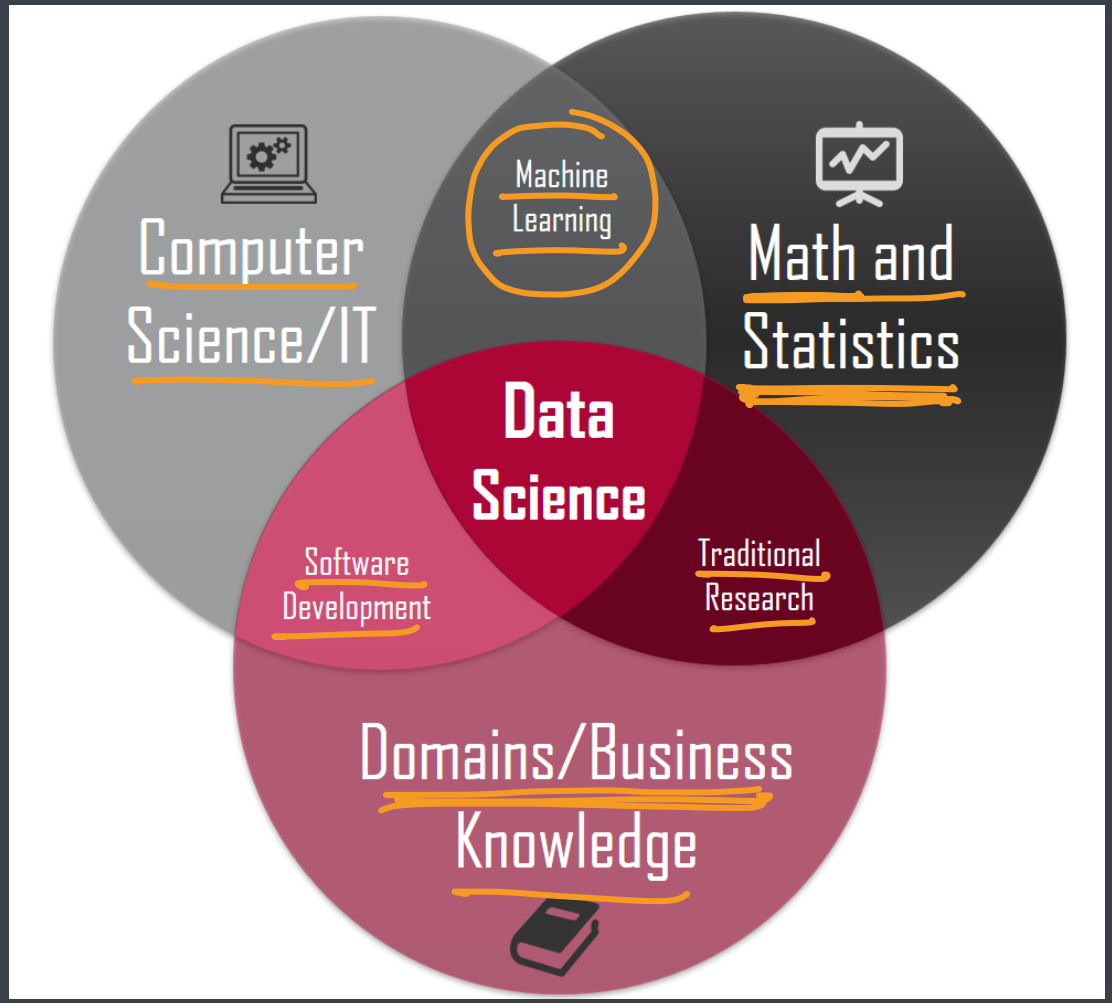
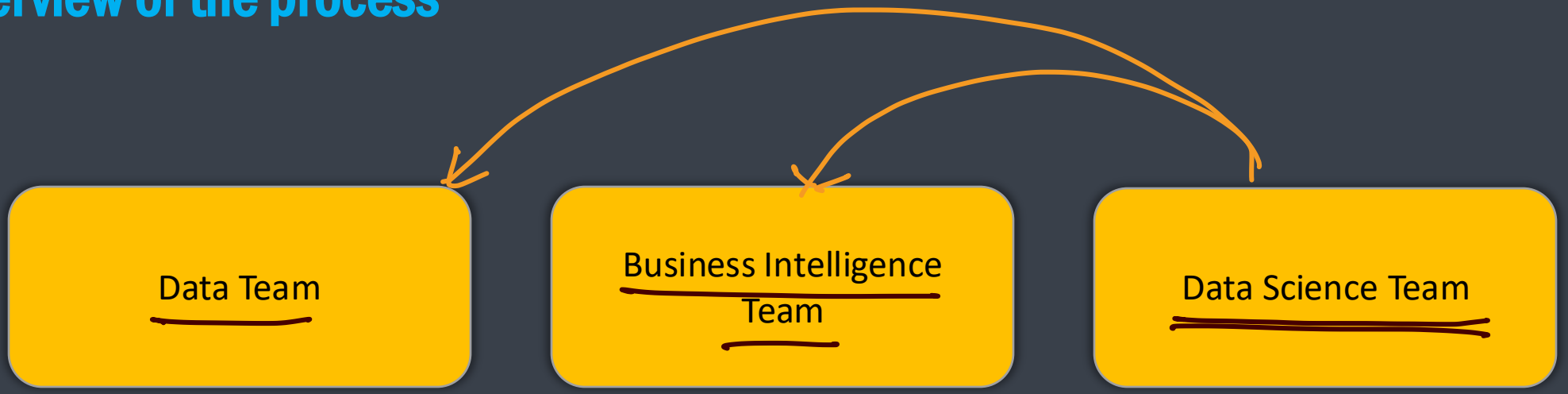
Data Science

What is Data Science ?

- Data science is an inter-disciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from many structural and unstructured data.[1][2] Data science is related to data mining, machine learning and big data
- Data science is a "concept to unify statistics, data analysis and their related methods" in order to "understand and analyze actual phenomena" with data
- It uses techniques and theories drawn from many fields within the context of mathematics, statistics, computer science, domain knowledge and information science.



Quick overview of the process



- Make the data available
- Clean the data
  - plan
  - collect
  - organise

- Find business insights
- Prepares the dashboard

- Use data analytics tools
- Develop models for various tasks
  - formula

Roles

- **Data Scientist**
  - A data scientist is an analytics professional who is responsible for collecting, analyzing and interpreting data to help drive decision-making in an organization
- **Data Architect**
  - A data architect is an IT professional responsible for defining the policies, procedures, models and technologies to be used in collecting, organizing, storing and accessing company information.
- **Data Analyst**
  - A data analyst is responsible for collecting, cleaning, and analyzing data that can be used to improve business decisions
  - They must be able to effectively communicate their findings to those who will make the decisions. Data analysts typically have a strong background in mathematics and computer science
- **BI Analyst**
  - BI analysts determine business-critical priorities and requirements, define KPIs (Key Performance Indicators), implement DW (Data Warehouse) strategies, and identify BI (Business Intelligence) by mining Big Data using advanced software and tools

Roles

- **Data Engineer**
  - A data engineer is responsible for collecting, managing, and converting raw data into information that can be interpreted by data scientists and business analysts
  - Data accessibility is their ultimate goal, which is to enable organizations to utilize data for performance evaluation and optimization

- **AI Engineer**
  - Artificial intelligence (AI) engineers are responsible for developing, programming and training the complex networks of algorithms that make up AI so that they can function like a human brain

