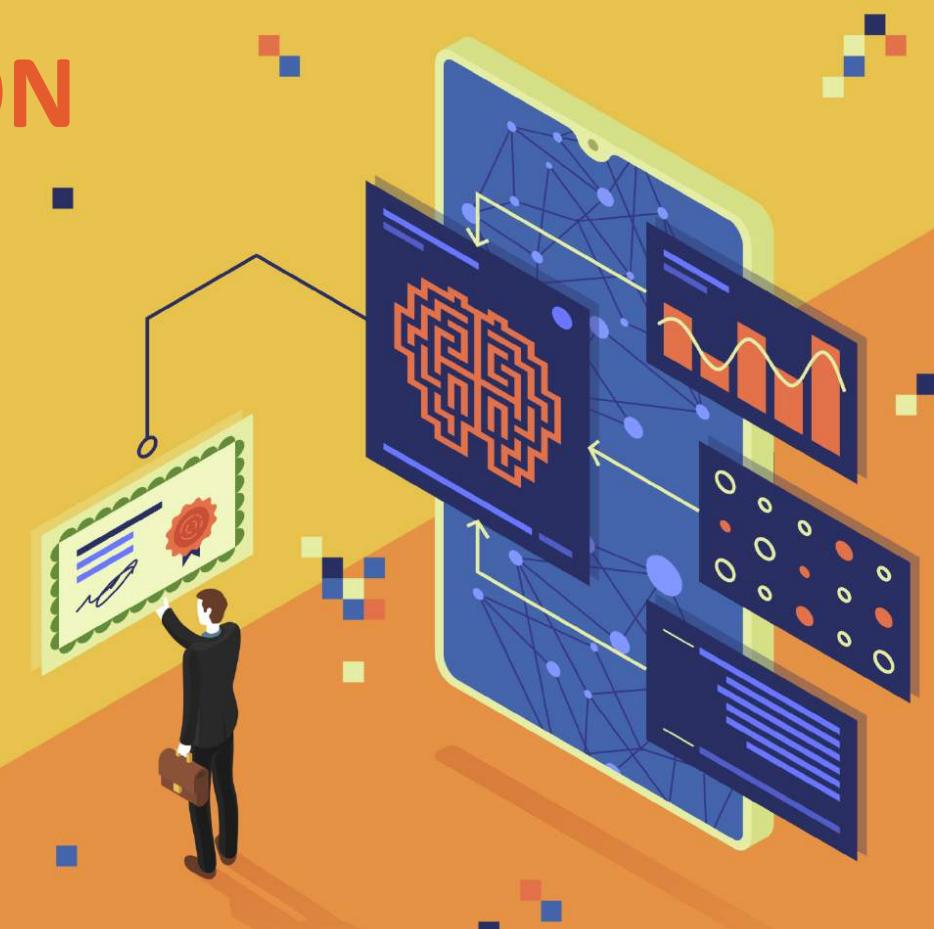
YOUTUBE VIDEO COVERING 5 HIGHLEVEL AI/ML SERVICES



https://www.youtube.com/watch?v=pwKMW_fiv-Y&t=1s



AMAZON REKOGNITION



AMAZON REKOGNITION: OVERVIEW

- Amazon Rekognition allows for video and image analysis by detecting objects, people, text and scenes.
- Amazon recognition does not require any machine learning expertise!
- It is based on advanced deep learning technology that learns every day as more data becomes available.
- Simply upload a video or image to Amazon S3 and analyze it using Amazon Rekognition.
- It could also provide:
 - Facial recognition/Facial analysis
 - Celebrity recognition
 - Face comparison
 - Text detection in image
- Facial recognition performance depends on lighting and resolution
- Video streams comes from Kinesis Video Streams H.264 encoded 5-30 FPS

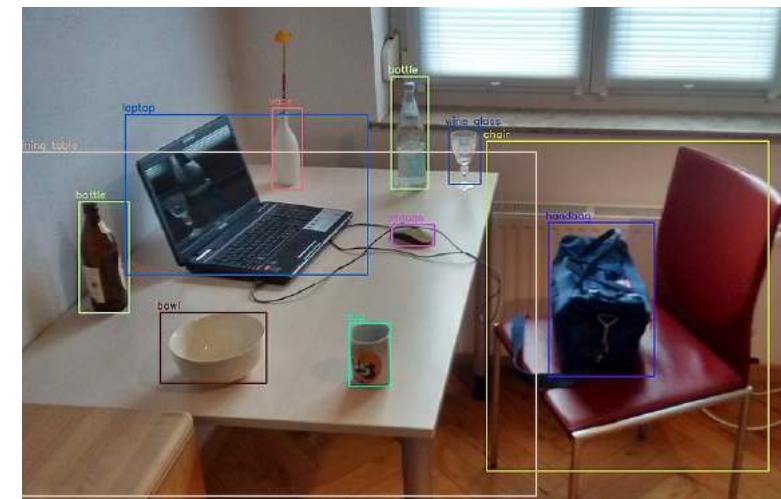


Photo Credit: <https://commons.wikimedia.org/wiki/File:Detected-with-YOLO--Schreibtisch-mit-Objekten.jpg>
Photo Credit: https://uk.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB:Face_detection.jpg

AMAZON RECOGNITION: FEATURES



EASY INTEGRATION

- Easy and simple API.

IMPROVING PERFORMANCE EVERYDAY!

- Learning, growing and becoming better as more data becomes available.

NO MANAGEMENT/HASSLE

- Amazon Rekognition can support millions of request with minimum latency.

REALTIME PERFORMANCE

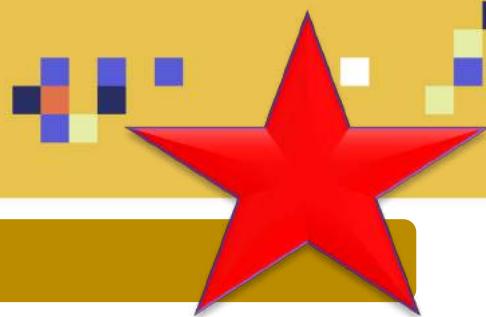
- Works in Realtime (with Kinesis Video Streams) and in batch mode to analyze millions of images.

LOW COST

- Only pay for per images and video time along with storage. Zero upfront cost.



AMAZON REKOGNITION: FEATURES



OBJECT DETECTION AND SCENE UNDERSTANDING

- Object and scene recognition besides activity detection such as “cat drinking milk”

FACIAL RECOGNITION

- Feed in a series of images to perform facial recognition.

FACIAL ANALYSIS

- Obtain information about your face such as age, level of happiness, and facial hair.

PATHING

- Amazon Rekognition can capture the path of objects in a given scene. Example: athletes movement in a field to perform post-game analysis.

UNSAFE CONTENT IDENTIFICATION

- Detect inappropriate or hazardous content.

CELEBRITY RECOGNITION

- Recognize faces of celebrities.

TEXT IN IMAGES

- Amazon Rekognition can detect text from images, such as street names and license plates.

AMAZON REKOGNITION: IN ACTION!



AWS Services Resource Groups 🔍

RyanAhmed N. Virginia Support

Amazon Rekognition Metrics Demos Object and scene detection

Rekognition automatically labels objects, concepts and scenes in your images, and provides a confidence score.

Done with the demo? [Learn more](#)

Results

City	99.1 %
Town	99.1 %
Building	99.1 %
Urban	99.1 %
Metropolis	98.5 %
High Rise	98.2 %

Show more ▶ Request ▶ Response

Object and scene detection

Choose a sample image Use your own image

Image must be .jpeg or .png format and no larger than 5MB. Your image isn't stored.

Upload or drag and drop

Label	Confidence (%)
City	99.1 %
Town	99.1 %
Building	99.1 %
Urban	99.1 %
Metropolis	98.5 %
High Rise	98.2 %

AMAZON REKOGNITION: IN ACTION!



DWS Services Resource Groups

RyanAhmed N. Virginia Support

Amazon Rekognition

Metrics

Demos

Object and scene detection

Image moderation

Facial analysis

Celebrity recognition

Face comparison

Text in image

Video Demos

Video analysis

Additional Resources

Getting started guide

Download SDKs

Developer resources

Pricing

FAQ

Forum

Facial analysis

Get a complete analysis of facial attributes, including confidence scores.

Choose a sample image

Use your own image

Image must be .jpeg or .png format and no larger than 5MB. Your image isn't stored.

Upload or drag and drop

Done with the demo? [Learn more](#)

Results

	>
looks like a face	99.9 %
appears to be male	99.4 %
age range	22 - 34 years old
smiling	99.9 %
appears to be happy	99.7 %
not wearing glasses	99.6 %
not wearing sunglasses	99.9 %
eyes are open	93.1 %
mouth is open	99.8 %
does not have a mustache	83.5 %

AMAZON REKOGNITION: IN ACTION!



AWS Services Resource Groups

Get a complete analysis of facial attributes, including confidence scores.

Amazon Rekognition

- Metrics
- Demos
- Object and scene detection
- Image moderation
- Facial analysis**
- Celebrity recognition
- Face comparison
- Text in image

Video Demos

- Video analysis

Additional Resources

- Getting started guide
- Download SDKs
- Developer resources
- Pricing
- FAQ
- Forum

Choose a sample image

Use your own image
Image must be .jpeg or .png format and no larger than 5MB. Your image isn't stored.

Upload or drag and drop

looks like a face 100 %

appears to be male 94.3 %

age range 23 - 35 years old

smiling 99.8 %

appears to be happy 99.3 %

not wearing glasses 99.1 %

not wearing sunglasses 99.8 %

eyes are open 64.7 %

mouth is open 98.9 %

does not have a mustache 98.4 %

does not have a beard 81.2 %

Done with the demo? [Learn more](#)

AMAZON COMPREHEND



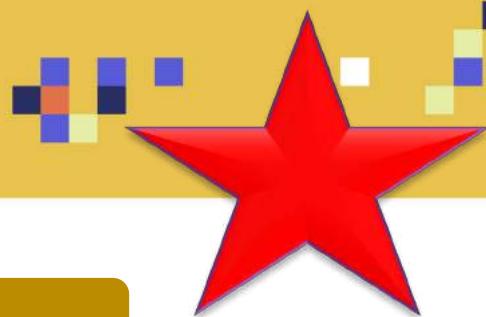
AMAZON COMPREHEND



- Amazon Comprehend is a natural language processing (NLP) service that relies on Machine learning/Artificial Intelligence to gain valuable insights from text.
- You do not need to know any machine learning or programming to use Amazon Comprehend!
- Use cases:
 - Use Comprehend to analyze customer-related text such as product reviews, social media posts, and emails.
 - This is crucial for companies because now they can perform sentiment analysis on the data (getting a feel of whether customers are happy or not!)
- Amazon Comprehend follows pay per use model
- No need for machine learning experience and servers to manage
- Comprehend is capable of:
 - Identify important phrases
 - List Brands
 - Sentiment analysis (whether customers are happy or not)
 - Extract locations and places



AMAZON COMPREHEND: FEATURES



EXTRACT CRITICAL CUSTOMER INSIGHTS

- Amazon Comprehend can provide important information about customers such as whether they are happy or not with the produce or service. This is done by analyzing customers data from social media posts and product reviews.

DOCUMENT ORGANIZATION

- Amazon Comprehend can take in a series of documents and organize them by topics. You can then use these organized topics to offer customized/targeted customer experience.

EASILY TRAINED ON YOUR OWN DATA (ZERO ML EXPERIENCE)

- Anyone can feed in data to Comprehend and it identifies key elements in the data such as part codes. It could also organize social media posts by product for example.

WORKS WITH GENERAL AND SPECIFIC INDUSTRY

- Amazon comprehend can be used for general text or for industry specific cases such as Amazon comprehend Medical (see next slide!)

AMAZON COMPREHEND MEDICAL



- Comprehend offers a specialized service in the medical field known as Amazon Comprehend Medical.
- Amazon comprehend medical can take in unstructured medical data such as doctor's notes and patient health records and generate medical information from unstructured data such as:
 - Medications
 - Dosages
 - Strengths
 - Frequencies
- Use case:
 - **Input:** feed in Amazon Comprehend Medical with unstructured doctors notes
 - **Output:** Amazon comprehend will generate medications, dosage, strength, and frequency



AMAZON COMPREHEND: IN ACTION!

The screenshot shows the AWS Amazon Comprehend service interface under the 'Real-time analysis' section. On the left, a sidebar lists 'Real-time analysis' (selected), 'Analysis jobs', 'Customization' (with 'Custom classification' and 'Custom entity recognition' options), and 'Amazon Comprehend Medical' (with 'Real-time analysis' and 'Analysis jobs' options). The main content area has a blue header bar with 'Learn more' and a description about indexing and analyzing unstructured text. Below this, the page title is 'Amazon Comprehend > Real-time analysis'. The central part is titled 'Real-time analysis' with an 'Info' link. It contains a text input box with the placeholder 'Input text' and a 'Supported languages' dropdown. A large red arrow points from the word 'INPUTS' to this text input box. Below it is a larger text area containing a sample sentence about Udemy. A second red arrow points from the word 'OUTPUTS' to a dashed red oval highlighting the 'Insights' section at the bottom. This section includes tabs for 'Entities', 'Key phrases', 'Language', 'Sentiment', and 'Syntax', with the 'Entities' tab currently selected.

INPUTS

OUTPUTS

Entities | Key phrases | Language | Sentiment | Syntax

AMAZON COMPREHEND: ENTITIES AND KEY PHRASES

Amazon Comprehend X

Services ▾ Resource Groups ▾

Entities Key phrases Language Sentiment Syntax

Analyzed text

Udemy.com is an online learning platform aimed at professional adults and students. Udemy, a portmanteau of you + academy, has more than 30 million students and 50,000 instructors teaching courses in over 60 languages. There have been over 245 million course enrollments. Students and instructors come from 190+ countries and 2/3 of students are located outside of the U.S. Udemy also has over 4,000 enterprise customers and 80% of Fortune 100 companies use Udemy for employee upskilling (Udemy for Business). Students take courses largely as a means of improving job-related skills.[2] Some courses generate credit toward technical certification. Udemy has made a special effort to attract corporate trainers seeking to create coursework for employees of their company.[3] As of 2019, there are more than 130,000 courses on the website.[4]

Results

Entity	Category	Confidence
Udemy.com	Title	0.51
Udemy	Organization	0.99+
more than 30 million students	Quantity	0.99+
50,000 instructors	Quantity	0.99+
60 languages	Quantity	0.95
over 245 million course	Quantity	0.99+
190+ countries	Quantity	0.99+
2/3 of students	Quantity	0.99+

Amazon Comprehend X

Services ▾ Resource Groups ▾

Entities Key phrases Language Sentiment Syntax

Real-time analysis

Analysis jobs

Customization

Custom classification

Custom entity recognition

Amazon Comprehend Medical

Real-time analysis

Analysis jobs

Analyzed text

Udemy.com is an online learning platform aimed at professional adults and students. Udemy, a portmanteau of you + academy, has more than 30 million students and 50,000 instructors teaching courses in over 60 languages. There have been over 245 million course enrollments. Students and instructors come from 190+ countries and 2/3 of students are located outside of the U.S. Udemy also has over 4,000 enterprise customers and 80% of Fortune 100 companies use Udemy for employee upskilling (Udemy for Business). Students take courses largely as a means of improving job-related skills.[2] Some courses generate credit toward technical certification. Udemy has made a special effort to attract corporate trainers seeking to create coursework for employees of their company.[3] As of 2019, there are more than 130,000 courses on the website.[4]

Results

Key phrases	Confidence
Udemy.com	0.98
an online learning platform	0.99+
professional adults and students	0.96
Udemy	0.64
a portmanteau	0.99+
+ academy	0.89
more than 30 million students	0.98

AMAZON COMPREHEND: LANGUAGE AND SENTIMENT

AWS Services Resource Groups *

Amazon Comprehend

Real-time analysis Analysis jobs Customization Custom classification Custom entity recognition

▼ Amazon Comprehend Medical Real-time analysis Analysis jobs

Udemy, a portmanteau of you + academy, has more than 30 million students and 50,000 instructors teaching courses in over 60 languages. There have been over 245 million course enrollments. Students and instructors come from 190+ countries and 2/3 of students are located outside of the U.S. Udemy also has over 4,000 enterprise customers and 80% of Fortune 100 companies use Udemy for employee upskilling (Udemy for Business). Students take courses largely as a means of improving job-related skills.[2] Some courses generate credit toward technical certification. Udemy has made a special effort to attract corporate trainers seeking to create coursework for employees of their company.[3]

842 of 5000 characters used.

Analyze

Insights Info

Entities Key phrases Language Sentiment Syntax

Analyzed text

Udemy.com is an online learning platform aimed at professional adults and students. Udemy, a portmanteau of you + academy, has more than 30 million students and 50,000 instructors teaching courses in over 60 languages. There have been over 245 million course enrollments. Students and instructors come from 190+ countries and 2/3 of students are located outside of the U.S. Udemy also has over 4,000 enterprise customers and 80% of Fortune 100 companies use Udemy for employee upskilling (Udemy for Business). Students take courses largely as a means of improving job-related skills.[2] Some courses generate credit toward technical certification. Udemy has made a special effort to attract corporate trainers seeking to create coursework for employees of their company.[3] As of 2019, there are more than 130,000 courses on the website.[4]

▼ Results

Language

English, en
0.99 confidence

► Application integration

AWS Services Resource Groups *

Amazon Comprehend

Real-time analysis Analysis jobs Customization Custom classification Custom entity recognition

▼ Amazon Comprehend Medical Real-time analysis Analysis jobs

Udemy, a portmanteau of you + academy, has more than 30 million students and 50,000 instructors teaching courses in over 60 languages. There have been over 245 million course enrollments. Students and instructors come from 190+ countries and 2/3 of students are located outside of the U.S. Udemy also has over 4,000 enterprise customers and 80% of Fortune 100 companies use Udemy for employee upskilling (Udemy for Business). Students take courses largely as a means of improving job-related skills.[2] Some courses generate credit toward technical certification. Udemy has made a special effort to attract corporate trainers seeking to create coursework for employees of their company.[3]

842 of 5000 characters used.

Analyze

Insights Info

Entities Key phrases Language Sentiment Syntax

Analyzed text

Udemy.com is an online learning platform aimed at professional adults and students. Udemy, a portmanteau of you + academy, has more than 30 million students and 50,000 instructors teaching courses in over 60 languages. There have been over 245 million course enrollments. Students and instructors come from 190+ countries and 2/3 of students are located outside of the U.S. Udemy also has over 4,000 enterprise customers and 80% of Fortune 100 companies use Udemy for employee upskilling (Udemy for Business). Students take courses largely as a means of improving job-related skills.[2] Some courses generate credit toward technical certification. Udemy has made a special effort to attract corporate trainers seeking to create coursework for employees of their company.[3] As of 2019, there are more than 130,000 courses on the website.[4]

▼ Results

Sentiment

Sentiment	Confidence
Neutral	0.98 confidence
Positive	0.01 confidence
Negative	0.00 confidence
Mixed	0.00 confidence

► Application integration

AMAZON COMPREHEND: SYNTAX

The screenshot shows the AWS Amazon Comprehend service interface. The left sidebar lists various analysis options like Real-time analysis, Customization, and Amazon Comprehend Medical. The main area displays an analyzed text snippet about Udemy and its results table.