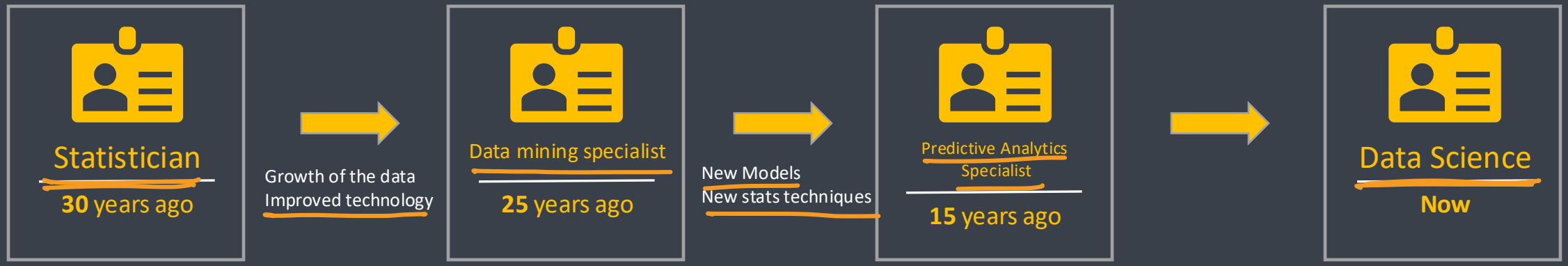
- **Business Analyst**
  - Business analysts are agents of change—professionals who analyze a business or organization, by documenting its systems and processes, assessing its business model, identifying vulnerabilities, and devising solutions
  - Business analysts go by many other job titles, including: Business Architect

Roles

- **Statistician**
  - A statistician is a person who works with theoretical or applied statistics
  - The profession exists in both the private and public sectors
  - It is common to combine statistical knowledge with expertise in other subjects, and statisticians may work as employees or as statistical consultants

- **ML Engineer**
  - A machine learning engineer (ML engineer) is a person in IT who focuses on researching, building and designing self-running artificial intelligence (AI) systems to automate predictive models

How is it evolved ?



Responsible for
- Gathering data
- Cleaning data
- Applying statistical methods
- Analyzing data

Responsible for
- Extracting patterns from the data

Responsible for
- Perform more accurate forecast

Analysis vs Analytics

- Analysis
  - Analysis is performed on the things that are already happened in the past
  - We do Analysis to explain How and or Why something happened
  - E.g.
    - Analyzing data by separating into chunks

- Analytics
  - Refers to the future after finding patterns
  - We use Analytics to explore potential future events
  - Branches
    - Qualitative
      - Intuition and experience
      - Analysis
    - Quantitative
      - Formulas
      - Algorithms

Data Science Techniques

Data Collection

- Raw data → No structure → meaningless
  - Can not be analyzed straight away
  - It is untouched data that is accumulated and stored on the server
  - Also known as raw facts or primary data
  - Can be collected by various techniques like
    - Survey
    - Automated tools

Data Pre-Processing

- This process tries to fix the problem that has occurred while data gathering
- Before processing with data analysis, it is important to remove the wrong data
- Techniques
  - Class Labeling
    - Labeling the data to the correct data types
    - e.g. numeric and categorical
  - Data cleansing
    - Deal with inconsistent data
    - Also known as data cleaning or data scrubbing
  - Dealing with missing values
  - Data balancing
  - Data shuffling
    - Prevents unwanted patterns
    - Improves predictive performance

footy = 90

Analyzing Data → human task

- Once the data is cleaned and formatted, it can be analyzed for various reasons
- It explains past performance
- It can answer simple questions like
  - What happened ?
  - When did it happen?
- Or it can answer complex questions like
  - How did marketing team performed last quarter in terms of revenue
  - How does that compare to the performance in the same quarter last year
- Frequently used terms
- Metric
  - used to gauge the business performance or progress
  - metric = measure + business meaning
- Key Performance Indicators (KPI):
  - Key: related to the business goals
  - Performance: how successfully you have performed within a specified timeframe
  - Indicators: shows values indicates somethings about the business
- dashboards

Predictive Analytics - Traditional → prediction - forecasting

- After the analysis is over, the next logical step is analytics
- It can be performed traditional statistical modelling like
  - Regression
    - A model used for quantifying causal relationships among the different variables included in your analysis
    - Mostly used for predicting future values
  - Clustering
    - Creating different clusters (groups) by understanding data
  - Factor Analysis
  - Time Series analysis

Predictive Analytics – Machine Learning → predicting

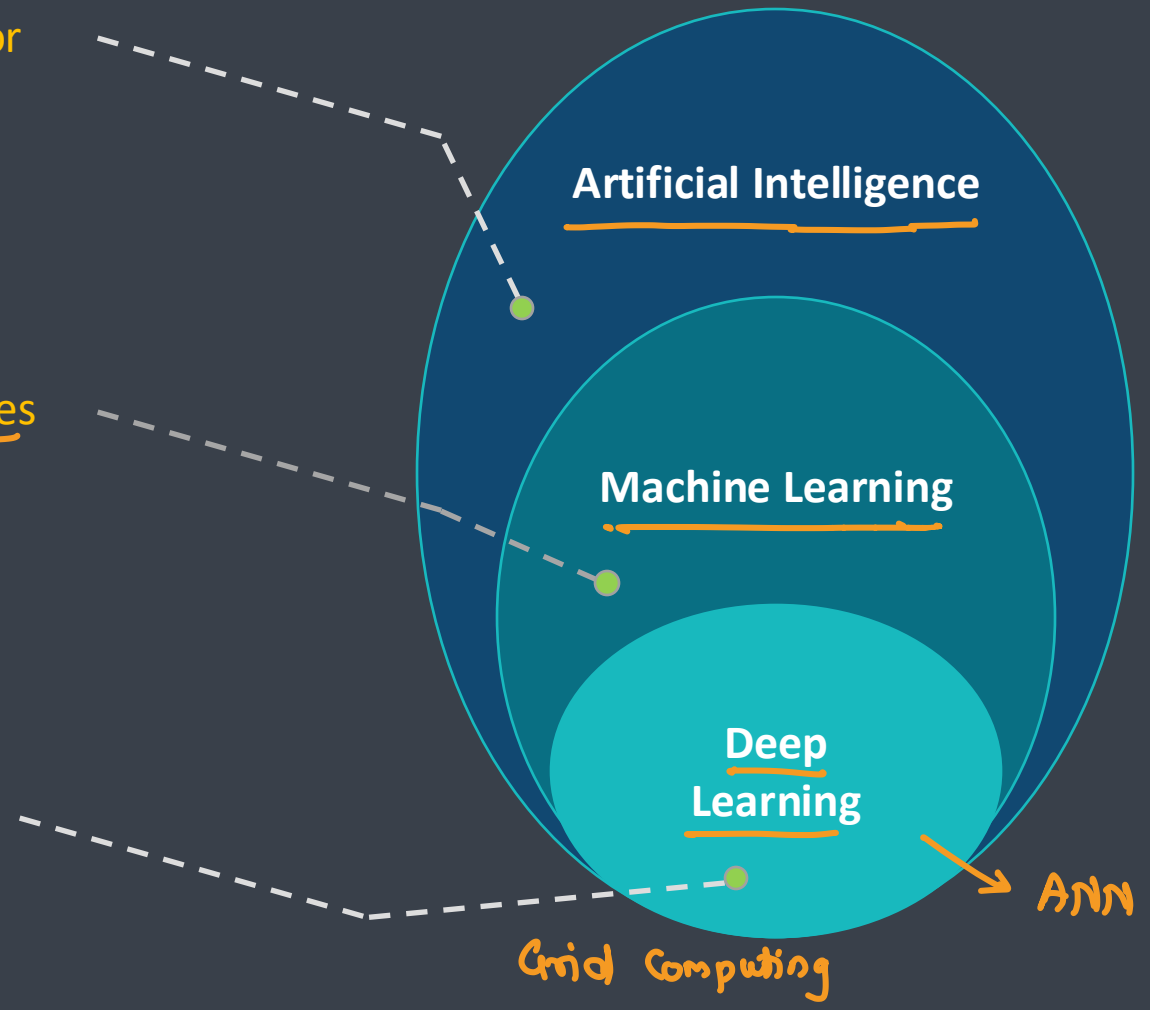
- Utilizes artificial intelligence to predict behavior in unprecedented ways
- There are different techniques
  - Supervised Learning
  - Unsupervised Learning
  - Reinforcement Learning