Analyzed text:

Udemy.com is an online learning platform aimed at professional adults and students. Udemy, a portmanteau of you + academy, has more than 30 million students and 50,000 instructors teaching courses in over 60 languages. There have been over 245 million course enrollments. Students and instructors come from 190+ countries and 2/3 of students are located outside of the U.S. Udemy also has over 4,000 enterprise customers and 80% of Fortune 100 companies use Udemy for employee upskilling (Udemy for Business). Students take courses largely as a means of improving job-related skills.[2] Some courses generate credit toward technical certification. Udemy has made a special effort to attract corporate trainers seeking to create coursework for employees of their company.[3] As of 2019, there are more than 130,000 courses on the website.[4]

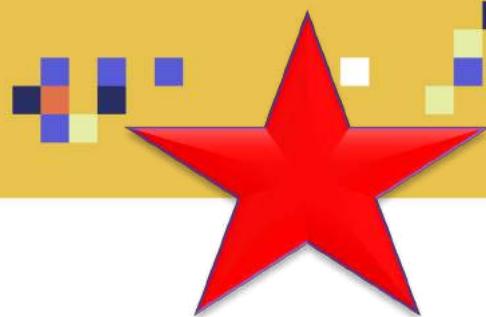
Results:

Word	Part of speech	Confidence
Udemy.com	Proper noun	0.67
is	Verb	0.99+
an	Determiner	0.99+
online	Adjective	0.84
learning	Noun	0.91
platform	Noun	0.99+
aimed	Verb	0.99+
at	Adposition	0.99+

AMAZON TRANSLATE

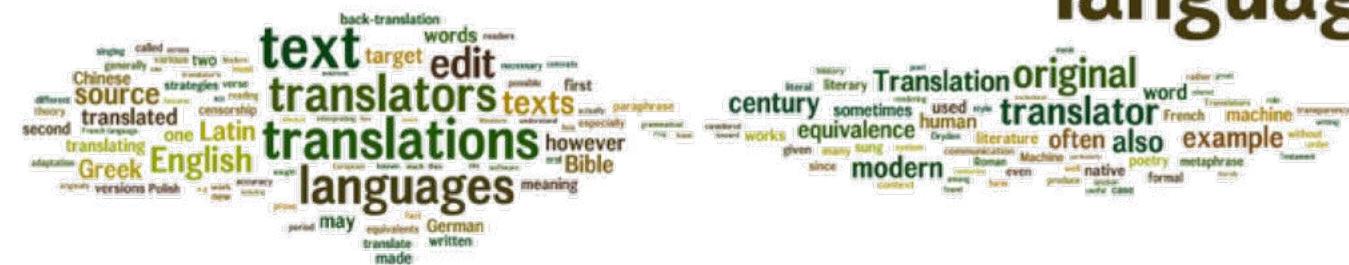


AMAZON TRANSLATE



- Amazon Translate offers a translation service using deep learning.
 - Amazon translate offers multiple language translation so it makes it easier for corporates and content producers to localize content to any country ,
 - Amazon translate offer more natural, robust and accurate translation compared to basic rule-based translation methods.

translation language



AMAZON TRANSLATE: FEATURES



CONTINOUS PERFORMANCE IMPROVEMENT

- Amazon Translate is based on deep learning. The algorithms are getting better everyday as more data becomes available.

EASY INTEGRATION

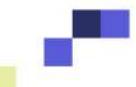
- Amazon Translate offers a simple API so it simplifies the process of building real-time application.

CUSTOMIZABLE

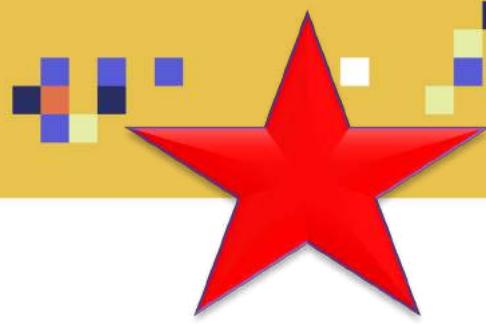
- Minimum need of editing by human translators. Amazon Translate offers customization features that allows you to select how to translate unique terms.

EASILY SCALABLE

- Amazon translate can be easily scaled and works with words, paragraphs and large scale documents.



AMAZON TRANSLATE: USE CASES



USE CASE #1: PERFORM SENTIMENT ANALYSIS ON CONTENT WITH MULTIPLE LANGUAGES

- By integrating Amazon Translate with Amazon Comprehend, you can get critical information about your customers such as reviews, are they happy with the product or not..etc.
- This could be done by using Amazon translate to convert from any language to English and then run Amazon Comprehend (natural language processing (NLP) application) to make sentiment analysis predictions

USE CASE #2: PERFORM REALTIME CONTENT TRANSLATION

- In Realtime, Amazon translate can translate content such as feed stories, news, and comments.
- By relying on English speaking workforce only to address customers Q&A, Amazon Translate can provide real-time translation to chat and email from any language to English.

AMAZON TRANSLATE: LANGUAGES SAMPLE

The screenshot shows the AWS Lambda service page. At the top, there's a navigation bar with 'Services' and 'Resource Groups'. Below the header, the page title is 'Amazon Translate' with the subtitle 'Natural and fluent translation'. A sub-header states: 'Amazon Translate provides fast, affordable, and quality translations for your multilingual needs.' Under the 'How it works' section, it says: 'Amazon Translate uses deep learning techniques to produce more accurate and fluent translation than traditional statistical and rule-based translation models.' In the 'Features and benefits' section, there are two items: 'High-quality' and 'Real-time'.

Target language

A dropdown menu titled 'Target language' is open. The first item, 'English (en)', is highlighted with a blue border. Other options listed include Arabic (ar), Chinese (zh), Chinese Traditional (zh-TW), Czech (cs), Danish (da), Dutch (nl), English (en), Finnish (fi), French (fr), German (de), Greek (el), Hebrew (he), Hindi (hi), Hungarian (hu), Indonesian (id), Italian (it), Japanese (ja), Korean (ko), Malay (ms), and Norwegian (no). A search bar is also visible at the top of the dropdown.

AMAZON TRANSLATE: ENGLISH TO ARABIC

The screenshot shows the AWS Amazon Translate interface. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, and user Rya. The left sidebar has tabs for Metrics, Real-time translation (which is selected), and Custom Terminology. The main content area is titled "Real-time translation" with an "Info" link. It shows a "Translation" section with "Source language" set to "Auto (auto)" and "Target language" set to "Arabic (ar)". A central text box contains English text about Udemy, and its Arabic translation is shown below it. At the bottom, there's a note about Udemy's 2019 statistics, a feedback link, and an "Additional settings" button.

Amazon Translate < Real-time translation

Real-time translation Info

Translation

Source language

Auto (auto)

Udemy.com is an online learning platform aimed at professional adults and students.

Udemy, a portmanteau of you + academy, has more than 30 million students and 50,000 instructors teaching courses in over 60 languages. There have been over 245 million course enrollments. Students and instructors come from 190+ countries and 2/3 of students are located outside of the U.S. Udemy also has over 4,000 enterprise customers and 80% of Fortune 100 companies use Udemy for employee upskilling (Udemy for Business). Students take courses largely as a means of improving job-related skills.^[2] Some courses generate credit toward technical certification. Udemy has made a special effort to attract corporate trainers seeking to create coursework for employees of their company.^[3]

As of 2019, there are more than 130,000 courses on the website.^[4]

842 characters, 842 of 5000 bytes used. [Info](#)

Detected language: English (en)

► Additional settings

Target language

Arabic (ar)

Udemy.com هو منصة تعلم عبر الإنترنت تهدف إلى البالغين والطلاب المبتدئين. Udemy، أحد أكاديمية+ مكم، أكثر من 30 مليون طالب و 50.000 مدرب تدريس دورات في أكثر من 60 لغة. وقد بلغ عدد المانحين بالدورات الدراسية أكثر من 245 مليون. الطلاب والمدربون يأتون من أكثر من 190 دولة، ويبلغ 2/3 من الطلاب خارج الولايات المتحدة الأمريكية لديها أيضًا أكثر من 4.000 عميل المؤسسة و 80% من الشركات في أكثر من 100 استخدام Udemy for Business (Udemy for Business) يأخذ الطلاب دورات إلى حد كبير كوسيلة لتحسين المهارات المتعلقة بالعمل.^[2] بعض الدورات تولد الاصحاح نحو الاعتماد الذي، بذلك، جيداً خاصة لجذب المدربين الشركات الذين يسعون إلى إنشاء دورات تدريبية لموظفي شركتهم.^[3]

اعتباراً من عام 2019، هناك أكثر من 130.000 دورة على الموقع.^[4]

Is this translation what you expected? Please leave us [feedback](#)

AMAZON TRANSCRIBE



AMAZON TRANSCRIBE



- Amazon Transcribe is a speech to text service (Automatic Speech Recognition (ASR)).
- You can feed in audio files from Amazon S3 to Amazon Transcribe and then get text file as an output.
- Amazon transcribe works in Realtime as well! Feed in live audio steam and generate transcripts on the fly!
- Applications:
 - Movie/YouTube Subtitles
 - Customer audio calls transcription

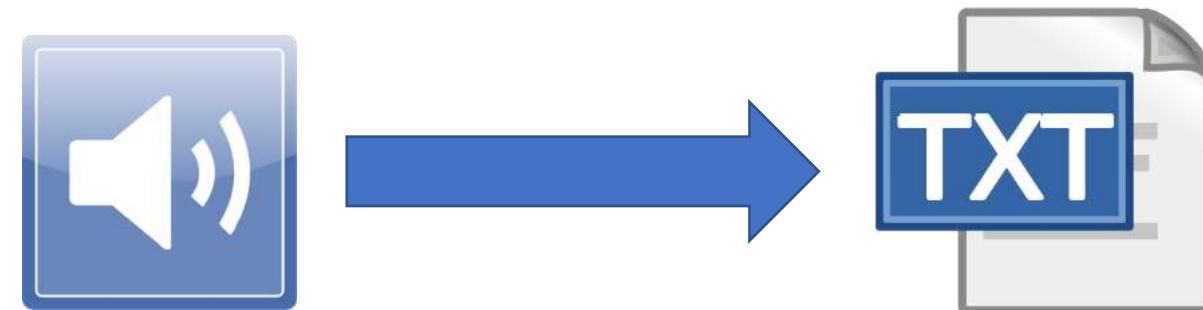
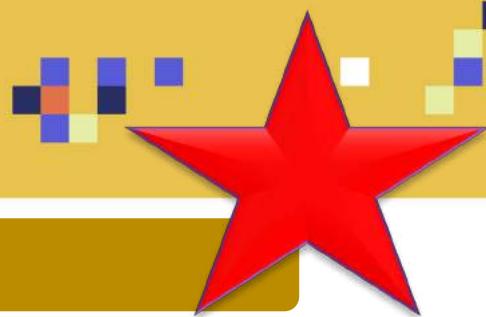


Photo Credit: <https://pixabay.com/vectors/audio-button-volume-box-glossy-152943/>

Photo Credit: <https://commons.wikimedia.org/wiki/File:Text-txt.svg>

AMAZON TRANSCRIBE: KEY FEATURES



NATURAL TRANSCRIPTIONS WITH PUNCTUATION

- Using advanced Deep Learning algorithms, Amazon Transcribe offers punctuation and formatting automatically

GENERATES TIMESTAMPS

- Amazon Transcribe returns a timestamp for each word, so that you can easily locate the audio in the original recording by searching for the text.

MANY USE CASES

- Works great with poor Audio quality such as customer phone calls

RICH VOCABULARY

- Amazon transcribe allows users to add their own custom vocabulary words to enrich its content.

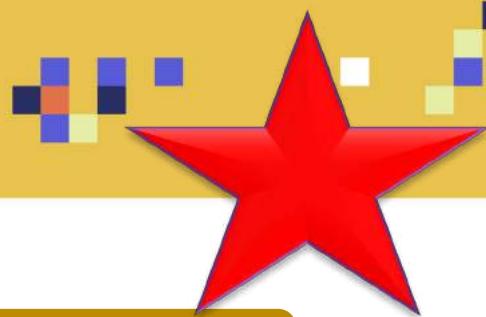
TRANSCRIBE AUDIO FROM MANY SPEAKERS

- Amazon Transcribe is capable of classifying text from many speakers. This is extremely efficient with cases such as phone calls and meeting transcriptions

MULTI CHANNEL IDENTIFICATION

- Amazon Transcribe is able to consume audio files from many channels.

AMAZON TRANSCRIBE: USE CASES



USE CASE #1: ENHANCED CUSTOMER SERVICE

- Integrating Amazon Transcribe with Amazon Comprehend can allow companies to record voice data such as phone calls, converting them into text (Using Transcribe) and then analyzing them (with comprehend) to obtain sentiment, intent and valuable insights.

USE CASE #2: AUTOMATIC SUBTITLES

- Amazon Transcribe could allow content creators to easily and effectively generate subtitles for their videos. Time stamps is also provided!

USE CASE #3: ARCHIVING AUDIO CONTENT

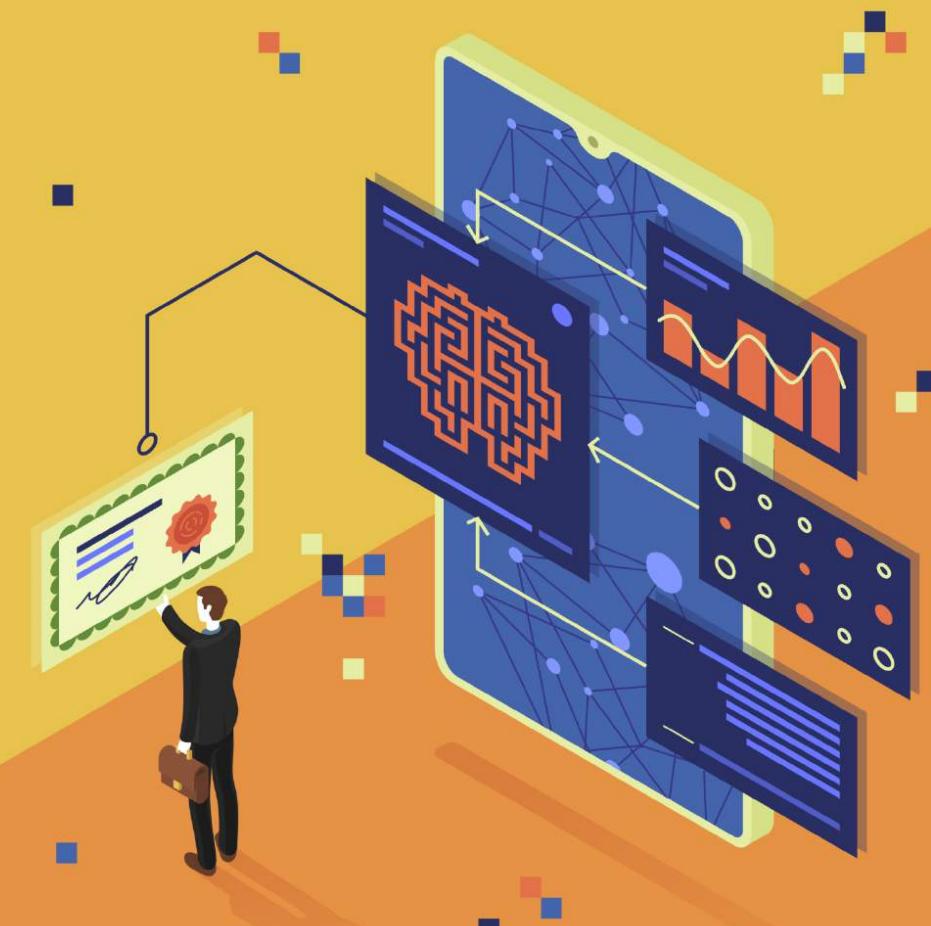
- To enhance corporate compliance and reduce risk for enterprises, companies can use transcribe to convert audio and video content into text (searchable archives). Amazon Elasticsearch service can be later used to perform text-based search across audio/video library.

AMAZON TRANSCRIBE: IN ACTION!



The screenshot shows the Amazon Transcribe service interface. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, 'Resource Groups' dropdown, and a bell icon. The main title is 'Amazon Transcribe' with a close button 'X'. Below it, the path 'Amazon Transcribe > Real-time transcription' is shown. On the left, a sidebar has 'Real-time transcription' selected (in orange), along with links for 'Transcription jobs' and 'Custom vocabulary'. The main content area is titled 'Real-time transcription' with an 'Info' link. It contains a sub-instruction: 'See how Amazon Transcribe creates a text copy of speech in real time. Choose **Start streaming** and talk.' Below this, a 'Transcription' section has a 'Download full transcript' button and an orange 'Start streaming' button. A language dropdown is set to 'English (United States)'. A text box displays two lines of transcribed text: 'I love this course because it would make me pass the machine learning certification exam.' followed by 'I love this course because it would make me pass the machine learning certification exam.'. Below the text box, it says '00:00 of 15:00 audio stream'. There are also 'Additional settings' and 'Application integration' sections.

AMAZON POLLY



AMAZON POLLY: OVERVIEW



- Amazon Polly is a Text to Speech (TTS) service.
- Amazon Polly offers natural speech by relying on Deep Learning.
- Users can select many voices and languages.
- Amazon Polly offers Neural Text-to-Speech (NTTS) voices makes it extremely natural!

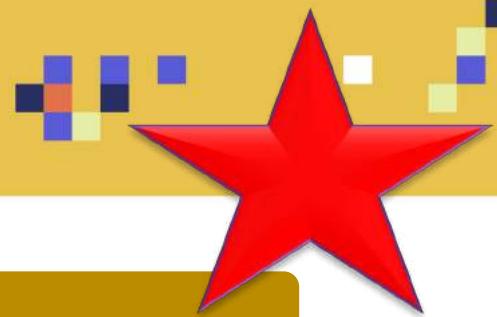


[Photo Credit: https://pixabay.com/vectors/audio-button-volume-box-glossy-152943/](https://pixabay.com/vectors/audio-button-volume-box-glossy-152943/)

[Photo Credit: https://commons.wikimedia.org/wiki/File:Text-txt.svg](https://commons.wikimedia.org/wiki/File:Text-txt.svg)

[Photo Credit: https://commons.wikimedia.org/wiki/File:Polly_want_a_finger-1_\(5325523581\).jpg](https://commons.wikimedia.org/wiki/File:Polly_want_a_finger-1_(5325523581).jpg)

AMAZON POLLY: BENEFITS



NATURAL VOICE

- Offers many languages with multiple lifelike voices (males/females)

SPEECH STORAGE

- Amazon Polly allows for free unlimited playback of voice audio formats like MP3 and OGG.

STEAMING

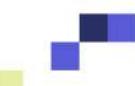
- Voices could be played immediately in Realtime.

CUSTUMIZABLE OUTPUT

- Polly supports lexicons and SSML tags that allows users to control pronunciation, volume, pitch, speed rate, etc. Example: yolo = you only live once!

COST EFFECTIVE

- Pay per use model.



AMAZON POLLY: IN ACTION!



AWS Services Resource Groups RyanAhmed

Amazon Polly

Text-to-Speech

Listen, customize, and download speech. Integrate when you're ready.

Type or paste your text in the window, choose your language and region, choose a voice, choose Listen to speech, and then integrate it into your applications and services.

With up to 3000 characters you can listen, download, or save immediately. For up to 100,000 characters, your task must be saved to an S3 bucket.

Plain text SSML ?

Hi! I'm Matthew. I will read any text you type here.

52 characters used Show default text Clear text

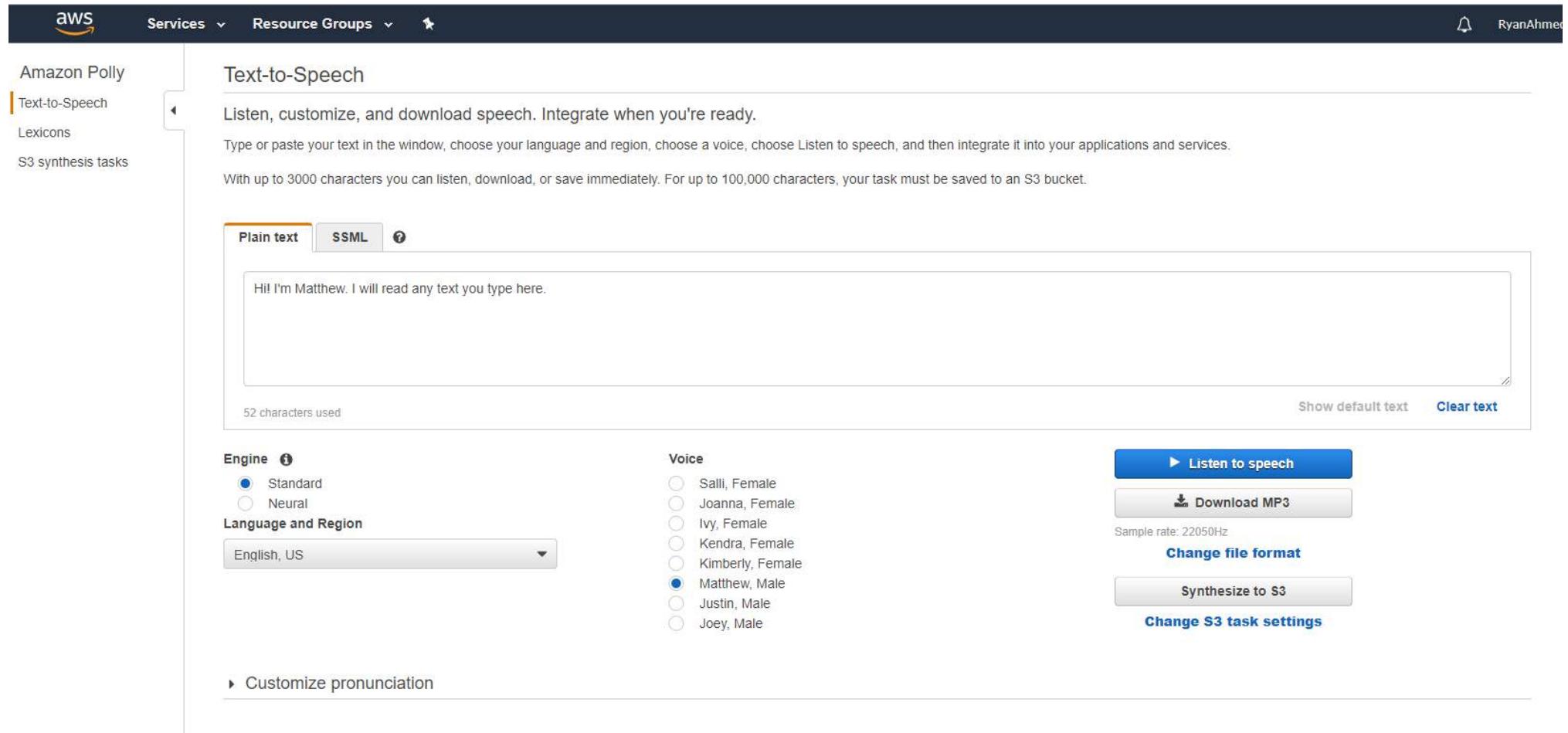
Engine ⓘ
 Standard
 Neural

Language and Region
English, US

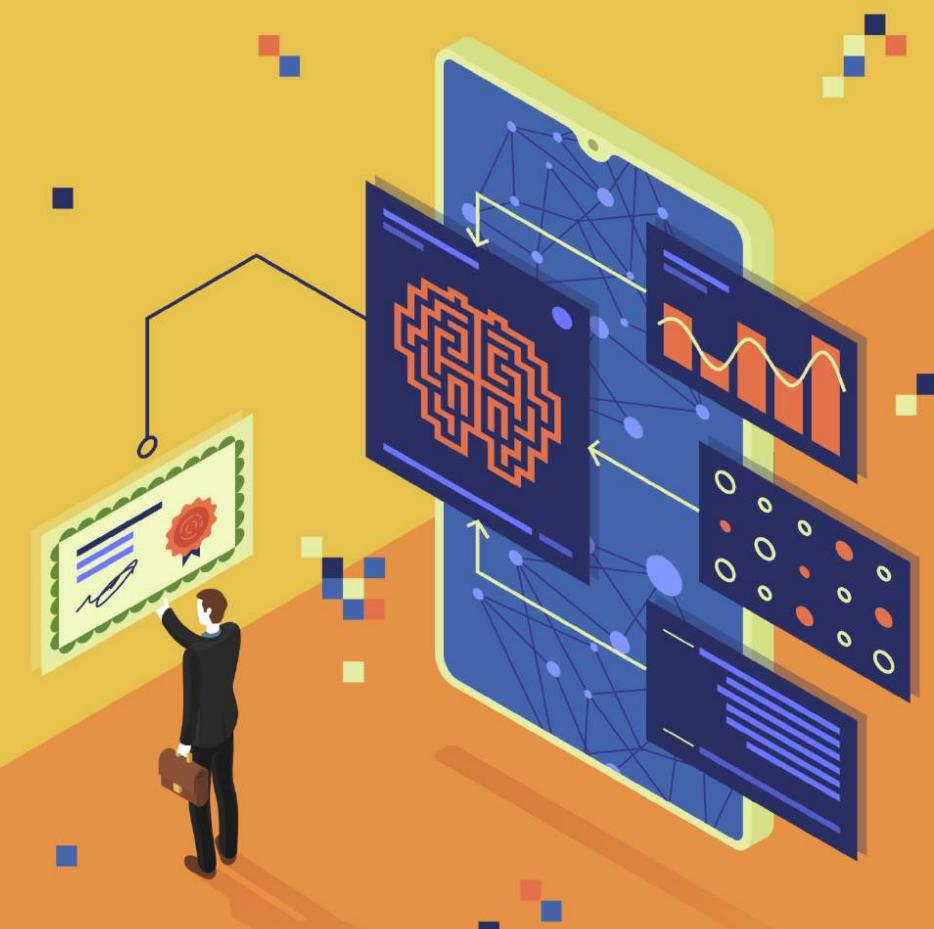
Voice
 Salli, Female
 Joanna, Female
 Ivy, Female
 Kendra, Female
 Kimberly, Female
 Matthew, Male
 Justin, Male
 Joey, Male

▶ Listen to speech
Download MP3
Sample rate: 22050Hz
Change file format
Synthesize to S3
Change S3 task settings

Customize pronunciation



AMAZON FORECAST

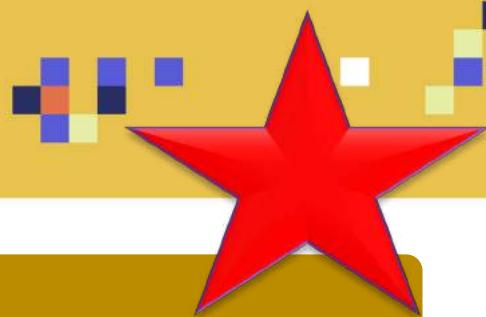


AMAZON FORECAST: OVERVIEW

- Amazon Forecast is used to generate accurate future predictions such as future business sales, demand, resources needed and revenue.
- Making these accurate predictions is critical for companies who want to manage future resources and make informed decisions.
- Amazon Forecast requires zero machine learning experience to use.
- Amazon Forecast requires no servers to provision so it's fully managed service.
- Amazon forecast is based on the same technology that Amazon uses to make predictions. It could combine several time series data to build accurate ML forecast model.
- Amazon Forecast works by looking at historical time series data to make predictions.
- Unlike vanilla prediction tools that make forecasts by simply looking at historical sales data, Amazon forecast is capable of combining other factors such as product features, price changes, discounts made, number of employees.



AMAZON FORECAST: FEATURES



FORECAST GENERATE 50% HIGHER ACCURACY

- Amazon Forecast is extremely accurate, offering up to 50% more accurate results by relying on machine learning.

MUCH MORE EFFICIENT

- Amazon Forecast can be used to generate accurate predictions in hours instead of month.
- Users need to simply move time series data from S3 to Amazon Forecast.
- Amazon Forecast then automatically performs analysis on the data, identifies key features, train and optimizes the model.
- Trained model could then be hosted for inference to generate forecasts.

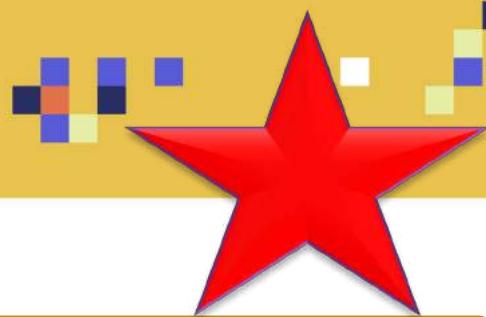
APPLIED TO ANY BUSINESS

- Amazon Forecast could be applied to any business to perform cash flow predictions, resource planning, and product demand.
- It could be applied in retail, logistics, and finance industries.

HIGH SECURITY

- Forecast encrypts your data using customer keys through Key Management Service.
- Data is encrypted at rest.
- Access to Amazon Forecast is also controlled using AWS Identity and Access Management (IAM).

AMAZON FORECAST: USE CASES



PRODUCT DEMAND PLANNING

- Amazon Forecast could be used to forecast inventory levels.
- By feeding in the algorithm with historical of sales, promotions and outlet locations along with weather, website traffic, the algorithm will train a model to generate accurate product demand forecasts.
- Doing so will empower companies to properly stock inventory in various store locations in anticipation of forecasted demand.

FINANCIAL PLANNING

- Amazon Forecast could be used to accurately predict company's financial information such as revenue, sales, expenses and cash flow.

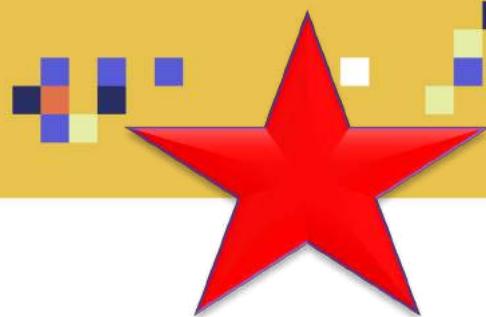
RESOURCE PLANNING

- Amazon forecast could be used to make predictions related to number of employees, raw materials, advertising, and revenue.

AMAZON LEX



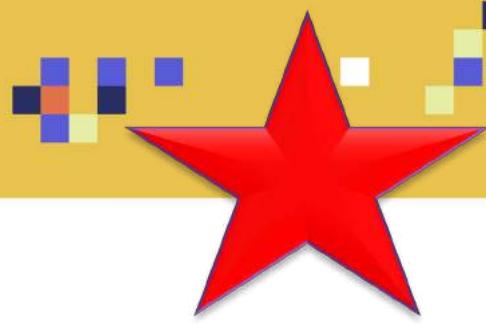
AMAZON LEX: OVERVIEW



- Amazon Lex is Amazon Alexa for anyone!
- Amazon Lex democratizes deep learning and allow any developer to harness the power of deep learning.
- Amazon Lex is a fully managed service so no need for servers provisioning.
- Amazon Lex relies on deep learning to create chatbots (voice and text).
- Amazon Lex is capable of converting speech to text and then understand the intent based on the that text.
- It combines two Technologies:
 - **Automatic speech recognition (ASR)**: converts speech to text
 - **Natural Language Understanding (NLU)**: understand the intent from text



AMAZON LEX: USE CASE



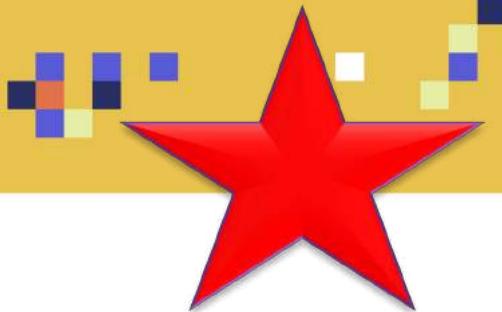
Call Center Chat Bots:

- Amazon Lex could be used to create a chatbot in call centers
- Amazon Lex understands customers intent and change passwords and schedule appointments.
- Human speech analysis is done at 8 kHz audio sampling rate.
- Amazon Lex uses AWS Lambda functions to query business applications and provide data back to customers.



[Photo Credit: https://pixabay.com/illustrations/chatbot-bot-assistant-support-icon-4071274/](https://pixabay.com/illustrations/chatbot-bot-assistant-support-icon-4071274/)

AMAZON LEX: USE CASE UNDER THE HOOD



"Customer makes phone call to schedule an appointment"



AMAZON CONNECT

"Amazon Connect audio to Lex"



AMAZON LEX

"Understands the intent of the customer and send request to Lambda function"



AWS LAMBDA

"Lambda calls database to find customer information"



CUSTOMER SCHEDULING SOFTWARE

"Lambda summons the customer scheduling software"



"Confirmation message sent to customer"



"Customer receives confirmation that the appointment has been scheduled"

**"UNFORTUNATELY ANOTHER JOB
REPLACED BY AI"**



Photo Credit: <http://www.freestockphotos.biz/stockphoto/15684>

<https://www.goodfreephotos.com/vector-images/blue-calendar-vector-clipart.png.php>

<https://www.needpix.com/photo/1027210/customer-support-service-help-communication-contact-operator-telephone-person>

<https://www.pexels.com/photo/man-having-a-phone-call-in-front-of-a-laptop-859264/>

AMAZON PERSONALIZE



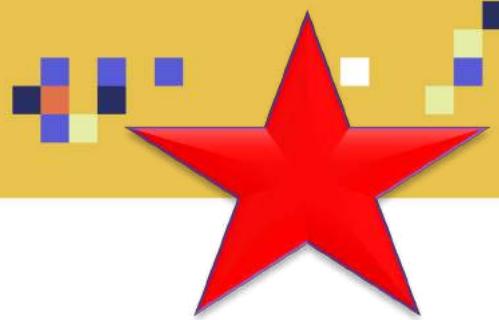
AMAZON PERSONALIZE: OVERVIEW



- Amazon Personalize is used to generate personalized recommendation to customers.
- Companies can use Amazon personalize to send product recommendations and launch a personalized targeted marketing campaigns.
- No machine learning is required!
- Amazon personalize works as follows:
 - Provide Amazon Personalize with customer activity stream online such as click streams and previous purchases.
 - Also provide Amazon personalize with inventory details.
 - Amazon personalize analyzes all the information and develop an optimized personalization model.



AMAZON PERSONALIZE: FEATURES



MAKE ACCURATE RECOMMENDATION

- Amazon personalize works well with hard problems such as new customers (no historical data), changing customer tastes and popularity biases.

REALTIME

- Amazon Personalize combines historical personalized information about each customer with real-time user activity data to make right product recommendations fast.

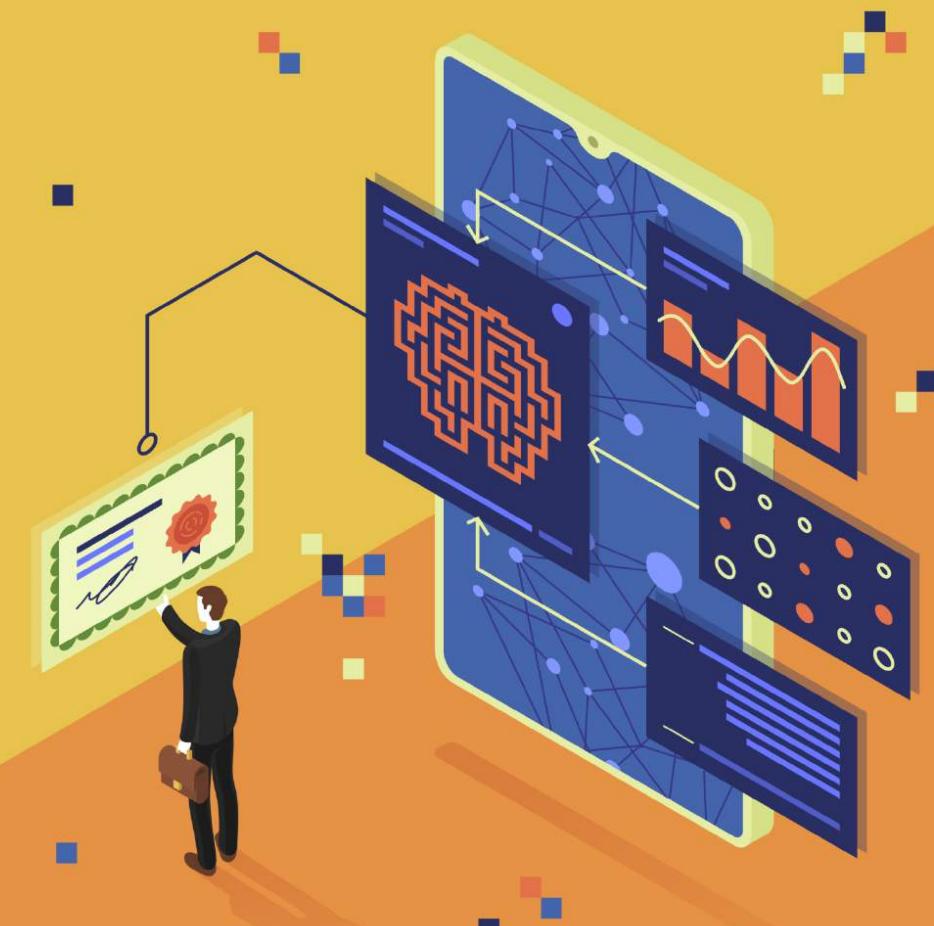
PERSONALIZE EVERYTHING!

- Amazon Personalize can be effectively and easily integrated with websites and mobile apps with a simple API call.

FAST DEPLOYMENT

- With Amazon Personalize requires zero machine learning experience so personalization can be achieved in hours not months!

AMAZON Textract



AMAZON TEXTRACT: OVERVIEW

- Amazon Textract could extract text and data from scanned documents.
- Textract can tell if certain characters are date of birth, SIN number and person's marital status for example.
- Textract is much more powerful compared to basic optical character recognition (OCR).
- Even if field locations are changing from one forum to another, Textract is able to extract this information.
- No machine learning experience is required.
- Textract could be used to create intelligent automated workflow as follows:
 - Textract processes millions of documents in couple of hours and extract information.
 - Using other AWS services, automated approvals could be made.