Biggest confusion

Artificial Intelligence:
- A technique which enables machine to mimic human behavior

Machine Learning:
- Subset of AI which uses statistical methods to enable machines to improve the experience

Deep Learning:
- Subset of ML which makes the computation of multi-layer neural network feasible



Artificial Intelligence

When did it start?

- Greek Mythology – Talos
  - Talos was a giant animated bronze warrior programmed to guard the island of Crete
- 1950 – Alan Turing
  - Alan Turing published a landmark paper in which he speculated about the possibility of creating machines that think
  - What he created is known as Turing Test which is used to determine whether or not the computer can think intelligently like human being
- 1951 – Game AI
  - Christopher Strachey wrote a checkers program and Dietrich Prinz wrote one for chess
- 1956 – The birth of AI
  - John McCarthy first coined the term Artificial Intelligence at Dartmouth Conference
- 1959 – First AI laboratory
  - MIT AI lab was first set up in 1959 and research on AI began

When did it start?

- 1960 – General Motors Robot
  - First robot was introduced to General Motors assembly line
- 1961 – First chatbot
  - The first AI chatbot called ELIZA was introduced in 1961
- 1997 - IBM Deep Blue
  - IBM's Deep Blue beats world champion Garry Kasparov in the game of chess
- 2005 - DARPA Grand Challenge
  - Stanford Racing Team's autonomous robotic car, Stanley wins the 2005 DARPA Grand Challenge
- 2011 – IBM Watson
  - IBM's question answering system, Watson, defeated the two grated Jeopardy champions Brad Ruther and Ken Jennings

What is AI?

- Artificial Intelligence (AI), sometimes called machine intelligence, is intelligence demonstrated by machines, in contrast to the natural intelligence displayed by humans
- Any device that perceives its environment and takes actions that maximize its chance of successfully achieving its goals
- The theory and development of computer system able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision making and translation
- Often used to describe machines (or computers) that mimic "cognitive" functions that humans associate with the human mind, such as "learning" and "problem solving"

Aspects of AI (1955)

learning, thinking, problem solving

- Simulating higher functions of the human brain
- Programming a computer to use general languages → NLP
- Arranging hypothetical neurons in a manner so that they can form concepts → DL
- A way to determine and measure problem complexity
- Self-improvement
- Abstraction: defined as the quality of dealing with ideas rather than events
- Randomness and creativity
  - → GenAI

Why are we talking about it now ?

More Computational Power

More storage

More Data

Better algorithms

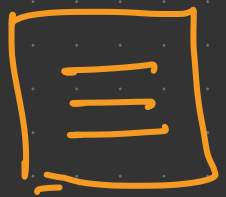
$$$

Broad investment

AI applications

- Google's search engine
- JPMorgan Chase's Contract Intelligence (COiN) platform uses AI, machine learning and image recognition software to analyse legal documents
- IBM Watson: Healthcare organizations use IBM AI (Watson) technology for medial diagnosis
- Google's AI Eye Doctor can examine retina scans and identify a condition called as diabetic retinopathy which can cause blindness
- Facebook uses ML and DL to detect facial features and tag your friends
- Twitter uses AI to identify hate speech and terroristic language in the tweets
- Smart Assistants: Siri, Google Assistant, Alexa, Cortana → NLP
- Tesla automated cars
- Netflix uses AI for movie recommendations
- Spam filtering

Machine Learning



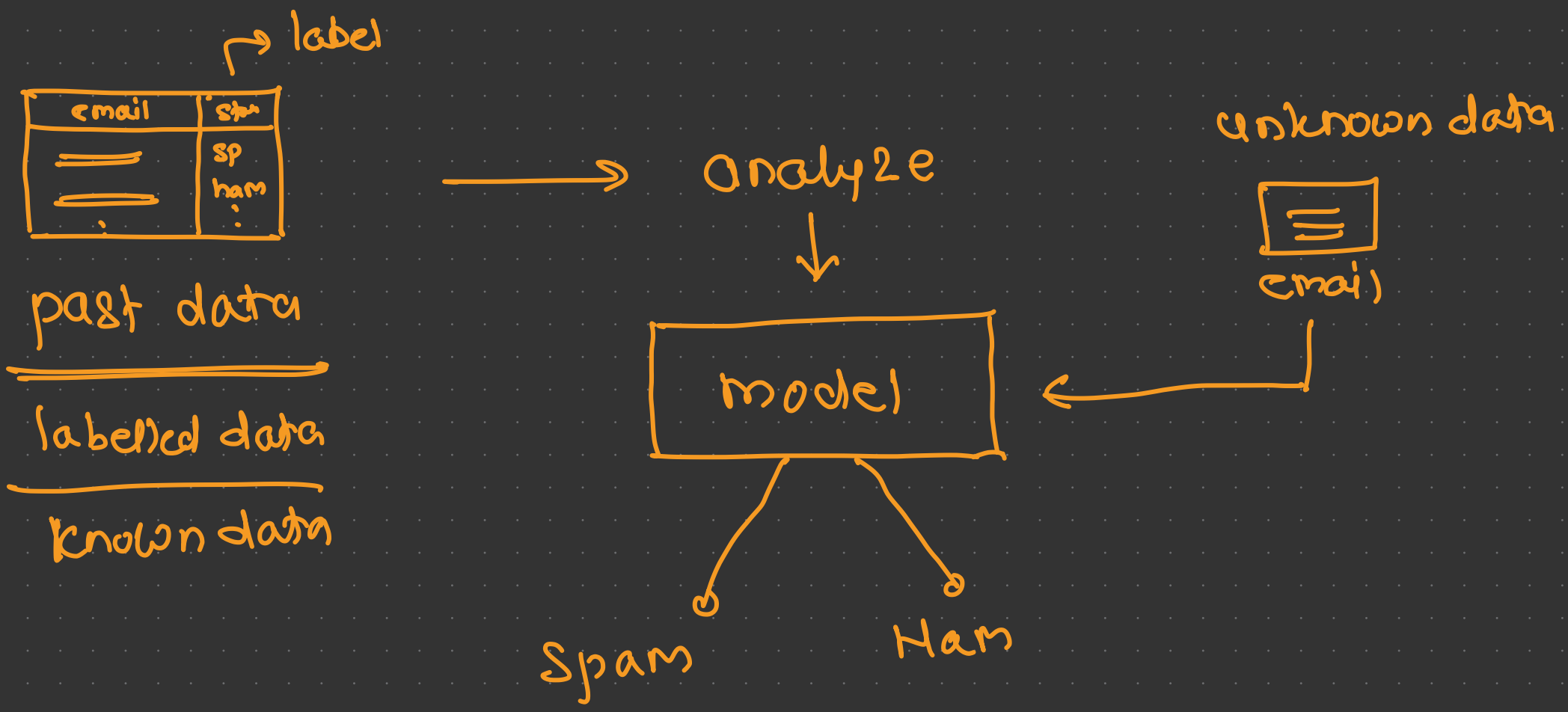
email

traditional approach

```
if source == "x.x.x.x":
    mark spam
elif email contains "words":
    mark spam
    ⋮
else:
    mark ham
```

① source = x.x.x.x
② words
③ ⋮
④

machine learning approach



Traditional approach

① algorithm [formula] is already known

② language is used to implement the algorithm

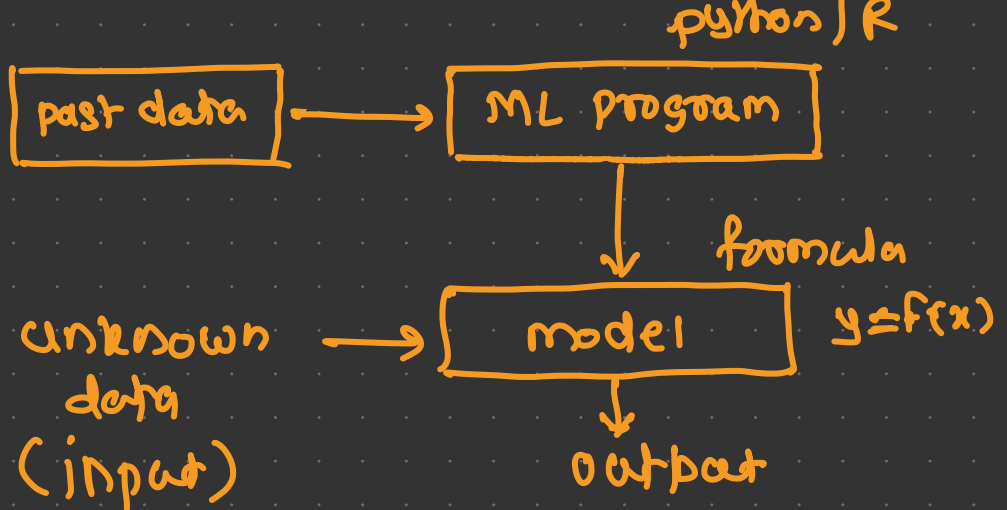
ML approach

$y = f(x)$

① formula is unknown

② language is used to find the model

③ a past data is mandatory



What is machine learning ?

- A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E
  - Tom Mitchell, 1997

- Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed
  - Arthur Samuel, 1959

- Machine Learning is the science (and art) of programming computers so they can learn from data

Where to use machine learning ?

- Problems for which existing solutions require a lot of fine-tuning or long lists of rules:
  - one Machine Learning algorithm can often simplify code and perform better than the traditional approach
- Complex problems for which using a traditional approach yields no good solution:
  - the best Machine Learning techniques can perhaps find a solution
- Fluctuating environments:
  - a Machine Learning system can adapt to new data
- Getting insights about complex problems and large amounts of data

Examples of Applications

- Analyzing images of products on a production line to automatically classify them
  - This is image classification, typically performed using convolutional neural networks
- Detecting tumors in brain scans
  - This is semantic segmentation, where each pixel in the image is classified (typically use CNNs)
- Automatically classifying news articles
  - This is natural language processing (NLP), and more specifically text classification
- Automatically flagging offensive comments on discussion forums
  - This is also text classification, using the same NLP tools
- Forecasting your company's revenue next year, based on many performance metrics
  - This is a regression task (i.e., predicting values) that may be tackled using any regression model
- Making your app react to voice commands
  - This is speech recognition, which requires processing audio samples: since they are long and complex sequences, they are typically processed using RNNs, CNNs, or Transformers

Examples of Applications

- Detecting credit card fraud
  - This is anomaly detection example
- Segmenting clients based on their purchases so that you can design a different marketing strategy for each segment
  - This is clustering example
- Representing a complex, high-dimensional dataset in a clear and insightful diagram
  - This is data visualization, often involving dimensionality reduction techniques
- Recommending a product that a client may be interested in, based on past purchases
  - This is a recommender system
- Building an intelligent bot for a game
  - This is often tackled using Reinforcement Learning

Types

Types of machine learning

- There are so many different types of Machine Learning systems that it is useful to classify them in broad categories, based on the following criteria
  - Whether or not they are trained with human supervision
    - supervised, unsupervised, and Reinforcement Learning
  - Whether or not they can learn incrementally on the fly
    - online versus batch learning
  - Whether they work by simply comparing new data points to known data points, or instead by detecting patterns in the training data and building a predictive model, much like scientists do
    - instance-based versus model-based learning

machine learning

supervised

- Regression
  - Linear
  - Lasso
  - Ridge
- classification
  - Logistic Regression
  - decision tree
  - Naive Bayes
  - Ensemble learning

unsupervised

- clustering
  - KMeans
  - Hierarchical
  - DBScan
- association Rules mining
  - Apriori
  - Eclat
- Dimensionality Reduction
  - PCA
  - LDA
  - GDA

Reinforcement

Self- Learning

- Q-learning
- Deep Q-learning

Supervised – prediction

Unsupervised – No prediction [EDA]

Reinforcement Learning → self learning

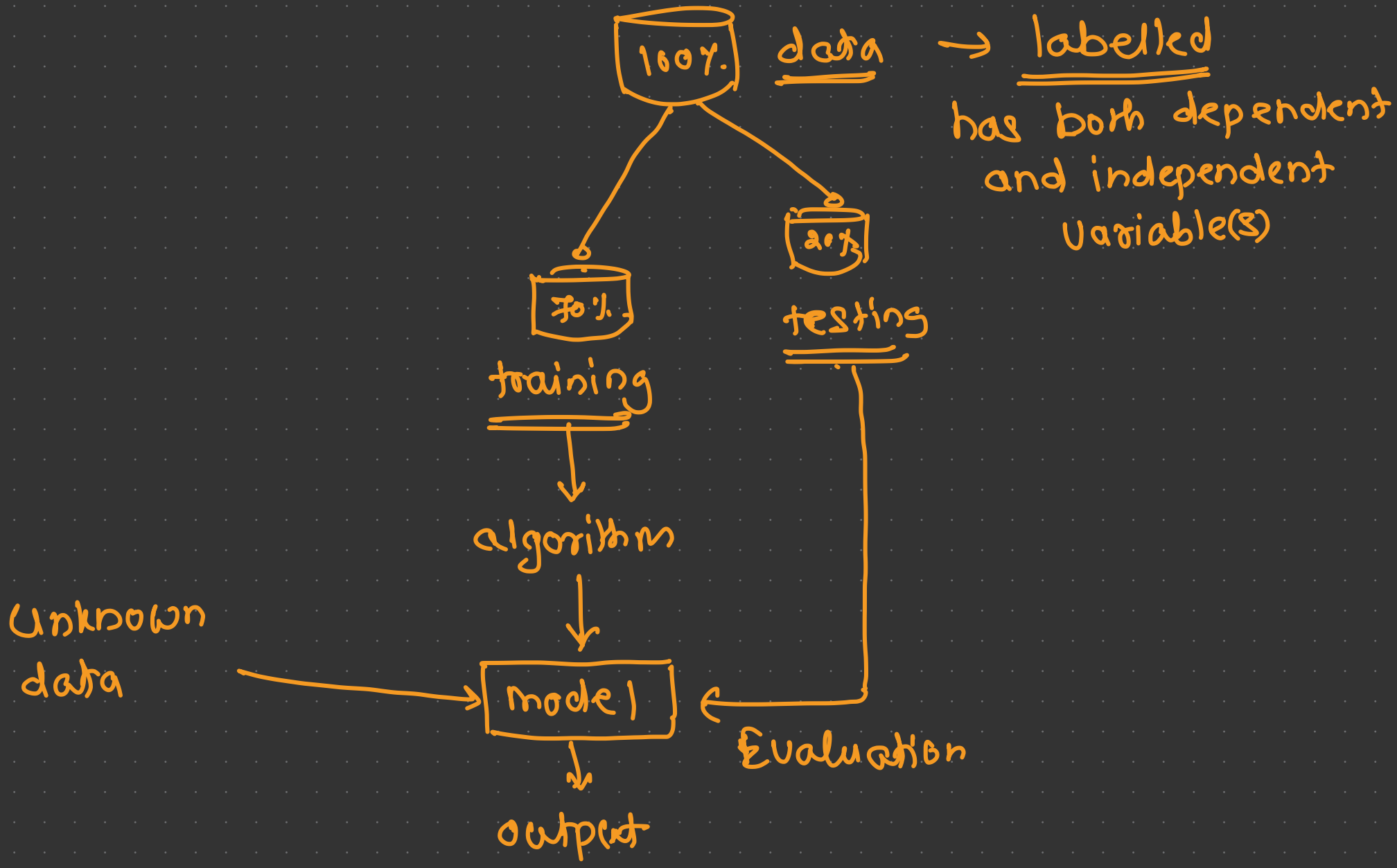
Supervised Learning → accuracy / performance of model can be measured