Photo Credit: <https://publicdomainvectors.org/en/free-clipart/Reading-girl/72485.html>

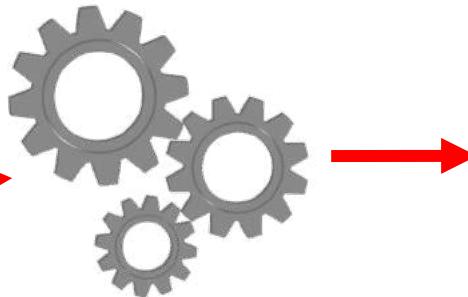
AMAZON TEXTRACT: OVERVIEW



1. Introduction		
Deep learning has dramatically improved the state-of-the-art in many different artificial intelligent tasks like object detection, speech recognition, machine translation (LeCun et al., 2015). Its deep architecture nature grants deep learning the possibility of solving many complex problems (Bengio et al., 2009). As a result, recent years have seen its application to a variety of different modern domains and tasks in addition to traditional tasks like object detection, face recognition, or language models, for example, Osako et al. (2015) uses the recurrent neural network to decode speech signals, Gupta et al. (2015) uses stacked autoencoders to generate images from text, and Wang et al. (2016) uses a generative adversarial model to generate images with different styles. Wang et al. (2016) uses deep learning to allow sentiment analysis from multiple modalities simultaneously, etc. This period is the era to witness the blossoming of deep learning research.		
However, to understand what has developed in the history and why current models exist in present forms, this paper summarizes the evolutionary history of several different deep learning models and explains the main ideas behind these models and their relationship to the ancestors. To understand the main idea in training in deep learning has moved over a long time period, as shown in Table 1. Therefore, this section also offers the readers a wide-thorough of the major milestones of deep learning research. We will cover the milestones as shown in Table 1, as well as many additional works. We will split the story into several sections to discuss the evolution of present models.		
This section is the chronological framework on the human brain modeling. Although the success of deep learning nowadays is not necessarily due to its resemblance of the human brain (more due to its deep architecture), the ambition to build a system that simulates brain indeed thrust the initial development of neural networks. Therefore, the next section begins with the history of neural networks, and ends the age when shallow neural networks were dominant.		
With the maturity of neural networks, the next continues to briefly discuss the necessity of extending shallow neural networks into deeper ones, as well as the previous deep neural networks and the challenges deep architecture introduces.		
With the establishment of the deep neural network, this paper diverges into three main forms of deep learning: open source, in-house, and in-house. In this section, we will see how Deep Belief Net and its construction component Restricted Boltzmann Machine evolve as a trade-off of modeling power and computation loads. In Section 5, this paper focuses on the development history of Convolutional Neural Networks, furnished with the prominent steps along with their corresponding papers. In Section 6, this section discusses the development of Recurrent Neural Networks, its encompasses the LSTM, attention models and the encoders they achieved.		
While this paper primarily discusses deep learning models, optimization of deep architecture is an inevitable topic in the society. Section 7 is devoted to a brief summary of optimization techniques, including advanced gradient method, Dropout, Batch Normalization, etc.		
This paper could be read as a complementary of (Schmidhuber, 2015). Schmidhuber's paper is aimed to assign credit to all those who contributed to the present state of the art, so his paper focuses on every single incremental work along the path; therefore cannot elaborate the details of each work.		
2.		
On the Origin of Deep Learning		

Table 1: Major milestones that will be covered in this paper		
Year	Contributor	Contribution
300 BC	Aristotle	Initiated the Averchianism, started the history of human attempt to understand brain.
1873	Alexander Bain	introduced Neural Groupings as the earliest model of neural network, inspired Hebbian Learning Rule.
1943	McCulloch & Pitts	introduced McCulloch-Pitts model, which is considered as the father of Artificial Neural Models.
1949	Donald Hebb	considered as the father of neural networks, introduced Hebbian Learning Rule, which lays the foundation of modern neural networks.
1958	Frank Rosenblatt	introduced the first perceptron, which highly resembles modern perception.
1974	Paul Werbos	introduced Backpropagation
1980	Toru Kohonen	introduced Self Organizing Map
1983	Kanade & Fukushima	introduced Neocognitron, which inspired Convolutional Neural Network
1982	John Hopfield	introduced Hopfield Network
1985	Hinton & Seidenfeld	introduced Boltzmann Machine
1986	Paul Smolensky	introduced Parallel Distributed Processing, which is later known as Restricted Boltzmann Machine
1986	Michael I. Jordan	defined and introduced Recurrent Neural Network
1990	Yann LeCun	introduced LeNet, showed the possibility of deep neural networks in image recognition.
1997	Schuster & Paliwal	introduced Bidirectional Recurrent Neural Network
2000	Geoffrey Hinton	introduced LSTM, solved the problem of vanishing gradient in recurrent neural networks
2006	Geoffrey Hinton	introduced Deep Belief Networks, also introduced learning rate decreasing technique, opened current deep learning era.
2008	Sakakibara & Hinton	introduced Deep Boltzmann Machine
2012	Geoffrey Hinton	introduced Dropout, an efficient way of training neural networks

AMAZON TEXTRACT

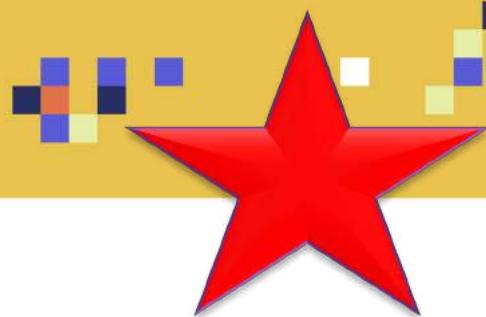


Year	Contributor	Contribution
1990	Yann LeCun	Introduced LeNet
2012	Geoffrey Hinton	Dropout

AWS DEEPLENS



AWS DEEPLENS: OVERVIEW



- AWS DeepLens is a deep learning enabled programmable video camera.
- You can use DeepLens to experiment with deep learning models that have been trained in SageMaker.
- It could support multiple frameworks such as TensorFlow, and Caffe.
- AWS DeepLens comes with preinstalled inference engine using Apache MXNet.
- You can integrate DeepLens with Amazon Rekognition to perform advanced image analysis along with Polly to develop speech-related projects.
- Check out this fun project!
 - <https://aws.amazon.com/deeplens/community-projects/Doorman/>
 - <https://www.youtube.com/watch?v=UXVD22jDbu8>

AWS DEEPLENS: COMMUNITY PROJECTS!



- Check out DeepLens community:
<https://aws.amazon.com/deeplens/community-projects/>

The screenshot shows the 'Community Projects' section of the AWS DeepLens website. At the top, there's a navigation bar with links to AWS DeepLens, Overview, Community Projects (which is the active tab), Resources, and FAQs. Below the navigation is a large banner featuring a camera module mounted on a stand. The main heading 'Community projects' is displayed over a blurred background image. A descriptive paragraph explains the purpose of the page: 'Explore the collection of AWS DeepLens projects contributed by the community of developers who participated in the AWS DeepLens Virtual Hackathon. These projects cover a range of categories, from safety and education to health and wellness and of course, pets and animals. You'll find a short video from the developers demonstrating their innovation and in most cases a more detailed description and link to their GitHub repo. Check them out and no doubt you'll be inspired!'. Below this, there are two calls to action: 'If you have a project you have created with AWS DeepLens that you would like to share on this page you can [submit the outline here](#) for us to take a look at.' and 'Still don't have an AWS DeepLens? You can [buy now](#).'. The next section, 'Meet the AWS DeepLens Hackathon Winners', displays three projects: 'First Place' (Read To Me), 'Second Place' (Dee), and 'Third Place' (SafeHaven). Each project has a thumbnail image, a title, and a brief description. Below these are four smaller project cards: 'ASLens', 'AMAZON DEEPLENS CHALLENGE FACE DETECTION: AWS RECOGNITION, CISCO CMX', 'DeepLens Camera', and 'DeepLens Camera'.

Community projects

Explore the collection of AWS DeepLens projects contributed by the community of developers who participated in the AWS DeepLens Virtual Hackathon. These projects cover a range of categories, from safety and education to health and wellness and of course, pets and animals. You'll find a short video from the developers demonstrating their innovation and in most cases a more detailed description and link to their GitHub repo. Check them out and no doubt you'll be inspired!

If you have a project you have created with AWS DeepLens that you would like to share on this page you can [submit the outline here](#) for us to take a look at.

Still don't have an AWS DeepLens? You can [buy now](#).

Meet the AWS DeepLens Hackathon Winners

First Place

Read To Me

This is a Deep Learning enabled application which is able to read books to kids.

Second Place

Dee

A fun, interactive device for children that asks them to answer questions by showing the right things.

Third Place

SafeHaven

Peace of mind for vulnerable people with Alexa Visitor ID. Supportive families receive doorstep photo alerts via SMS.

ASLens

AMAZON DEEPLENS CHALLENGE FACE DETECTION: AWS RECOGNITION, CISCO CMX

DeepLens Camera

DeepLens Camera

AWS DEEPRACER



AWS DEEPRACER: OVERVIEW

- AWS DeepRacer allows anyone to experiment with reinforcement learning in a fun way!
- Reinforcement learning is a machine learning technique in which agents try to maximize cumulative reward by exploring their environment.
- Reinforcement learning is super powerful and allows for training without the need of expensive and time consuming labeled dataset.
- AWS DeepRacer offers the following:
 - **Simulator:** Use AWS DeepRacer 3D racing simulator to assess trained models in SageMaker.
 - **Car:** Deploy trained models on AWS DeepRacer, tons of fun!
 - **League:** Compete in a global competition.



Photo Credit: <https://www.maxpixels.net/Speedway-Racing-Nascar-Auto-Racing-Car-Sport-558070>

AWS DEEPRACER: FEATURES



LOTS OF FUN!

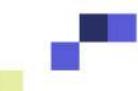
- Learn machine learning in a practical and fun way using DeepRacer.

EXPERIMENTATION

- AWS DeepRacer Simulator allows for running multiple experiments on single and dual cars (head to head).
- Apply Reinforcement learning and other artificial neural networks knowledge in practice!

COMPETE!

- The AWS DeepRacer League allow companies to meet best talent.



AWS MACHINE LEARNING CERTIFICATION



DOMAIN #4: MACHINE- LEARNING IMPLEMENTATION AND OPERATIONS (20% EXAM).



AWS ML CERTIFICATION EXAM DOMAINS



Domain	% of Examination
Domain 1: Data Engineering	20%
Domain 2: Exploratory Data Analysis	24%
Domain 3: Modeling	36%
Domain 4: Machine Learning Implementation and Operations	20%
TOTAL	100%

Source: [https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20\(1\).pdf](https://d1.awsstatic.com/training-and-certification/docs-ml/AWS%20Certified%20Machine%20Learning%20-%20Specialty_Exam%20Guide%20(1).pdf)



DOMAIN #1 OVERVIEW:

SECTION #1: INTRODUCTION, DATA/ML LINGO, AWS DATA STORAGE

- What is Machine Learning and Artificial Intelligence?
- What is Amazon Web Services (AWS)?
- Artificial Intelligence and Machine learning Lingo (data types, Labeled vs. unlabeled, sagemaker groundtruth)
- structured vs. unstructured and database vs. data lake vs. data storage
- AWS Data Storage (Redshift, RDS, S3, DynamoDB)

SECTION #2: AMAZON S3

- Amazon S3 in Depth (partitions, tags)
- Amazon S3 Storage Tiers and Lifecycles
- Amazon S3 Encryption and Security
- Amazon S3 Encryption and Security – Part #2 (ACL, CloudWatch, CloudTrail, VPC)
- Additional Notes (Elasticsearch, ElastiCache, and Database vs. data warehouse)

DOMAIN #1 OVERVIEW:



SECTION #3: AWS DATA MIGRATION, GLUE, PIPELINE, STEP AND BATCH

- AWS Glue (crawlers, features, built-in transformations etc)
- AWS Data pipeline
- AWS Data Migration Service (DMS)
- AWS Batch
- Step Function

SECTION #4: DATA STREAMING & KINESIS

- Kinesis Overview
- Kinesis Video Streams
- Kinesis Data Streams
- Kinesis Firehose
- Kinesis Analytics and Random Cut Forest



DOMAIN #2 OVERVIEW:

SECTION #5: JUPYTER NOTEBOOKS, SCIKIT LEARN, PYTHON PACKAGES, AND DISTRIBUTIONS

- Introduction
- Jupyter Notebooks and Scikit Learn
- Python Packages (Pandas, Numpy, Matplotlib and Seaborn)
- Distributions (Normal, Standard, Poisson, Bernoulli)
- Time Series

SECTION #6: AMAZON ATHENA, QUICKSIGHT AND ELASTIC MAP REDUCE

- Amazon Athena Features
- Amazon Athena Deep Dive (Security, Cost, and glue integration)
- Amazon QuickSight Features
- Amazon QuickSight (integration with AWS services)
- Amazon QuickSight ML insights and Use Cases
- Elastic Map Reduce (EMR)
- Apache Hadoop with EMR
- Apache Spark with EMR

DOMAIN #2 OVERVIEW:



SECTION #7: FEATURE ENGINEERING

- Introduction to Feature Engineering
- Amazon SageMaker GroundTruth
- Feature Selection
- Scaling
- Imputation
- Outliers
- One Hot Encoding
- Binning
- Log Transformation
- Shuffling, Feature Splitting, Unbalanced Datasets
- Text Feature Engineering overview
- Bag of words, punctuation, and dates (easy ones!)
- Term Frequency Inverse Document Frequency (TF-IDF)
- N-Grams (Unigram vs. Bigram vs. Trigram)
- Orthogonal Sparse Bigram (OSB)
- Cartesian Product Transformation

DOMAIN #3 OVERVIEW:



SECTION #8: MACHINE AND DEEP LEARNING BASICS - PART #1

- Artificial Neural Networks Basics: Single Neuron Model
- Activation Functions
- Multi-Layer Perceptron Model
- How do Artificial Neural Networks Train?
- ANN Parameters Tuning - Learning rate and batch size
- Tensorflow playground
- Gradient Descent and Backpropagation
- Overfitting and Under fitting
- How to overcome overfitting?
- Bias Variance Trade-off
- L1 Regularization
- L2 Regularization

SECTION #9: MACHINE AND DEEP LEARNING BASICS - PART #2

- Artificial Neural Networks Architectures
- Convolutional Neural Networks
- Recurrent Neural Networks
- Vanishing Gradient Problem
- LSTM Networks
- Model Performance Assessment - Confusion Matrix
- Model Performance Assessment - Precision, recall, F1-score
- Model Performance Assessment - ROC, AUC, Heatmap, and RMSE
- K-Fold Cross validation
- Transfer Learning
- Ensemble Learning - Bagging and Boosting



DOMAIN #3 OVERVIEW:



SECTION #10: MACHINE AND DEEP LEARNING IN AWS – BUILT-IN ALGORITHMS PART #1

- AWS SageMaker
- Deep Learning on AWS
- SageMaker Built-in algorithms
- Object Detection
- Image Classification
- Semantic Segmentation
- SageMaker Linear Learner
- Factorization Machines
- XG-Boost
- SageMaker Seq2Seq
- SageMaker DeepAR
- SageMaker Blazing Text

SECTION #11: MACHINE AND DEEP LEARNING IN AWS – BUILT-IN ALGORITHMS PART #2

- Random Cut Forest
- Neural Topic Model
- LDA
- K-Nearest Neighbours (KNN)
- K Means
- Principal Component Analysis (PCA)
- IP insights
- Reinforcement Learning
- Object2Vec
- Automatic Model Tuning
- SageMaker and Spark



DOMAIN #3 OVERVIEW:



SECTION #12: MACHINE AND DEEP LEARNING IN AWS - HIGH LEVEL AI/ML PART #3

- ReKognition
- Amazon Comprehend and Comprehend Medical
- Translate
- Transcribe
- Polly
- Forecast
- Lex
- Personalize
- Textract
- AWS DeepLens
- AWS DeepRacer



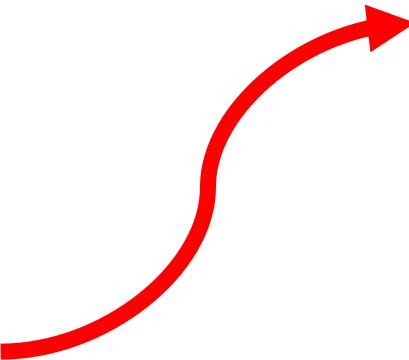
DOMAIN #4: MACHINE LEARNING IMPLEMENTATION AND OPERATIONS



SECTION #13: MACHINE LEARNING IMPLEMENTATION AND OPERATIONS

- SageMaker components overview
- AI/ML Models Deployment
- SageMaker Resources and Instances Management
- Inference Pipeline
- Amazon SageMaker Neo
- Docker Containers
- Model production variants
- Canary Deployment
- Greengrass
- Model Evaluation and validation
- SageMaker Security (IAM, VPC, Encryption)
- SageMaker Security (CloudWatch, CloudTrail)

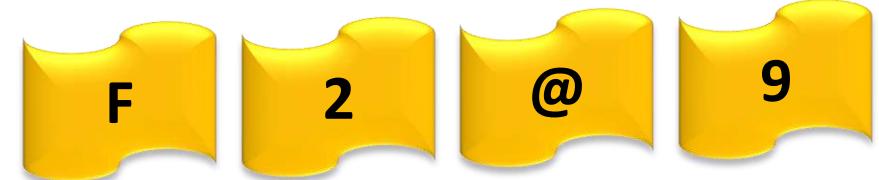
WE ARE HERE!