- The majority of practical machine learning uses supervised learning
- Supervised learning is where you have input variables (x) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output

x_1	x_2	y

$$Y = f(X)$$ → model → formula

- The goal is to approximate the mapping function so well that when you have new input data (x) that you can predict the output variables (Y) for that data
- It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process
- We know the correct answers, the algorithm iteratively makes predictions on the training data and is corrected by the teacher
- Learning stops when the algorithm achieves an acceptable level of performance


100%
data
labelled
has both dependent and independent variable(s)
80%
training
20%
testing
algorithm
Unknown data
model
Evaluation
output

Supervised Learning – Problems

- **Regression** → dependent variable is non-categorical [discrete]
  - Related to predicting future values
  - E.g.
    - Population growth prediction
    - Expecting life expectancy
    - Market forecasting/prediction
    - Advertising Popularity prediction
    - Stock prediction
  - Algorithms
    - Linear and multi-linear regression
    - Logistic regression
    - Naïve Bayes
    - Support Vector Machine

Lasso Regression

Ridge Regression

Elastic Net

Exp	Sal
⋮	⋮

	x		y → dependent
no	rev	Year	profit
–	–	2010	2.3
–	–	2011	2.5
–	–	2012	3.2
		⋮	⋮
		2025	?

discrete value

45M

algorithm – model

Linear Reg

profit = f(rev..)

model

salary = f(exp)

Supervised Learning – Problems

- **Classification** → dependent variable is of categorical type
  - Related to classify the records
  - E.g.
    - Find whether an email received is a spam or ham
    - Identify customer segments → 1st, 2nd, 3rd
    - Find if a bank loan is granted → yes / No
    - Identify if a kid will pass or fail in an examination
  - Algorithms
    - Logistic Regression
    - Decision Tree
    - Random Forest → Ensemble learning
    - Support Vector Machine
    - K-nearest neighbor

Ada Boost

Gradiant Boost

eXtreme Gradiant Boost (XGBoost)

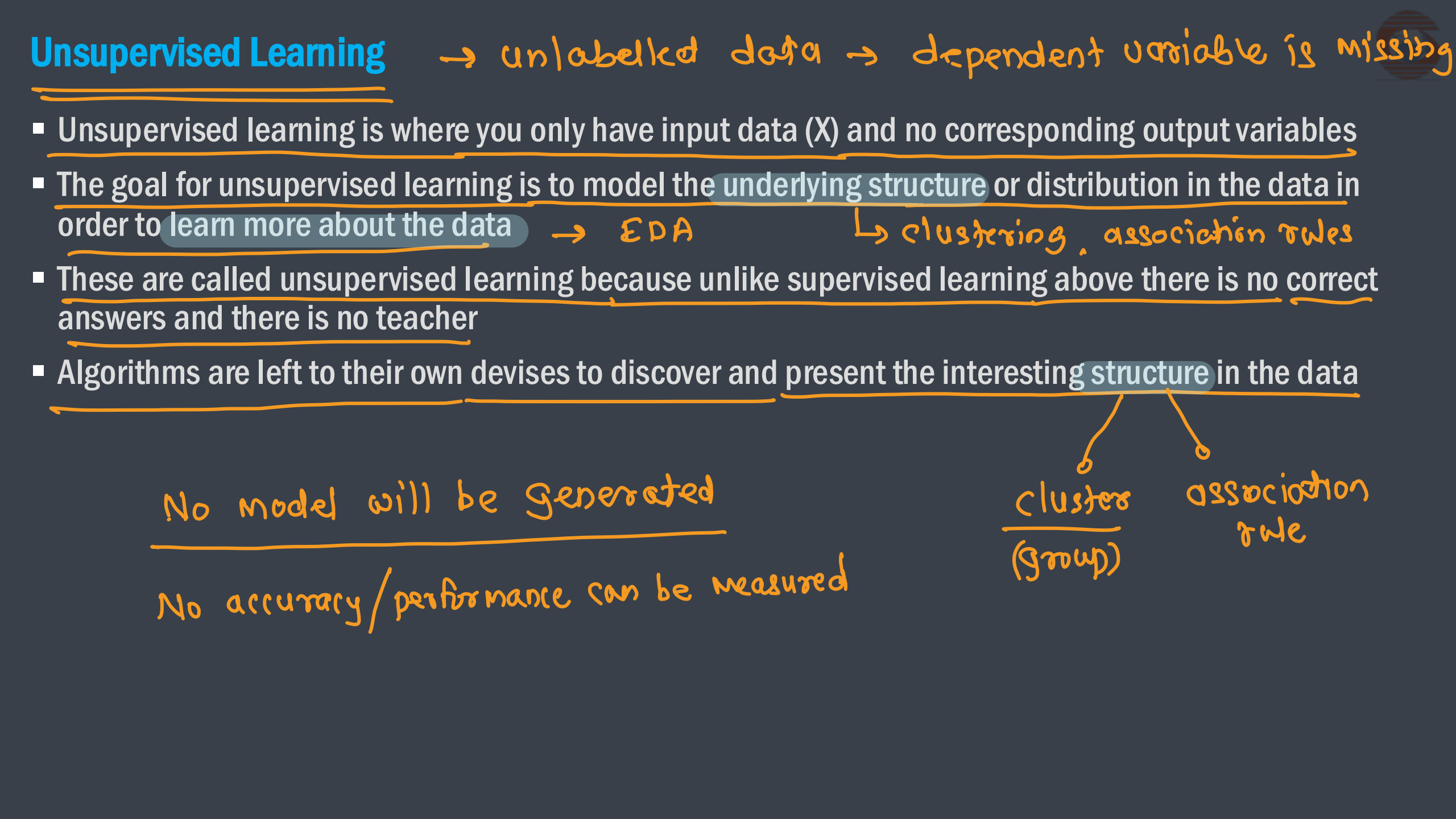
y → dependent → categorical

No	name	marks	result
1	Yug	90%	pass
2	par	85%	failed
3	abc	85%	failed
⋮	⋮	⋮	⋮

categories → labels / classes

Unsupervised Learning → unlabelled data → dependent variable is missing

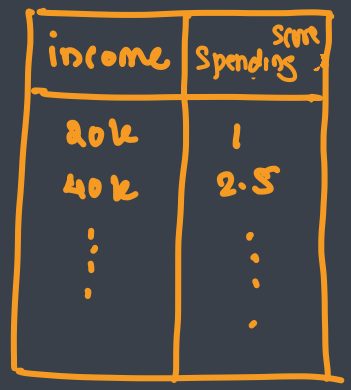
- Unsupervised learning is where you only have input data (X) and no corresponding output variables
- The goal for unsupervised learning is to model the underlying structure or distribution in the data in order to learn more about the data → EDA → clustering, association rules
- These are called unsupervised learning because unlike supervised learning above there is no correct answers and there is no teacher
- Algorithms are left to their own devises to discover and present the interesting structure in the data



Unsupervised Learning - Problems

- **Clustering**
  - discover the inherent groupings in the data, such as grouping customers by purchasing behaviour
  - E.g.
    - Batsman vs bolwer
    - Customer spending more money vs less money
  - Algorithms
    - K-means clustering
    - Hierarchical clustering

    DBScan clustering



Unsupervised Learning - Problems

- **Association Rules Mining** —
  - An association rule learning problem is where you want to discover rules that describe large portions of your data, such as people that buy X also tend to buy Y
  - E.g.
    - Market basket analysis
  - Algorithms
    - Apriori
    - Eclat

Feature Engineering
- → Feature selection
- → Feature Extraction

* Dimensionality Reduction
  - → Reducing high dimensions ⇒ low dimensions space
  - ⇒ dimension = feature = column = variable
  - algorithms
    - PCA - correlation
    - LDA

Dimensionality Reduction

- The number of input features, variables, or columns present in a given dataset is known as dimensionality, and the process to reduce these features is called dimensionality reduction
- A dataset contains a huge number of input features in various cases, which makes the predictive modeling task more complicated → complex, time taking
- Because it is very difficult to visualize or make predictions for the training dataset with a high number of features, for such cases, dimensionality reduction techniques are required to use
- Dimensionality reduction technique can be defined as, "It is a way of converting the higher dimensions dataset into lesser dimensions dataset ensuring that it provides similar information." → correlation
- These techniques are widely used in machine learning for obtaining a better fit predictive model while solving the classification and regression problems

Dimensionality Reduction

- Features Selection
  - Filter
  - Wrapper
  - Embedded
- Features Extraction
  - Principal Component Analysis (PCA)
  - Linear Discriminant Analysis (LDA)
  - Generalized Discriminant Analysis (GDA)

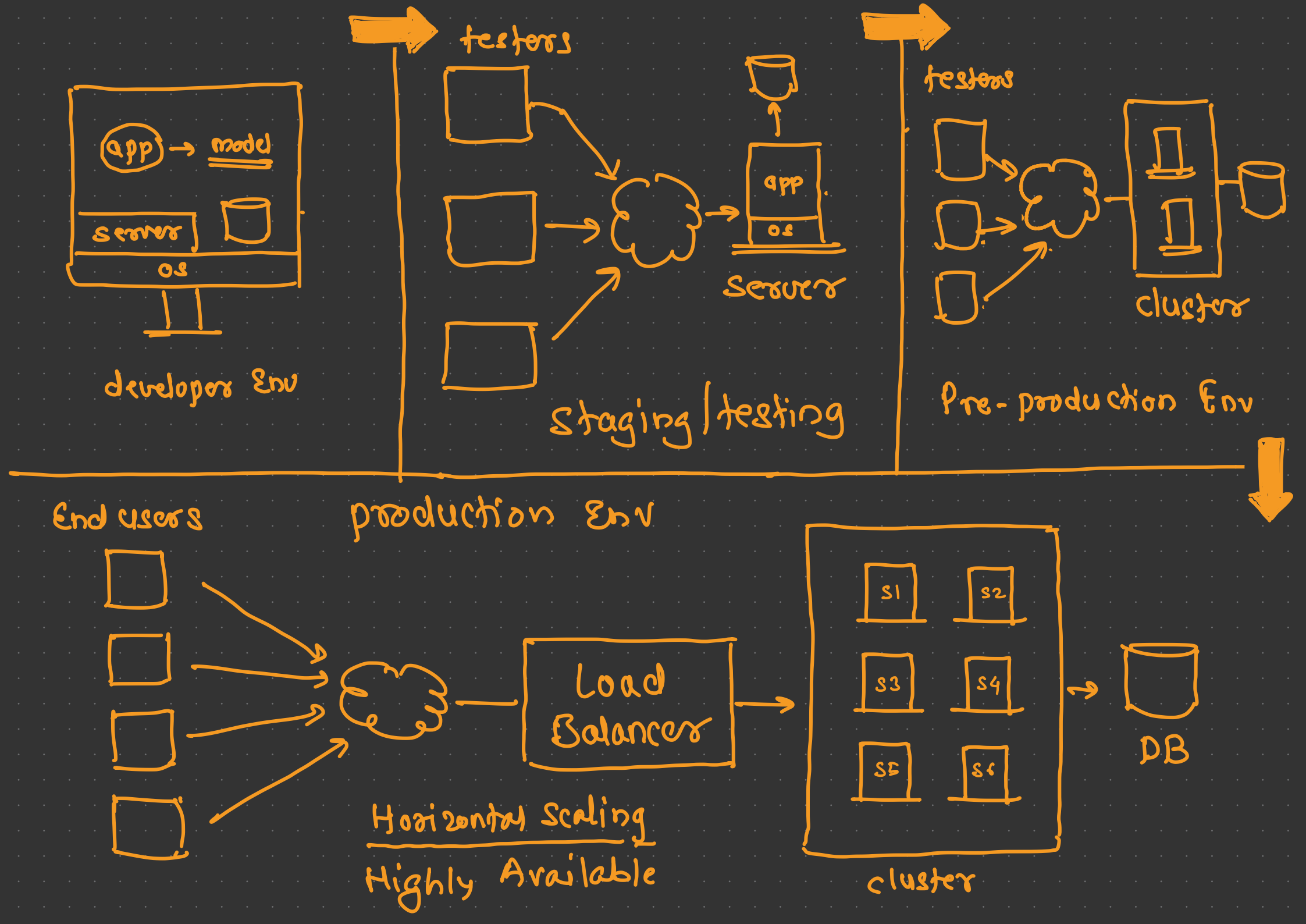
Reinforcement Learning – agent based Learning

- It is about taking suitable action to maximize reward in a particular situation
- It is employed by various software and machines to find the best possible behavor or path it should take in a specific situation
- Reinforcement learning differs from the supervised learning in a way that in supervised learning the training data has the answer key with it so the model is trained with the correct answer itself whereas in reinforcement learning, there is no answer but the reinforcement agent decides what to do to perform the given task
- In the absence of training dataset, it is bound to learn from its experience

Reinforcement Learning

- Examples
  - Resources management in computer clusters
  - Traffic Light Control
  - Robotics
  - Web system configuration
  - Chemistry
- Algorithms
  - Q-Learning
  - Deep Q-Learning