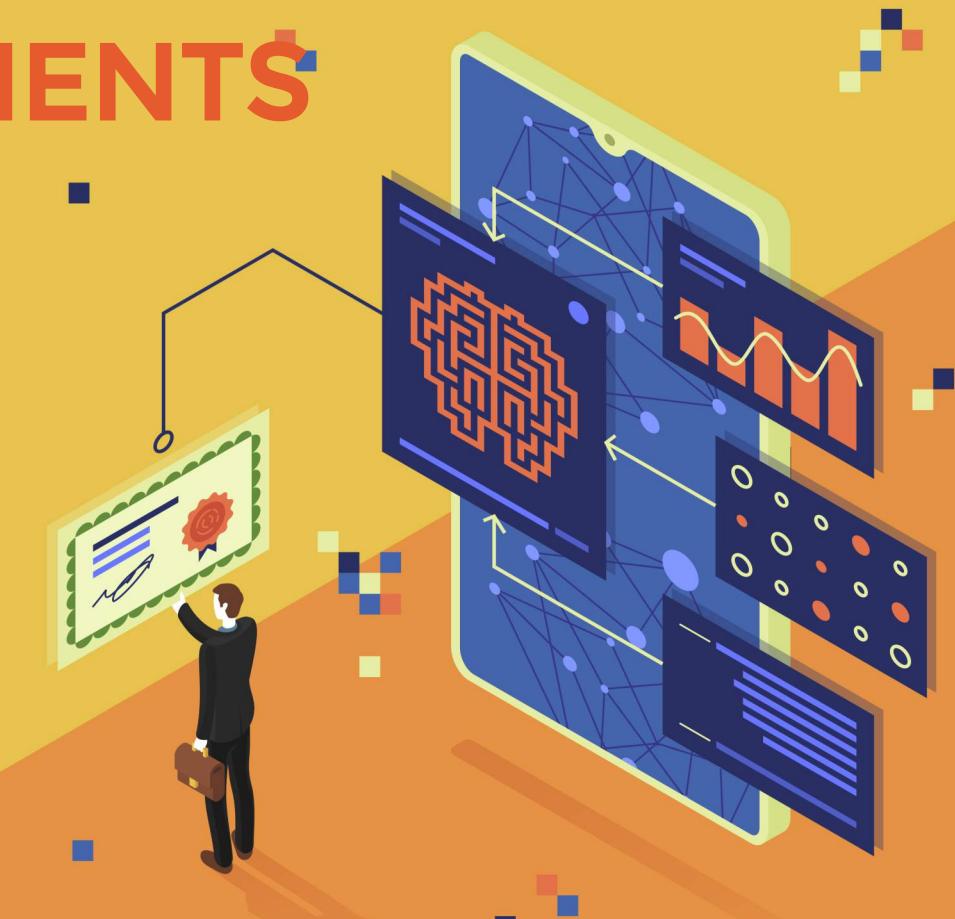
RECALL OUR MINI CHALLENGE AND PRIZE!



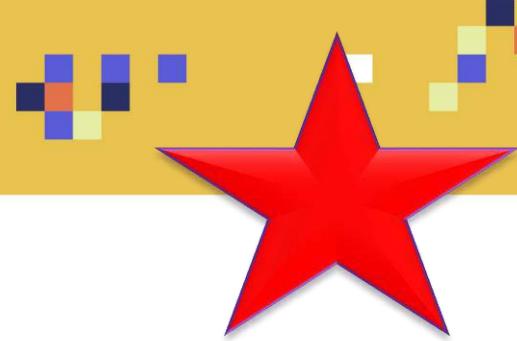
- For those of you who will successfully complete the entire section and watch the videos till the end, they will receive a valuable prize!



SAGEMAKER COMPONENTS REVIEW



AMAZON SAGEMAKER COMPONENTS



- Two components are present in Amazon SageMaker:
 - Model training
 - Model deployment
- To start training an AI/ML model using Amazon SageMaker, you will need to create a training job with the following:
 - **Amazon S3 bucket URL (training data)**: where the training data is located.
 - **Compute resources**: Amazon SageMaker will train the model using instances managed by Amazon SageMaker.
 - **Amazon S3 bucket URL (Output)**: this bucket will host the output from the training.
 - **Amazon Elastic Container Registry path**: where the training code is stored.
- Amazon SageMaker launches an ML compute instances once a training job is initiated.
- Amazon SageMaker uses: (1) training code and (2) training dataset to train the model.
- Amazon SageMaker saves the trained model artifacts in an S3 bucket.



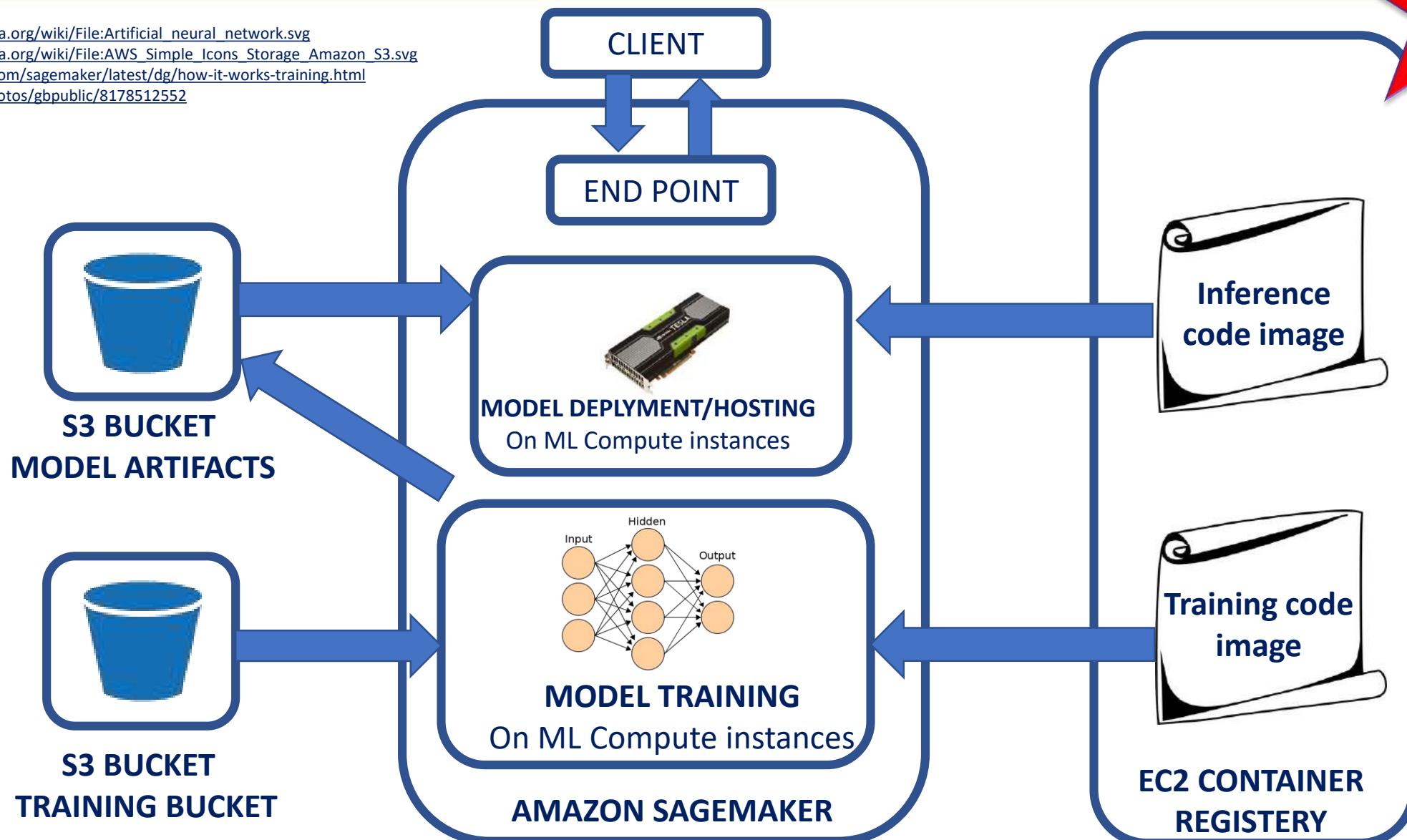
AMAZON SAGEMAKER MODEL TRAINING AND DEPLOYMENT OVERVIEW

Source:

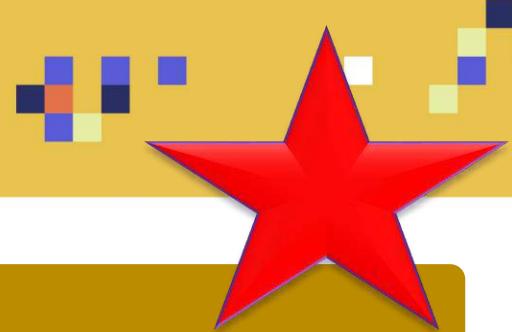
<https://docs.aws.amazon.com/sagemaker/latest/dg/how-it-works-training.html>



https://commons.wikimedia.org/wiki/File:Artificial_neural_network.svg
https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg
<https://docs.aws.amazon.com/sagemaker/latest/dg/how-it-works-training.html>
<https://www.flickr.com/photos/gbpublic/8178512552>



■ TRAINING OPTIONS OFFERED BY SAGEMAKER



USE AN ALGORITHM PROVIDED BY AMAZON SAGEMAKER

- Amazon SageMaker provides ready, off the shelf training algorithms such as: Linear Learner Algorithm and the XGBoost Algorithm, K Means, Principal Component Analysis, image classification, LDA, Sequence to Sequence Algorithm.

USE APACHE SPARK WITH AMAZON SAGEMAKER

- Apache Spark can be used to train models with Amazon SageMaker.

CUSTOM CODE TRAINING USING POPULAR DEEP LEARNING FRAMEWORKS

- custom python code with TensorFlow or Apache MXNet for model training.

USE YOUR OWN CUSTOM ALGORITHMS:

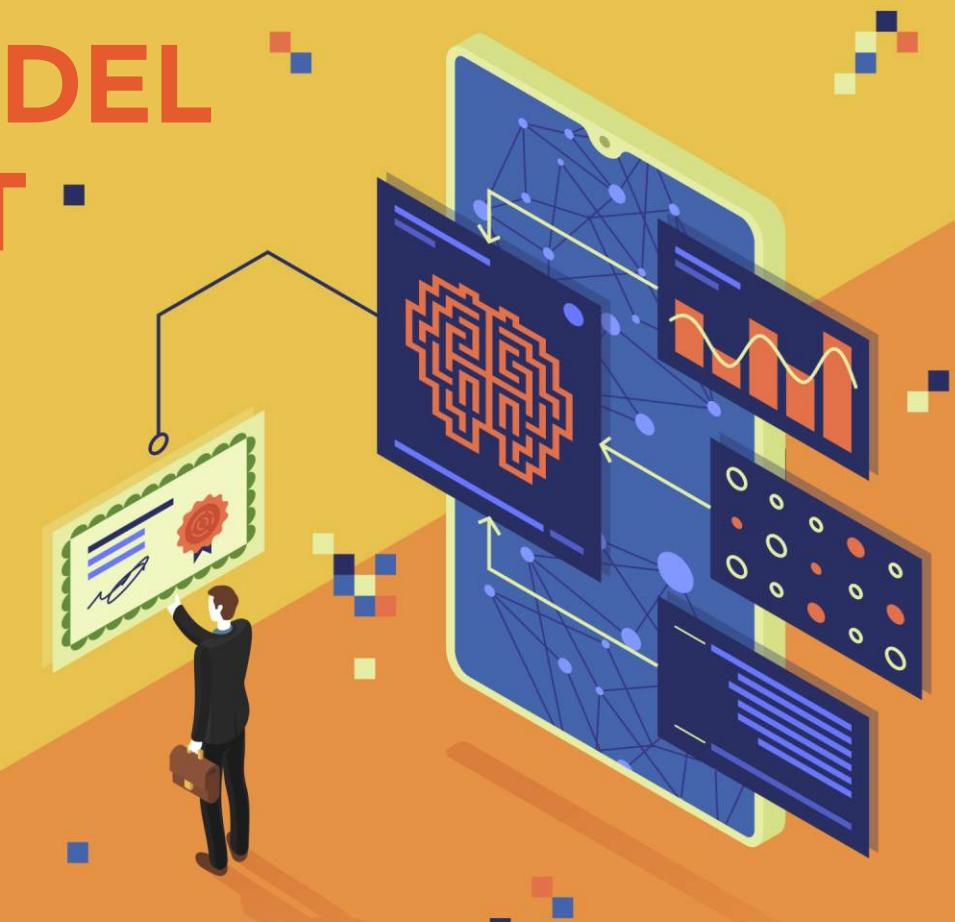
- The code could be placed in a docker container and then a registry path of the image could be provided in an Amazon SageMaker CreateTrainingJob API call.

AWS MARKETPLACE

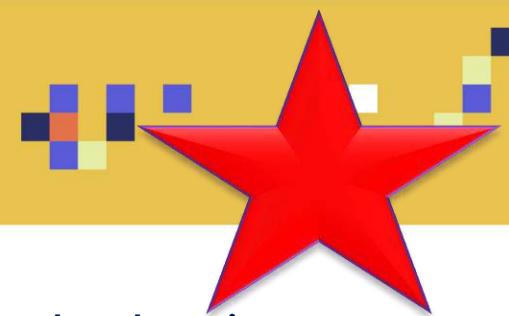
- Choose an algorithm from Amazon marketplace, <https://aws.amazon.com/marketplace/solutions/machine-learning>

Source: <https://docs.aws.amazon.com/sagemaker/latest/dg/how-it-works-training.html>

SAGEMAKER MODEL DEPLOYMENT



MODEL DEPLOYMENT BY SAGEMAKER



- After the AI/ML model is trained, it can be deployed as follows:
 - **For Single Prediction at a time:** set up a persistent endpoint using Amazon SageMaker hosting services.
 - **For Multiple Predictions:** Use Amazon SageMaker batch transform in order to obtain predictions for an entire dataset.
- **SageMaker Inference Pipeline:** SageMaker offers tools to process batch transforms in a pipeline format.
- **Batch Transform:** is used to allow preprocessing of large datasets and performing model inferencing quickly/efficiently without a need to have a persistent endpoint.
- **SageMaker Automatically Scaling:** as the workload changes throughout the day, SageMaker can dynamically adjust the number of instances provisioned so it could save money and compute resources.
- **Amazon SageMaker Elastic Inference (EI):** EI could be used to speed up inference and reduce latency by adding an accelerator without the need of having a dedicated GPU (which will cost much more)
- **Amazon SageMaker Neo:** Used to train TensorFlow, Apache MXNet, PyTorch, ONNX, and XGBoost models once and optimize them for deployment on ARM, Intel, and Nvidia processors. (*Before Neo, you will need to spend major man-hour efforts to deploy AI/ML Models on a specific hardware with specific compiler, memory, operating systems...etc.*)

SAGEMAKER RESOURCES AND INSTANCE TYPES



INSTANCE TYPES SELECTION



- Amazon SageMaker offers a wide variety of instance types to fit many machine learning use cases:
 - Training vs. deployment
 - Deep learning vs. simple regression
- Instance types varies based on: CPU, GPU, memory, and networking capacity
- Resources could be scaled to meet target workload requirements.
- Instance types for deep learning will require a GPU and include the following:
 - P3: 8 Tesla V100 GPU's
 - P2: 16 K80 GPU's
 - G3: 4 M60 GPU's (all Nvidia chips)
- Inference instances are less computationally expensive therefore a compute instance would be sufficient.
 - C4
 - C5

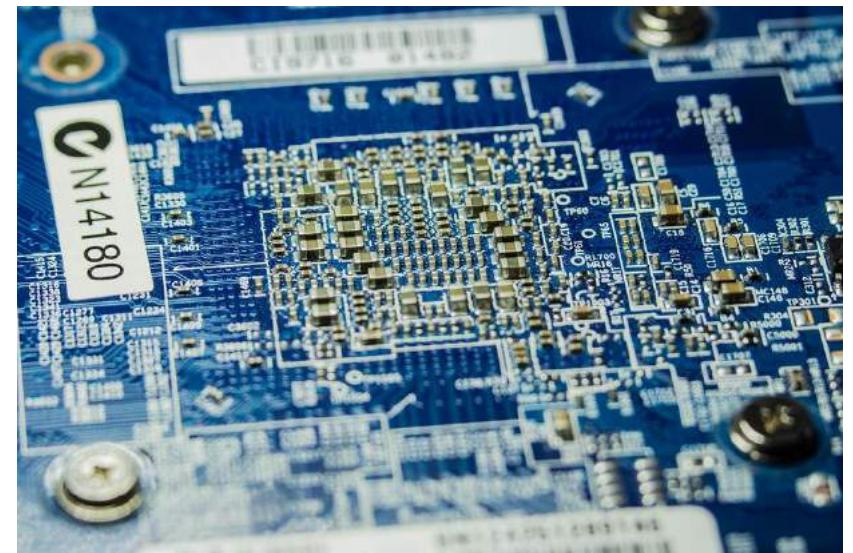


Photo Credit: <https://pixabay.com/photos/transistors-gpu-processor-pc-chip-1137502/>



INSTANCE TYPES SELECTION



- Check this out for a full list of ML instance Types:
<https://aws.amazon.com/sagemaker/pricing/instance-types/>

STANDARD

Instance type	vCPU	GPU	Mem (GiB)	GPU Mem (GiB)	Network Performance
Standard – Current Generation					
mL.t2.medium	2	-	4	-	Low to Moderate
mL.t2.large	2	-	8	-	Low to Moderate

MEMORY OPTIMIZED

Memory Optimized - Current Generation	2	-	16	-	Up to 10 Gbps
mL.r5.large	2	-	16	-	Up to 10 Gbps
mL.r5.xlarge	4	-	32	-	Up to 10 Gbps
mL.r5.2xlarge	8	-	64	-	Up to 10 Gbps
mL.r5.4xlarge	16	-	128	-	Up to 10 Gbps

COMPUTE OPTIMIZED

Compute Optimized - Current Generation	2	4	Up to 10 Gbps
mL.c5.large	2	4	Up to 10 Gbps
mL.c5.xlarge	4	8	Up to 10 Gbps
mL.c5.2xlarge	8	16	Up to 10 Gbps
mL.c5.4xlarge	16	32	Up to 10 Gbps
mL.c5.9xlarge	36	72	10 Gigabit

ACCELERATED COMPUTING

Accelerated Computing - Current Generation	8	1xV100	61	16	Up to 10 Gbps
mL.p3.2xlarge	8	1xV100	61	16	Up to 10 Gbps
mL.p3.8xlarge	32	4xV100	244	64	10 Gigabit
mL.p3.16xlarge	64	8xV100	488	128	25 Gigabit
mL.p3dn.24xlarge	96	8xV100	768	256	100 Gigabit



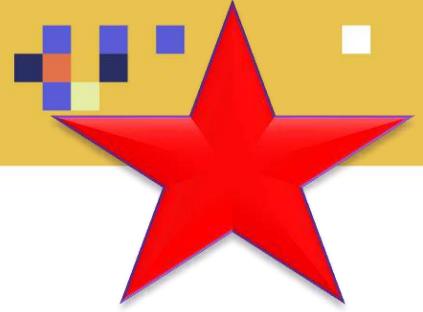
SPOT INSTANCES

- A Spot offers a lower price compared to an on-Demand instance.
- “Spot price” is the price you pay for the spot instance and is adjusted based on availability zone and demand.
- The spot instance will run when:
 - When capacity permits.
 - When the maximum hourly price set is more than the Spot price, the instance will run.
- Choose Spot instances when:
 - You have limited resources
 - When you have some degrees of flexibility. Note that your applications can be interrupted.
 - You want to perform optional tasks, data analysis, and batch jobs.
- Spot instances could save up to 90% compared to on-demand instances



Photo Credit: <https://pxhere.com/en/photo/747848>

AMAZON ELASTIC INFERENCE: OVERVIEW



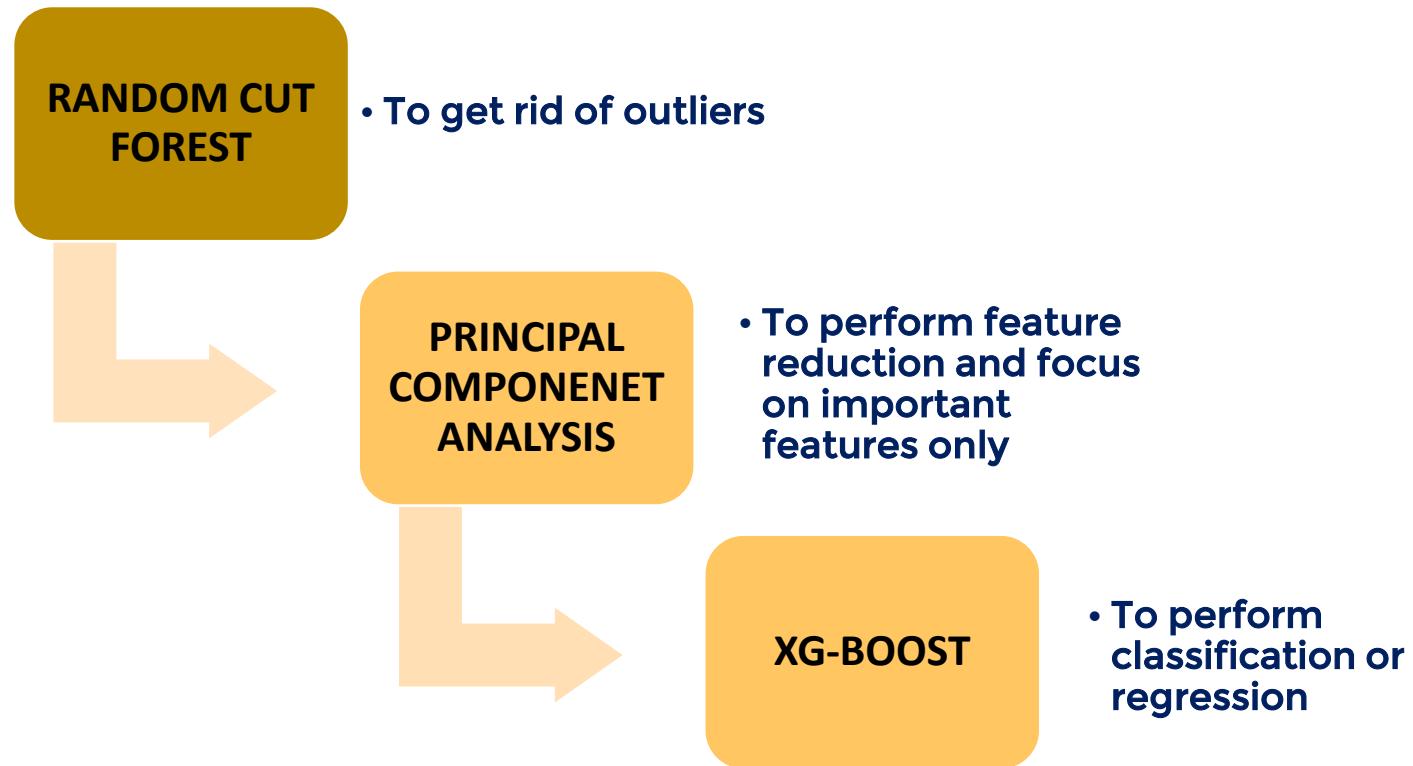
- After the AI/ML models are trained, they are deployed for inference to make predictions.
- Using a standalone dedicated GPU to perform inference is generally an overkill!
- Inference requires a small amount of GPU compute power
- This will result in a very high cost with an underutilized resources.
- Amazon Elastic Inference is designed to help reduce deep learning inference cost by 75%.
- Amazon Elastic inference allows you to attach low-cost GPU-powered acceleration to Amazon EC2 and SageMaker instances with no code changes.
- Amazon Elastic Inference supports TensorFlow, Apache MXNet.



INFERENCE PIPELINE



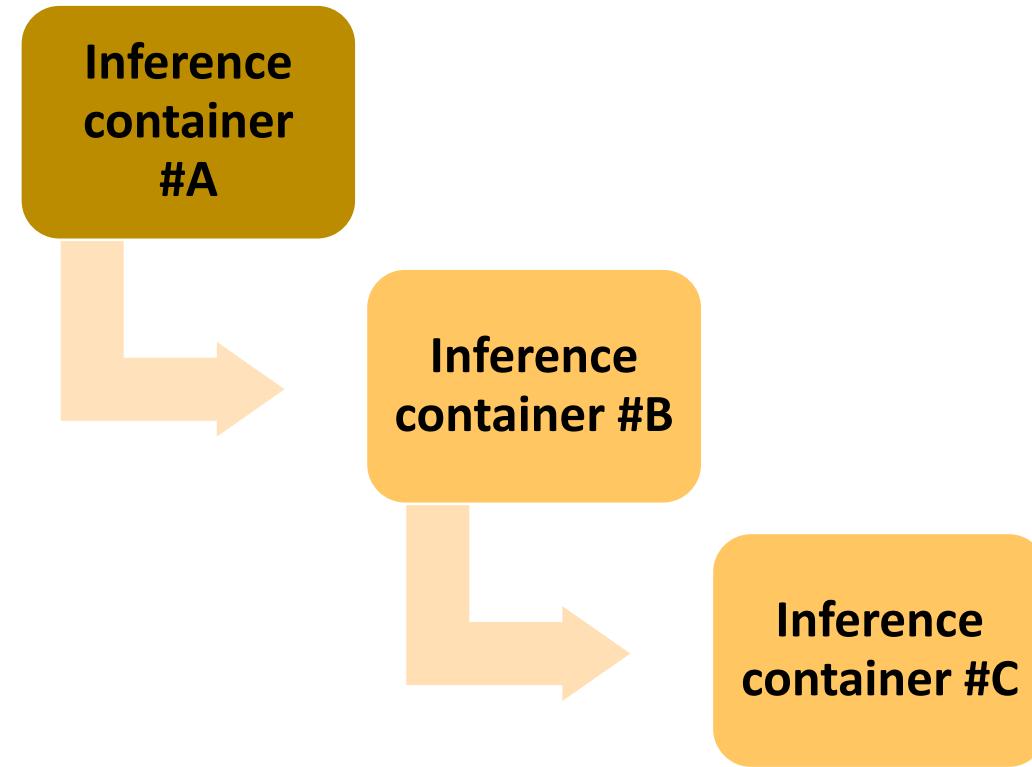
- Sometimes we might need to apply many algorithms in a sequential fashion as follows:



INFERENCE PIPELINE



- SageMaker Model might be composed of a “pipeline” which is a sequence of many containers that process data in a sequential fashion.
- Built-in algorithms or custom-based algorithms in containers could be used to create a Pipeline in Sagemaker



AUTOMATIC SCALING



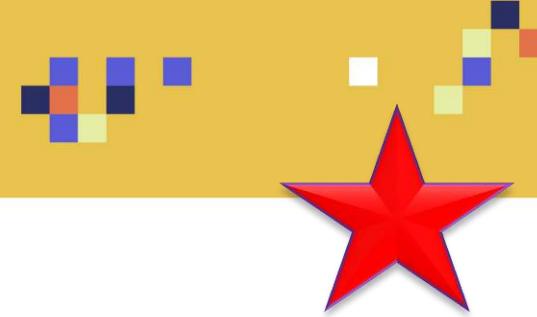
- Amazon SageMaker offers an automatic scaling feature in which the number of instances could be adjusted in response to the workload.
 - If the workload goes up, the number of instances is increased.
 - If workload goes down, the number of instances is reduced.
- SageMaker users have to develop a scaling policy which in turn uses Amazon CloudWatch along with target values assigned by the user.
- Automatic scaling has the following components:
 - Permissions: those are required to perform automatic scaling actions.
 - AWS Identity and Access Management (IAM) role linked to an AWS service.
 - Target metric: CloudWatch metric used by Amazon SageMaker automatic scaling to decide when to scale.
 - Min/Max capacity: min/max number of instances used during scaling.
 - Cool down period: time required after a scale-in or scale-out activity completes before another scale-out activity can start.



MODEL EVALUATION (VALIDATION)



MODEL VALIDATION



- After the model is trained, its performance needs to be assessed.
- This is critical to ensure that the model meets the business goals.
- There are generally two ways of validating the model, online validation and offline validation.

ONLINE VALIDATION (LIVE DATA)

- Validation performed using real world dataset on live traffic
 - Models trained online are called production variants
 - Direct 10% of the traffic to the model under test and 90% to the original model
 - A/B Testing

OFFLINE VALIDATION (HISTORICAL DATA)

- Validation performed using historical data (not live traffic)
 - Training data = 75% entire dataset
 - Validation data = 25% entire dataset

OFFLINE VALIDATION OPTIONS



- There are generally two types of offline validation: holdout set and k-fold.

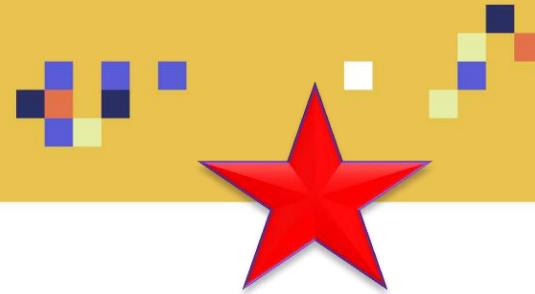
HOLDOUT SET

- Holdout set is a portion of the dataset (20%-30%) that is not used during model training.
 - The holdout set is used to assess the performance of the model using data that the model has never seen before during training.
 - Good trained model should be able to “generalize” and not “memorize”.

K-FOLD

- Dataset is split into k parts (k ranges from 5 to 10).
 - You treat each of these parts as a holdout set for k training runs, and use the other k-1 parts as the training set for that run.
 - k trained models are generated
- Comparing these models performance is used to assess the final model (generalization capability).

K-FOLD CROSS VALIDATION



- Cross-validation is used to assess the generalization capability of the model (with data that has never been seen before).
- K-fold cross validation is a technique used to assess the performance of machine learning models.
- K-Fold works by randomly dividing the data into equal sized K groups (folds).
- First fold is considered the validation dataset, and the model is trained on the remaining $k - 1$ folds.
- By comparing the performance of the model with these many folds, you can now know if the model is over fitted or not.
- If the error is comparable across various runs then the model is generalized.

Model1: Trained on Fold1 + Fold2 and Tested on Fold3

Model2: Trained on Fold2 + Fold3 and Tested on Fold1

Model3: Trained on Fold1 + Fold3 and Tested on Fold2

MODEL VALIDATION: XGBOOST METRICS



Metrics Computed by the XGBoost Algorithm

The XGBoost algorithm computes the following nine metrics during training. When tuning the model, choose one of these metrics as the objective to evaluate the model.

Metric Name	Description	Optimization Direction
validation:accuracy	Classification rate, calculated as #(right)/#(all cases).	Maximize
validation:auc	Area under the curve.	Maximize
validation:error	Binary classification error rate, calculated as #(wrong cases)/#(all cases).	Minimize
validation:f1	Indicator of classification accuracy, calculated as the harmonic mean of precision and recall.	Maximize
validation:logloss	Negative log-likelihood.	Minimize
validation:mae	Mean absolute error.	Minimize
validation:map	Mean average precision.	Maximize
validation:merror	Multiclass classification error rate, calculated as #(wrong cases)/#(all cases).	Minimize
validation:mlogloss	Negative log-likelihood for multiclass classification.	Minimize
validation:mse	Mean squared error.	Minimize
validation:ndcg	Normalized Discounted Cumulative Gain.	Maximize
validation:rmse	Root mean square error.	Minimize

<https://docs.aws.amazon.com/sagemaker/latest/dg/xgboost-tuning.html>

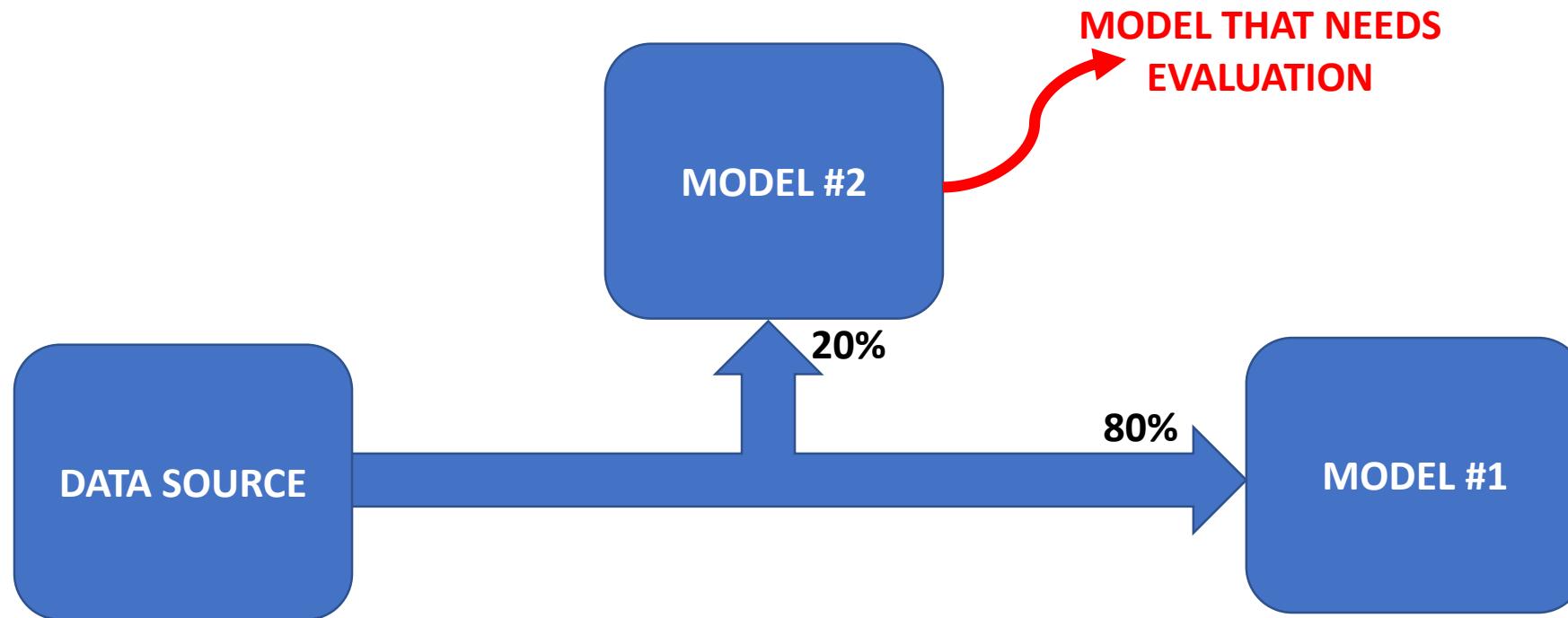
PRODUCTION VARIANTS



■ PRODUCTION VARIANTS/CANARY DEPLOYMENT



- Production variants are used to test out several models on incoming live traffic.
- If many models are deployed, variants weights can tell Amazon SageMaker how to distribute traffic among the models.
- A subset of the live traffic (let's say 20%) could be directed to the model that requires validation.
- After the model is evaluated and its performance is assessed, 100% of the traffic could be directed to it.



■ PRODUCTION VARIANTS: INITIAL VARIANT WEIGHT



InitialVariantWeight:

- It specifies the traffic distribution among the models specified in an endpoint.
- Traffic is calculated as follows:

Traffic (%)

$$= \frac{\text{VariantWeight}}{\text{Sum of all Variant weight values across production variants}}$$

ENDPOINT CONFIGURATION

PRODUCTION VARIANT #1 (MODEL #1)

Initialvariantweight = 0.1

PRODUCTION VARIANT #2 (MODEL #2)

Initialvariantweight = 0.9

CANARY DEPLOYMENT



CANARY DEPLOYMENT



- Canary deployment is used to minimize the risk of deploying a new AI/ML model.
- Instead of directing the entire traffic to the new model, an incremental approach is taken.
- 10% of the traffic is directed to the new model and the remaining 90% is directed to the old (trusted) model.
- As we have more confidence in the new model, we will start to replace the current (old) model completely.
- You have to include a parameter that indicate which model is currently running and reporting metrics.

AMAZON SAGEMAKER NEO



AMAZON SAGEMAKER NEO: OVERVIEW

- Amazon SageMaker Neo solves one of the most difficult challenges which is how to deploy and tune machine learning models to run on a specific target hardware.
- Optimizing machine learning models to run on a specific hardware is an extremely difficult task since it requires Expert knowledge of:
 - Hardware architecture
 - Instruction set
 - Memory



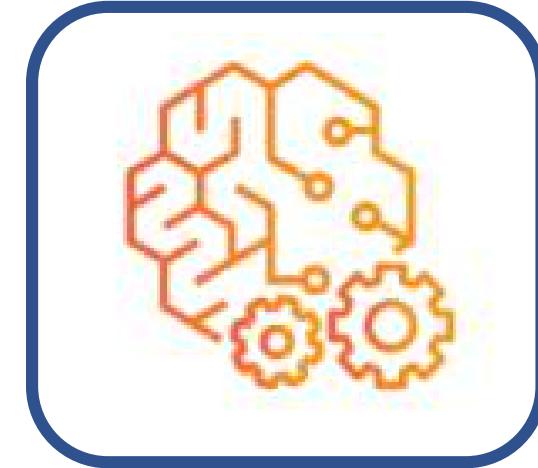
SAGEMAKER NEO

NEO OPTIMIZES TRAINED MODEL TO A SPECIFIC TARGET HARDWARE

AMAZON SAGEMAKER NEO: OVERVIEW



- Amazon SageMaker Neo allows for Training the machine learning models once and running trained machine learning models anywhere on the edge or in the cloud.
- Edge devices generally have limited memory and compute requirements.
- Therefore, Neo will optimize models to run extremely fast on the selected edge device so reduce latency.
- SageMaker Neo consists of a compiler and a runtime.



SAGEMAKER NEO

NEO OPTIMIZES TRAINED MODEL TO A SPECIFIC TARGET HARDWARE



AMAZON SAGEMAKER NEO: APPLICATION EXAMPLE



- For autonomous vehicles as an example, two challenges exist:
 - Many hardware platforms exist so building and optimizing ML models to run on a specific target hardware is challenging and time consuming.
 - Latency, sensors and compute need to react quickly with minimum latency.
- Amazon Neo is here to solve these problems!
- Neo optimized ML models so it could run much faster with minimum latency.



Photo Credit: <https://www.flickr.com/photos/smoothgroover22/15104006386>

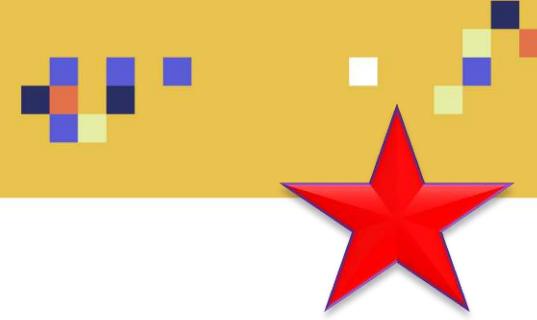


AMAZON SAGEMAKER NEO: STEPS



- Steps:
 - Train a machine learning model using SageMaker and with your favorite framework: MXNet, TensorFlow, PyTorch, or XGBoost.
 - Select target hardware platform from Intel, NVIDIA, or ARM.
 - Run SageMaker Neo and it will compile the model and generate an executable. Neo achieves massive gains in performance by using an artificial neural network that will implement hardware-specific performance optimizations.
 - Deploy trained model in cloud or edge.
- Neo Greengrass can allow for compute and ML inference capabilities to the edge.
- AWS Greengrass allows for direct model deployment to the edge with over the air updates.