Batch Learning
Online Learning

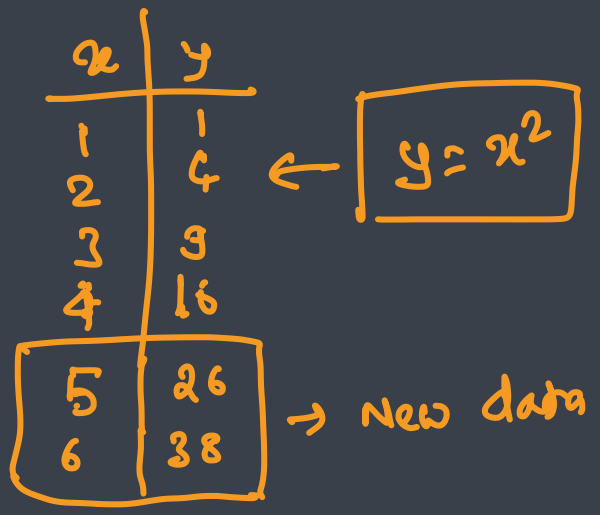

app
model
server
os
developers Env
testers
app
os
Server
staging / testing
testers
cluster
Pre-production Env
End users
production Env
Load Balancer
Horizontal Scaling
Highly Available
S1
S2
S3
S4
S5
S6
cluster
DB

Batch Learning → offline learning

- In batch learning, the system is incapable of learning incrementally
- it must be trained using all the available data
- This will generally take a lot of time and computing resources, so it is typically done offline
- First the system is trained, and then it is launched into production and runs without learning anymore, it just applies what it has learned
- This is also called as offline learning

Batch Learning - cons

- If you want a batch learning system to know about new data, you need to train a new version of the system from scratch on the full dataset, then stop the old system and replace it with the new one
  - The whole process of training, evaluating, and launching a Machine Learning system can be automated easily
- Training using the full set of data can take many hours
  - Typically train a new system only every 24 hours or even just weekly
- Training on the full set of data requires a lot of computing resources (CPU, memory space, disk space, disk I/O, network I/O)



Online Learning

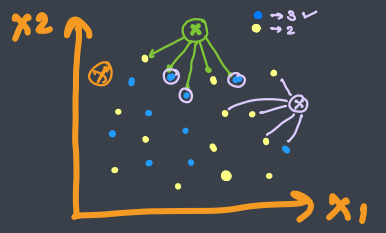
- In online learning, you train the system incrementally by feeding it data instances sequentially, either individually or in small groups called mini-batches
- Each learning step is fast and cheap, so the system can learn about new data on the fly, as it arrives
- Online learning is great for systems that receive data as a continuous flow (e.g., stock prices) and need to adapt to change rapidly or autonomously
- It is also a good option if you have limited computing resources
  - once an online learning system has learned about new data instances, it does not need them anymore, so you can discard them
  - This can save a huge amount of space
- Online learning algorithms can also be used to train systems on huge datasets that cannot fit in one machine's main memory (this is called **out-of-core learning**)

Online Learning

- One important parameter of online learning systems is how fast they should adapt to changing data: this is called the **learning rate**
- If you set a high learning rate, then your system will rapidly adapt to new data, but it will also tend to quickly forget the old data
  - you don't want a spam filter to flag only the latest kinds of spam it was shown
- if you set a low learning rate, the system will have more inertia
  - that is, it will learn more slowly, but it will also be less sensitive to noise in the new data or to sequences of nonrepresentative data points (outliers)

Instance Based
Model Based

Instance Based

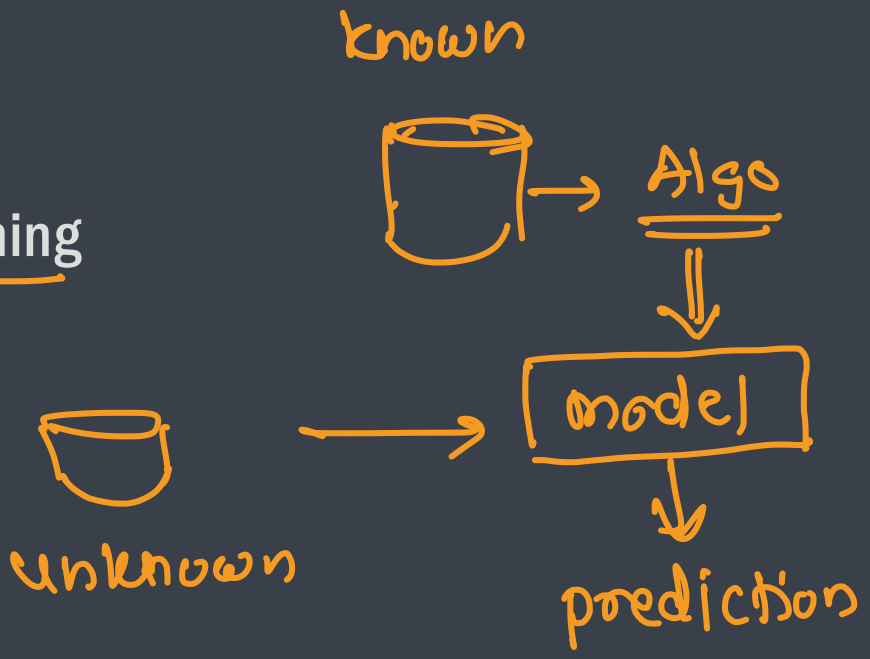


- The system learns the examples by heart
- Then generalizes to new cases by using a similarity measure to compare them to the learned examples (or a subset of them)
  - ↳ distance
- It is called instance-based because it builds the hypotheses from the training instances
  - ↳ Records
- It is also known as **memory-based learning** or **lazy-learning**
- **Advantages:**
  - Instead of estimating for the entire instance set, local approximations can be made to the target function
  - This algorithm can adapt to new data easily, one which is collected as we go
- **Disadvantages:**
  - Classification costs are high
  - Large amount of memory required to store the data, and each query involves starting the identification of a local model from scratch
- E.g.
  - K Nearest Neighbor (KNN)

Model Based

model = formula

- Train model from training data to estimate model parameters i.e. discover patterns
- Store the built model in suitable format → pickle
- Generalize the rules of model
- Predict the unseen instance (data) using the model
- It requires a known model form
- It takes less memory compared to the instance based learning
- E.g.
  - Linear Regression



End to End Process

Steps

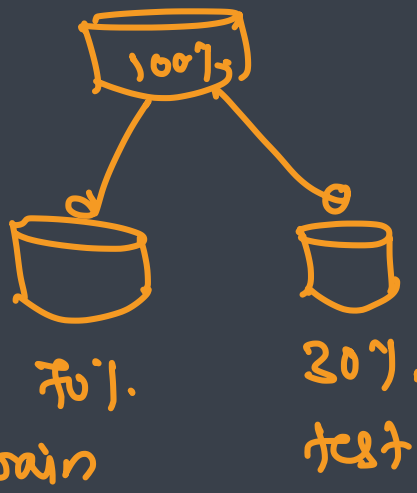
- Look at the big picture
- Get the data → collect / organize
- Discover and visualize the data to gain insights
- Prepare the data for Machine Learning algorithms → data cleansing
- Select a model and train it → by trial and error method
- Fine-tune your model → hyper parameter tuning → optimizing model
- Present your solution
- Launch, monitor, and maintain your system

Look at the Big Picture

- Frame the Problem → Domain knowledge
  - The first question to ask your boss is what exactly the business objective is
  - Building a model is probably not the end goal
  - How does the company expect to use and benefit from this model?
  - Knowing the objective is important because it will determine
    - how you frame the problem → Regression / classification
    - which algorithms you will select → Random forest
    - which performance measure you will use to evaluate your model → MSE / MAE / R2 or Confusion Matrix / F1 score
    - how much effort you will spend tweaking it → hyper parameter
- Select a Performance Measure
  - Your next step is to select a performance measure
  - A typical performance measure for regression problems is the Root Mean Square Error (RMSE) / R2
  - It gives an idea of how much error the system typically makes in its predictions, with a higher weight for large errors

Get the data

- Decide the data source
  - REST API:
  - Web scraping
  - organize
    - files → csv / json / xml
    - db
    - storage ..
- Download the data and make it available for the further learning
- Take a Quick Look at the Data Structure
  - Understand the data set and features
    - dependent & independent → correlation analysis
  - Evaluate the features and decide which one(s) are needed
    - → feature engineering
- Create a Test Set
  - Keep some records aside for testing and validation



- data cleansing

  Basic

Discover and Visualize the Data to Gain Insights → data analysis

↳ EDA

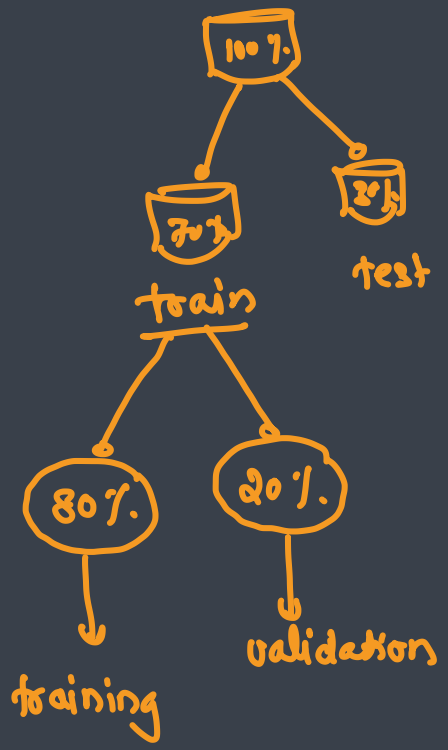
- Visualize the data
  - Use libraries like matplotlib or seaborn → charts / graphs
  - Understand the pattern and relationship
- Look for correlation
- Experiment with attribute combinations

Prepare the Data for Machine Learning Algorithms

- Data Cleaning
  - Process of cleaning the data set to prepare it for ML algorithm
  - Steps
    - Check for the missing data
    - Check for wrong data types
    - Add features if needed
    - Remove unwanted features
- Feature Scaling
  - ML algorithms don't perform well when the input numerical attributes have very different scale
  - Scale the features to bring all of them to a single scale
- Handle categorical / text data
  - Use transformers to convert categorical to numerical

Select and Train a Model

- Training the model using train data set
  - Create a model using selected algorithm
  - Save the model for future use → pickle → file
- Evaluation the model
  - Evaluate the model to see if there is any chance to improve the accuracy
  - Techniques
    - Cross Validation
      - → k-fold cross validation



Fine-Tune Your Model → finding best values for hyper parameters

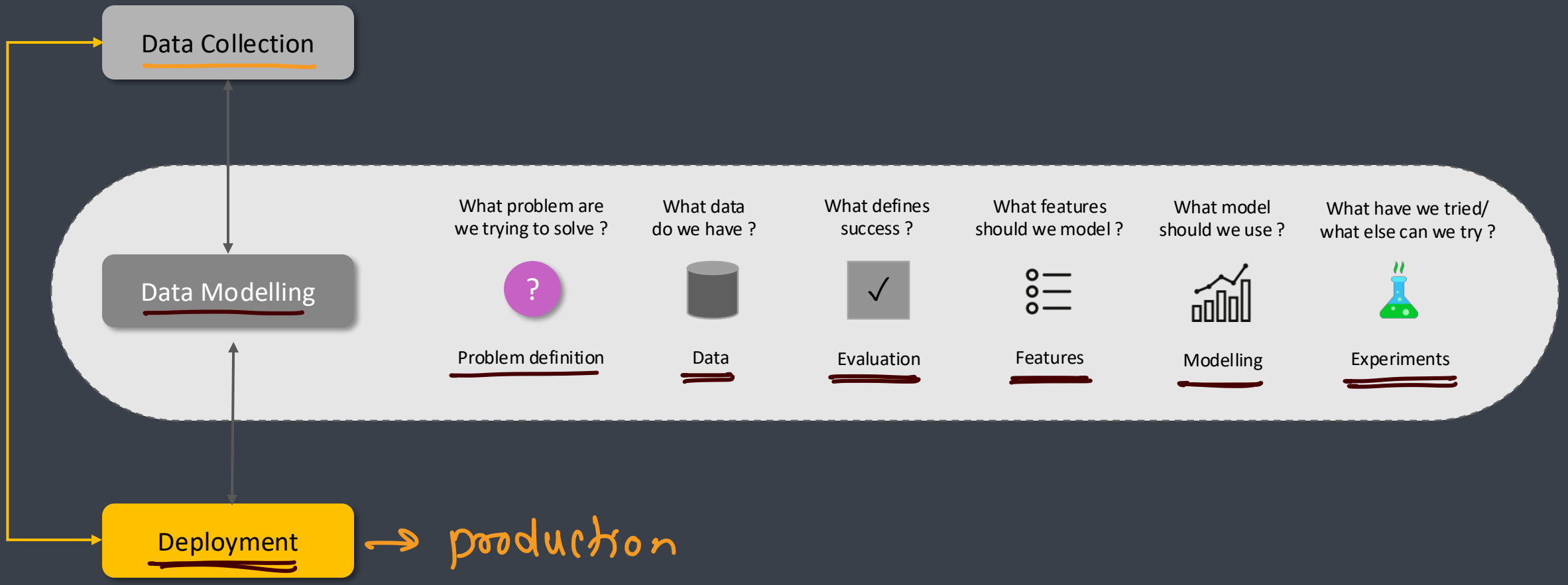
↳ configuration of algorithm

- Grid Search
  - One option would be to fiddle with the hyperparameters manually, until you find a great combination of hyperparameter values
  - This would be very tedious work, and you may not have time to explore many combinations
  - You can also automate this process using libraries like sci-kit
- Randomized Search
  - The grid search approach is fine when you are exploring relatively few combinations
  - But when the hyperparameter search space is large, it is often preferable to use randomized search
- Ensemble Methods
  - Another way to fine-tune your system is to try to combine the models that perform best
  - The group (or "ensemble") will often perform better than the best individual model, especially if the individual models make very different types of errors. → bagging / boosting / stacking
- Analyze the Best Models and Their Errors
- Evaluate Your System on the Test Set

Launch, Monitor, and Maintain Your System

- Deploy the application for the end users → production
- Monitor the application's performance
- If the data keeps evolving, update your datasets and retrain your model regularly
- You should probably automate the whole process as much as possible → MLOps
  - Collect fresh data regularly and label it
  - Write a script to train the model and fine-tune the hyperparameters automatically. This script could run automatically, for example every day or every week, depending on your needs → cron jobs
  - Write another script that will evaluate both the new model and the previous model on the updated test set, and deploy the model to production if the performance has not decreased (if it did, make sure you investigate why)

Summary


Data Collection
Data Modelling
Deployment
production
What problem are we trying to solve ?
Problem definition
What data do we have ?
Data
What defines success ?
Evaluation
What features should we model ?
Features
What model should we use ?
Modelling
What have we tried/ what else can we try ?
Experiments