AMAZON SAGEMAKER NEO: HOW IT WORKS?



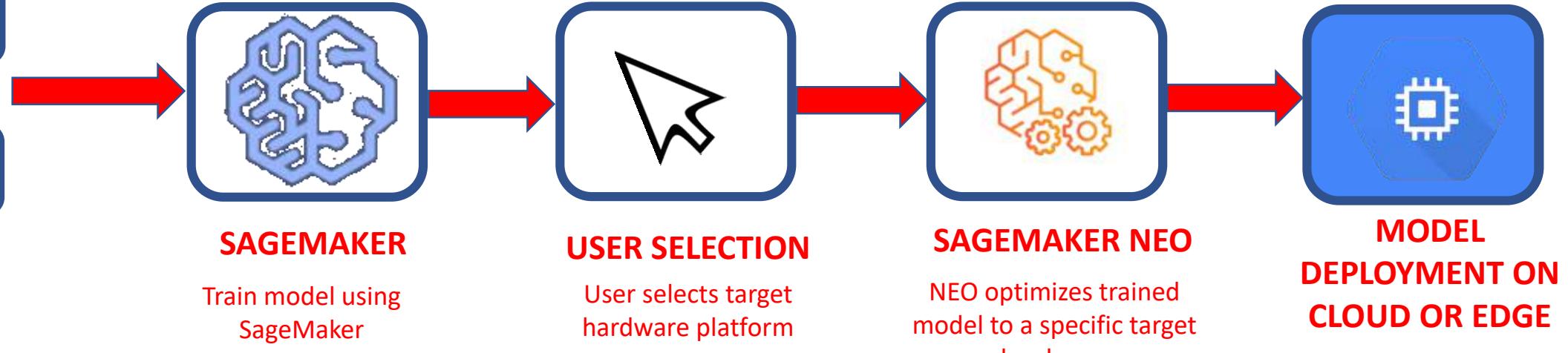
Source: <https://aws.amazon.com/sagemaker/neo/>

TENSORFLOW

PYTORCH

APACHE
MXNET

XGBOOST

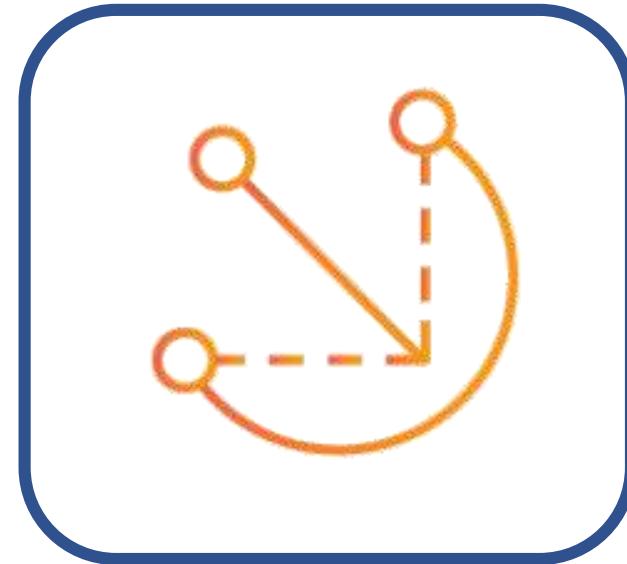


AWS GREENGRASS



AWS GREENGRASS: OVERVIEW

- AWS Greengrass is an extremely powerful tool that enables developers to build IoT devices and application logic.
- AWS IoT Greengrass is a software that allows local devices to have cloud capabilities.
- AWS IoT Greengrass allows for data collection and secure communication with several local devices on the network.
- Local devices can use AWS IoT Greengrass to communicate securely and send data to the AWS cloud.
- AWS IoT Greengrass developers rely on AWS Lambda functions to create serverless applications.
- Watch Video: <https://aws.amazon.com/greengrass/>

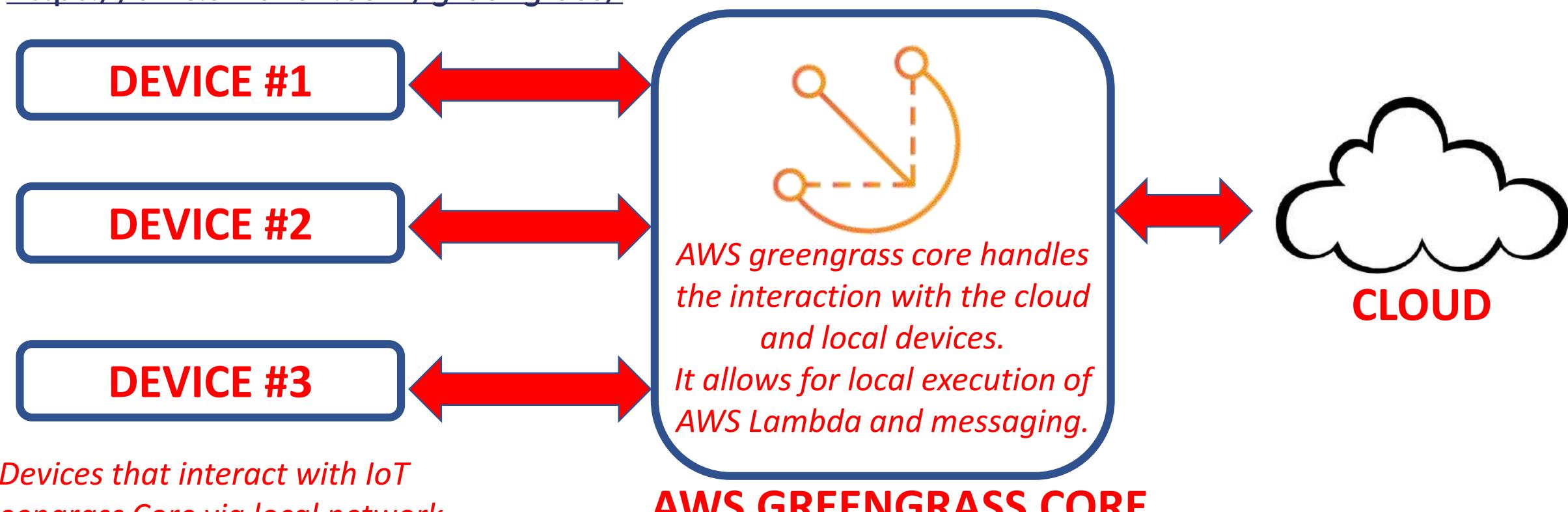


AWS GREENGRASS

AWS GREENGRASS: HOW DOES IT WORK?



- AWS IoT Greengrass allows for easy development of IoT solutions that connect many devices to each other and to the cloud.
- Devices could be microcontroller or appliances.
- AWS IoT Greengrass Core works as a hub to communicate.
- <https://aws.amazon.com/greengrass/>



SAGEMAKER AND DOCKER CONTAINERS



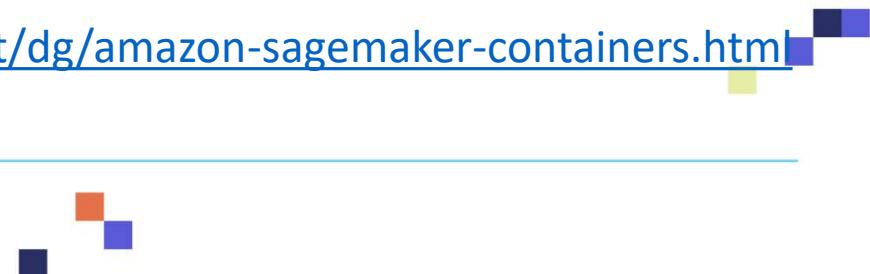
WHAT IS A CONTAINER?



- A container packages up code along with all its dependencies so that it could reliably run across many compute environments.
- Containers are so powerful since they isolate software from its environment.
- A docker container image contains everything needed to successfully run an applications such as:
 - Code
 - System tools
 - Libraries
 - Runtime
 - Settings

SageMaker Developer Guide: <https://gmoein.github.io/files/Amazon%20SageMaker.pdf>

Resource: <https://docs.aws.amazon.com/sagemaker/latest/dg/amazon-sagemaker-containers.html>



SAGEMAKER CONTAINERS



- Models in SageMaker are hosted in Docker containers
 - Pre-built Models in scikit Learn and Spark ML
 - Pre-built Models in Tensorflow, MXNet, and PyTorch
 - Custom algorithm
- Amazon SageMaker containers is a library that enables developers to:
 - Create containers to run scripts
 - Train AI/ML models
 - Deploy models
- How to install Amazon SageMaker Containers library?
 - You can run this command from the Dockerfile

RUN pip install sagemaker-containers



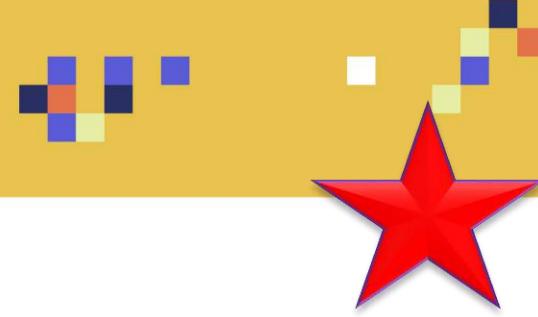
SAGEMAKER CONTAINERS



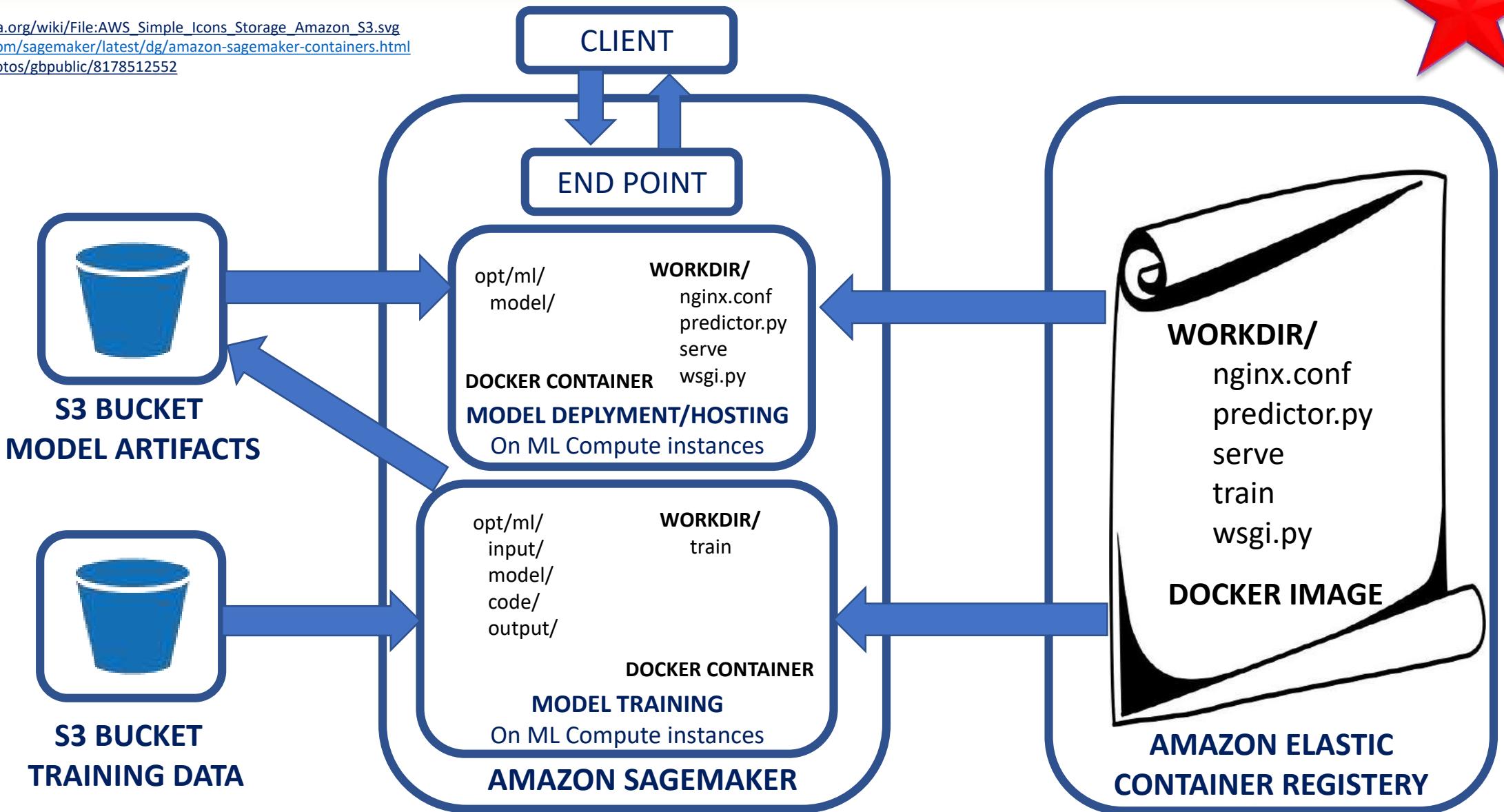
- The library specifies where to store the code and other resources for both training and inference.
- The Dockerfile copies the code to be run into the location expected by an Amazon SageMaker-compatible container.
- When the container is initiated, the Dockerfile indicates the entry point containing the code to run.
- After a Docker image is built, it can be pushed to the Amazon Elastic Container Registry (ECR).



AMAZON SAGEMAKER CONTAINERS SCHEMATIC



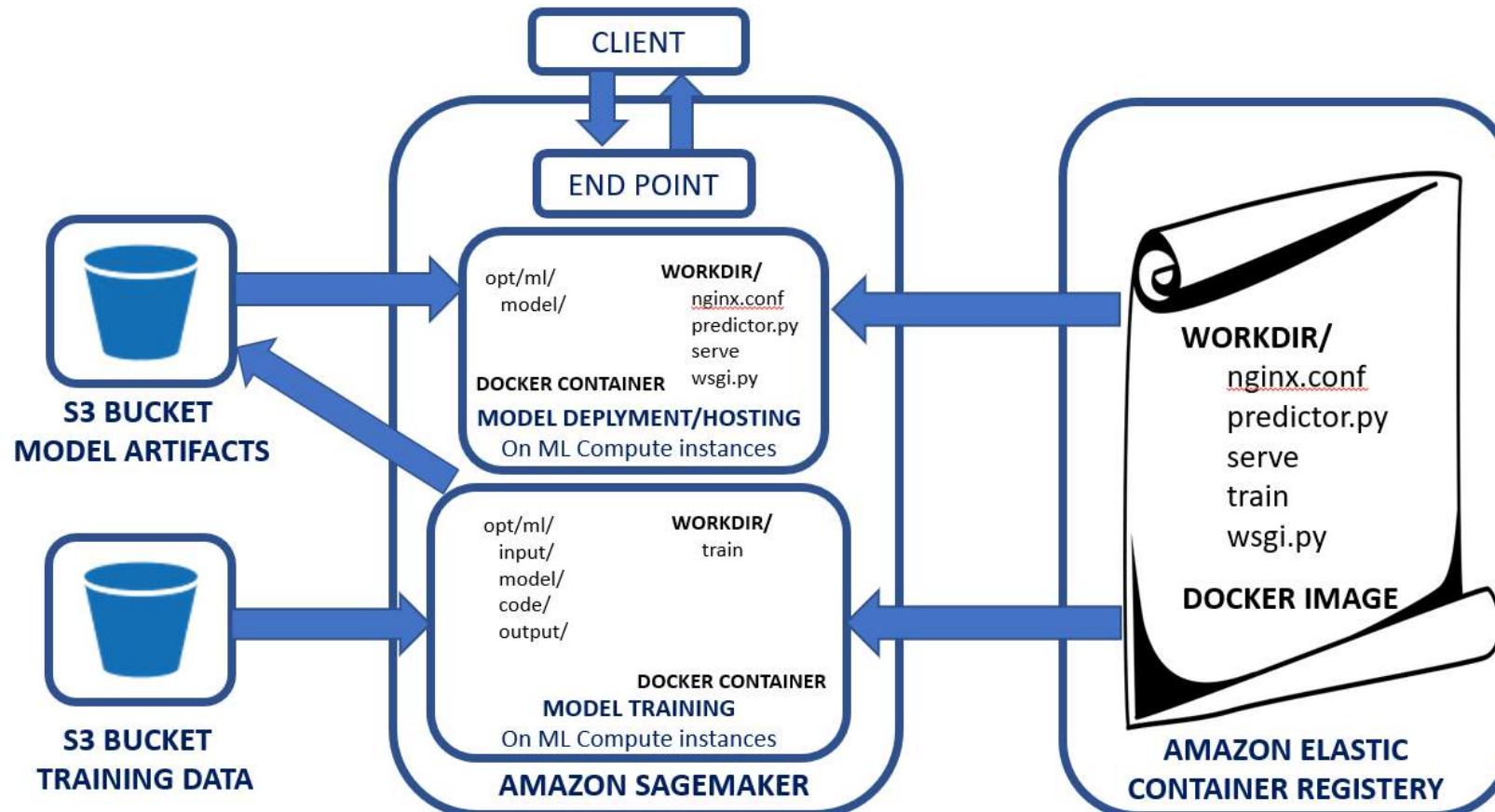
https://commons.wikimedia.org/wiki/File:AWS_Simple_Icons_Storage_Amazon_S3.svg
<https://docs.aws.amazon.com/sagemaker/latest/dg/amazon-sagemaker-containers.html>
<https://www.flickr.com/photos/gbpublic/8178512552>



AMAZON SAGEMAKER CONTAINERS SCHEMATIC



- Docker containers are created from docker images
- Images are built from a Dockerfile
- Images are saved Amazon Elastic Container Registry



SAGEMAKER CONTAINER STRUCTURE DURING TRAINING AND INFERENCE

- The following files are created in the container's /opt/ml directory when Amazon SageMaker trains a model.

DURING TRAINING

```
/opt/ml
├── input
│   ├── config
│   │   ├── hyperparameters.json
│   │   └── resourceConfig.json
│   └── data
│       └── <channel_name>
│           └── <input data>
└── model
└── code
    └── <script files>
└── output
    └── failure
```

DURING INFERENCE

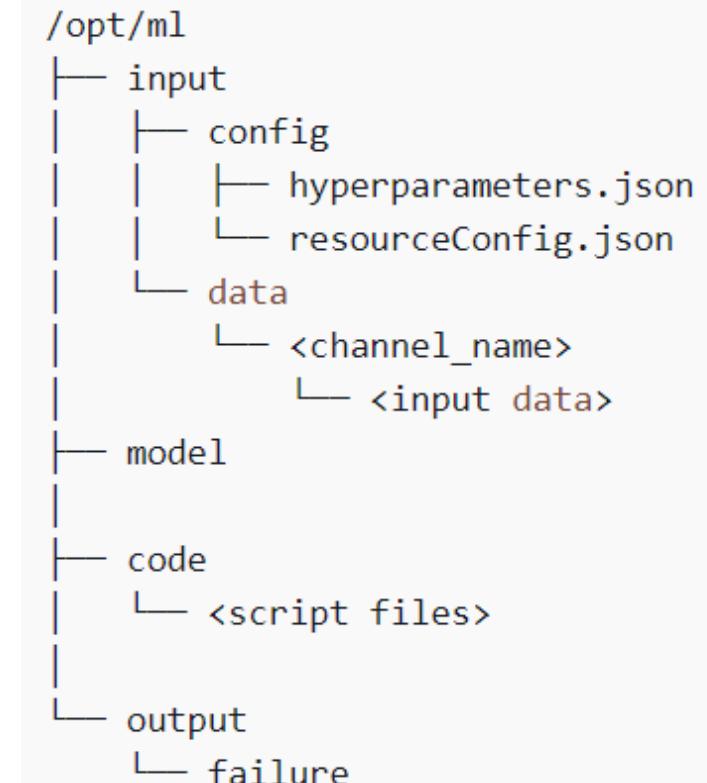
```
/opt/ml
└── model
    └── <model files>
```

Source: <https://docs.aws.amazon.com/sagemaker/latest/dg/amazon-sagemaker-containers.html>

SAGEMAKER CONTAINER STRUCTURE DURING TRAINING

- **/opt/ml/input/**
 - includes JSON files that configure the algorithm's hyperparameters and network architecture.
 - The directory also indicates the channels that SageMaker will use to access the data in S3.
- **/opt/ml/code/**
 - Contains scripts to run.
- **The /opt/ml/model/**
 - Contains model generated by the algorithm.
- **/opt/ml/output/**
 - Includes data such as why a training job has failed.

DURING TRAINING



Source: <https://docs.aws.amazon.com/sagemaker/latest/dg/amazon-sagemaker-containers.html>

SAGEMAKER CONTAINER STRUCTURE DURING HOSTING/DEPLOYMENT/INFERENCE



- In the container, the model files are in the same place that they were written to during training.

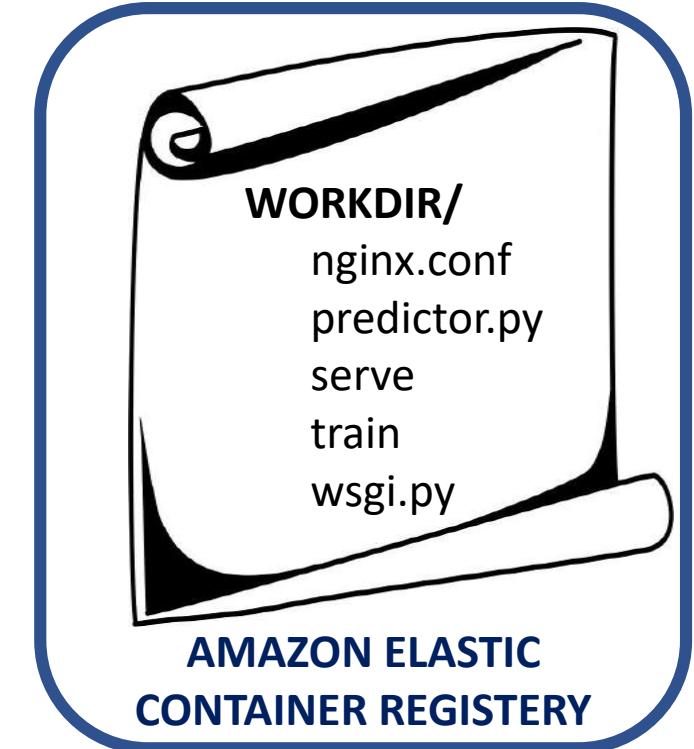
DURING
HOSTING/DEPLOYMENT/
INFERENCE

```
/opt/ml
└── model
    └── <model files>
```

DOCKER IMAGE STRUCTURE



- When amazon sagemaker models are trained, they are deployed to an HTTP endpoint.
- This allows to perform real-time model inference.
- The container contains a serving stack to process requests.
- In SageMaker, the serving stack files are installed in the container's WORKDIR which contains the following files.
 - *nginx.conf: configuration file for nginx front end.*
 - *predictor.py: program that implements Flask web server*
 - *Serve: program that run when container is started for hosting.*
 - *Train: program that is invoked when the container is run during training.*
 - *wsgi.py - A wrapper used to invoke Flask application.*



<https://gmoein.github.io/files/Amazon%20SageMaker.pdf>

AMAZON SAGEMAKER SECURITY



AMAZON SAGEMAKER SECURITY OVERVIEW



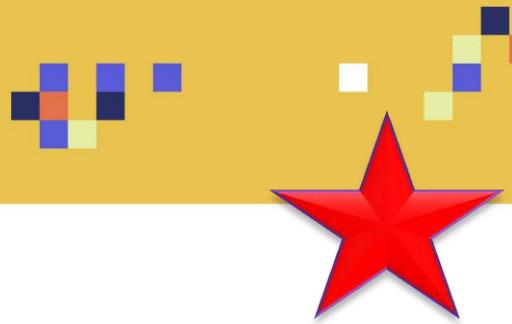
AWS SAGEMAKER SECURITY: OVERVIEW



- Amazon SageMaker ensures the highest level of security to its customers.
- Amazon SageMaker follows a *shared security model* as follows:
 - **Security of the cloud:**
 - ❖ AWS ensures the protection of the infrastructure.
 - ❖ All services offered by AWS are very secure.
 - ❖ Security is being regularly audited by third party to ensure compliance.
 - **Security in the cloud:**
 - ❖ Users of AWS are responsible for their own data sensitivity and organization requirements.



AWS S3 SECURITY: RESOURCE Vs. USER BASED POLICES



- S3 buckets are private by default, only the owner can access the bucket.
- Owner can allows access to others by creating an access policy.
- There are generally two types of polices in AWS to grant permission to resources in AWS S3:
 - **Resource-based:** these are access polices that you attach to your buckets.
 - ❖ Example: bucket polices and Access Control List (ACL)
 - **User-based:** polices assigned to specific users in the account
 - ❖ Example: Identity and Access Management (IAM) policies
- Both polices use JSON-based access policy language.



SAGEMAKER SECURITY



IDENTITY AND ACCESS MANAGEMENT (IAM)

- IAM allows for managing access to AWS services in a secure fashion. This is achieved by creating AWS users/groups and then giving/denying access to AWS resources.

MULTIFACTOR AUTHENTICATION (MFA)

- MFA adds additional level of security by asking users to enter a unique authentication using an MFA mechanism besides the normal sign-in credentials.

DATA ENCRYPTION

- Encryption at rest or in transit can enhance data security.

CLOUDTRAIL

- CloudTrail is used to track activity made by users, roles on AWS services. CloudTrail provides a record of past requests, IP addresses, timing of the requests.

SSL/TLS

- Transport Layer Security and Secure Sockets Layer are cryptographic protocols used to ensure communications security over a network.

VPC

- Amazon Virtual Private Cloud (VPC) allows users to create an AWS resources inside a virtual Network. This means that the traffic will never leave or go through the public internet and will stay inside the VPC for maximum security.

MACIE

- Allows for finding and securing personal data in S3

AMAZON SAGEMAKER ENCRYPTION



AWS SAGEMAKER SECURITY: PROTECTION AT REST USING ENCRYPTION



- **Amazon S3:**
 - S3 buckets for model artifacts and training data can be encrypted
- **AWS Key Management Service (KMS):**
 - Amazon SageMaker notebooks, training jobs, hyperparameter tuning jobs, batch transform jobs, and endpoints are encrypted with KMS Key.



AWS SAGEMAKER SECURITY: PROTECTION IN TRANSIT USING ENCRYPTION



- SageMaker encrypts machine learning model artifacts in transit and at rest.
- Inter-network data in transit are TLS encrypted.
- Requests to SageMaker API and console are made over a secure (SSL) connection.
- AWS IAM roles are assigned to Amazon SageMaker to provide permissions to access resources for both training and deployment.



AWS SAGEMAKER SECURITY: INTER-CONTAINER TRAFFIC ENCRYPTION



- When distributed training is performed, ML algorithms transmit data such as model weights.
- Data could be protected by using inter-container traffic encryption.
- Inter-container traffic encryption is enabled via console or API when a training job is set up
- Inter-container traffic encryption will:
 - Increase cost
 - Increase training time (when deep learning is implemented)

AWS SAGEMAKER SECURITY: PROTECTION USING VPC



- **VPC endpoints increase security by allowing users to access AWS services using AWS network without having to send traffic over the public internet.**
- **Amazon SageMaker runs training jobs in an Amazon Virtual Private Cloud (VPC) by default to ensure data security.**
- **Additional level of security could be added (optionally) to protect training containers and data by configuring a private VPC.**
- **By default, all SageMaker Notebooks are Internet-enabled which might pose a security risk. If disabled, the VPC needs an interface endpoint (PrivateLink) or NAT Gateway, and allow outbound connections, for training and hosting to work.**

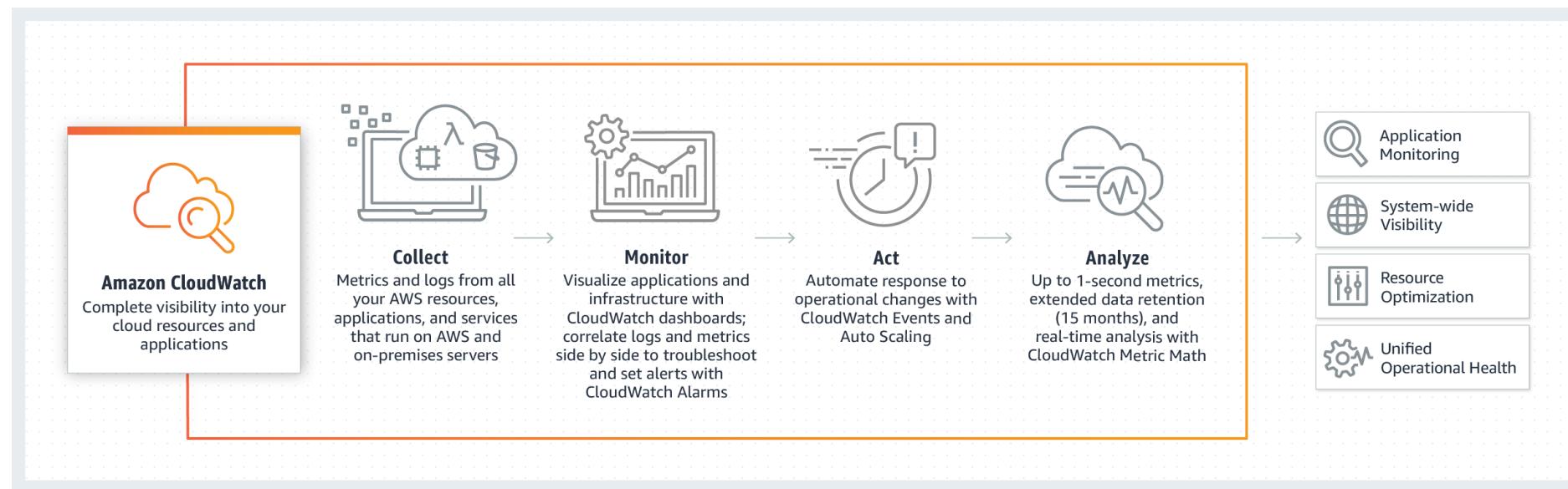
AMAZON CLOUDWATCH



AWS SAGEMAKER SECURITY: CLOUDWATCH



- Amazon CloudWatch collects data such as logs, metrics, and events so users can track their applications and take actions accordingly such as resources optimization.
- CloudWatch could be used to detect anomalies, trigger an alarm, visualize metrics/logs, and automatically take actions.



Source: <https://aws.amazon.com/cloudwatch/>

AWS SAGEMAKER SECURITY: CLOUDWATCH



FULL VISIBILITY AN ALL AWS APPLICATIONS/RESOURCES

- CloudWatch allows for metric and logs collection from all AWS resources which allows for gaining a system-wide visibility.

EASE OF USE

- AWS CloudWatch is super easy to use
- Works with over 70 AWS services such as EC2, S3, Lambda, and DynamoDB.
- Publishes 1-minute metrics and logs (could be enhanced with up to 1-second).

IMPROVED/OPTIMIZED PERFORMANCE

- Amazon CloudWatch allows for alarms to be triggered when a certain threshold is reached.
- Automatic scaling could use CloudWatch to increase or decrease the number of instances and therefore optimize cost.

OBTAIN VALUABLE INSIGHTS

- CloudWatch offers detailed dashboards that show valuable insights.
- Data is retained for 15 months.
- 1-sec granularity.

AWS SAGEMAKER SECURITY: CLOUDWATCH



Screenshot of the AWS CloudWatch Overview page.

CloudWatch Services:

- CloudWatch Dashboards (0)
- CloudWatch Alarms (0)
- CloudWatch ALARM (0)
- CloudWatch INSUFFICIENT (0)
- CloudWatch OK (0)
- CloudWatch Billing
- CloudWatch Logs (Log groups, Insights)
- CloudWatch Metrics
- CloudWatch Events
- CloudWatch Rules
- CloudWatch Event Buses
- CloudWatch ServiceLens (NEW)
- CloudWatch Service Map
- CloudWatch Traces
- CloudWatch Synthetics (NEW)
- CloudWatch Canaries
- CloudWatch Thresholds
- CloudWatch Contributor Insights (NEW)
- CloudWatch Settings (NEW)
- CloudWatch Favorites

Update: Monitor your applications using CloudWatch ServiceLens that correlates metrics, logs, and traces. [Learn more](#)

CloudWatch: Overview (Time range: 1h 3h 12h 1d 3d 1w custom) [Actions](#)

All resources

Get started with CloudWatch

- View alarms**: Set alarms on any of your metrics to receive notifications when your metric crosses your specified threshold.
- View logs**: Monitor using your existing system, application and custom log files.
- View dashboards**: Create re-usable dashboards which allow you to monitor your AWS resources in one location.
- View events**: Write rules to indicate which events are application and what automated actions to take.

Alarms by AWS service

Services	Status	Alarm	Insufficient	OK
AWS services publishing metrics in your account will appear here.				

Learn more about CloudWatch.

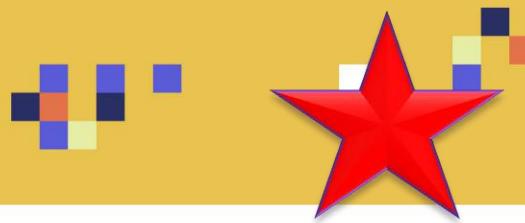
Recent alarms [View recent alarms dashboard](#)

Recent alarms will appear here.
Learn more about CloudWatch Alarms.

AMAZON CLOUDTRAIL



AWS SAGEMAKER SECURITY: CLOUDTRAIL FEATURES



ALWAYS ON

- AWS CloudTrail automatically record the activity on every AWS account.
- Data is available for download from the previous 90 days.

HISTORY

- AWS users are able to search account activity history and improve their security process.

WORKS IN MANY REGIONS

- AWS CloudTrail could be configured to track account history from many regions into one Amazon S3 bucket.

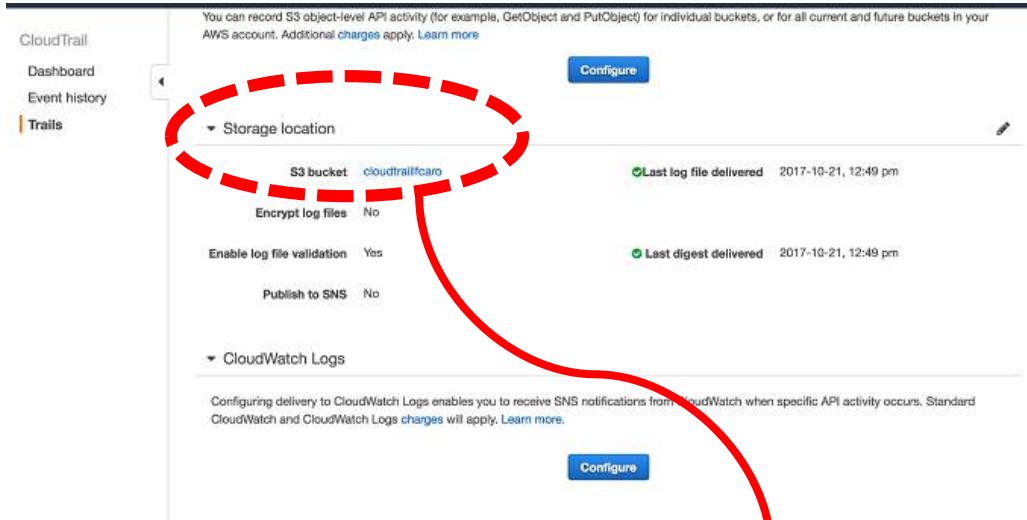
LOG FILE ENCRYPTION

- AWS CloudTrail logs are encrypted by default when they arrive at the Amazon S3 using server-side encryption (SSE).
- AWS Key Management Service (AWS KMS) key could be used for additional security.

QUERY WITH ATHENA

- After the data is available in S3, Athena could be used to query the data and analyze the data

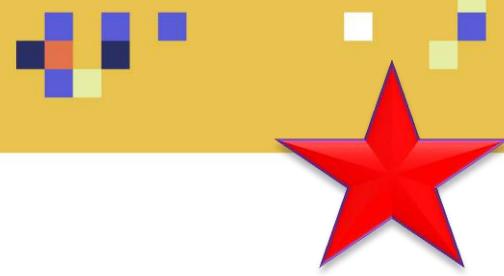
AWS SAGEMAKER SECURITY: CLOUDTRAIL EXAMPLE



Great article: <https://aws.amazon.com/blogs/big-data/streamline-aws-cloudtrail-log-visualization-using-aws-glue-and-amazon-quicksight/>

```
{ "Records": [ { "eventVersion": "1.01", "userIdentity": { "type": "IAMUser", "principalId": "AIDAJDPLRKLG7UEXAMPLE", "arn": "arn:aws:iam::123456789012:user/Alice", "accountId": "123456789012", "accessKeyId": "AKIAIOSFODNN7EXAMPLE", "userName": "Alice", "sessionContext": { "attributes": { "mfaAuthenticated": "false", "creationDate": "2014-03-18T14:29:23Z" } } }, "eventTime": "2014-03-18T14:30:07Z", "eventSource": "cloudtrail.amazonaws.com", "eventName": "StartLogging", "awsRegion": "us-west-2", "sourceIPAddress": "72.21.198.64", "userAgent": "signin.amazonaws.com", "requestParameters": { "name": "Default" }, "responseElements": null, "requestID": "cdc73f9d-aea9-11e3-9d5a-835b769c0d9c", "eventID": "3074414d-c626-42aa-984b-68ff152d6ab7" } ] }
```

AWS SAGEMAKER SECURITY: LOGGING AND MONITORING



AMAZON CLOUDWATCH ALARMS

- Used to send an alarm once a certain threshold is exceeded for multiple number of cycles.
- Could be used to ensure the health of training instances

AWS CLOUDTRAIL LOGS

- CloudTrail is used to track activity made by users, roles, or on AWS service.
- CloudTrail provides a record of past requests, IP addresses, timing of the request...etc.