taplingua

TALENT COCO

Day 2 - AI/ML Projects

Predicting heart disease through classification

What we will cover

Layout the roadmap for the project and how it fits into the larger world of AI/ML models

Discuss the problem a hand - classifying data if the patient has heart disease

Start coding in your model 

Steps for AI/ML projects

Collect Data

Data Analysis (Feature engineering and selection) → Machine Learning

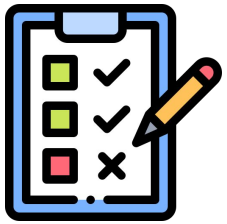
1. Define the Problem

What is the problem statement and business case?

2. Get Data

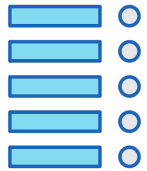
Get the data

3. Evaluate



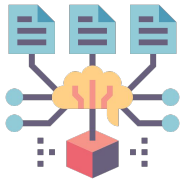
Define success

4. Features



Pick the features

5. Model



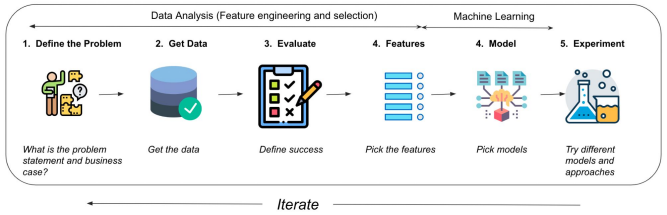
Pick models

6. Experiment

Try different models and approaches

Iterate

Deploy model



1. Define the problem

What is the problem statement and business case?

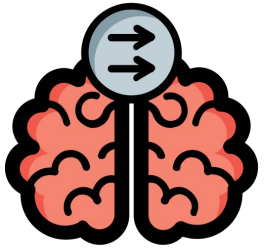
The client is a major US hospital that has hired us to see if we can help their doctors more easily predict heart disease in patients. They have shared with a large dataset of clinical data of past patients. They want us to see if we can build an ML model that can help the physicians diagnose disease faster

Do we need ML or a traditional instruction based system?

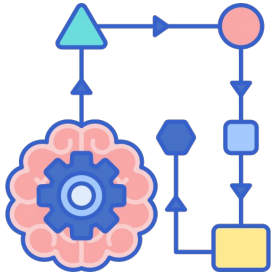
Main types of machine learning

Supervised

Unsupervised



Transfer learning



Reinforcement learning

Supervised learning

In Supervised learning we have labeled data

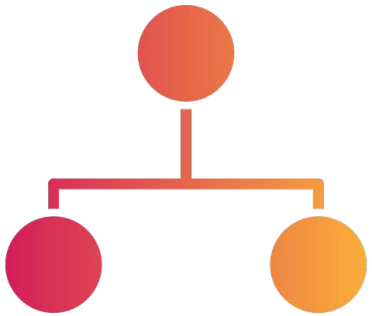
Data

Data is labeled

Id	BP (Systolic)	Chest pain	LDL	VDL	HDL	Heart Disease
1	90	Yes	180	250	95	Yes
2	110	No	180	250	95	No
3	120	No	180	250	95	No

Our heart disease dataset has a lot of clinical data as well as the labels

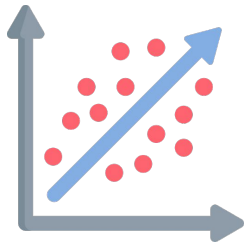
Supervised Learning



Classification

Given the data, is it yes or no?

Binary classification - yes or no
Multi class - many options e.g. type of flower

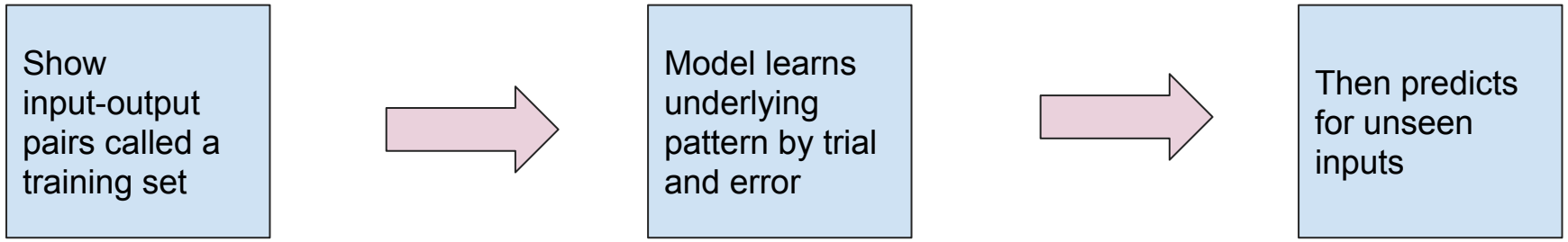


Regression

What is the amount or value of something?

price of a house based on number of bedrooms
predict the amount of rainfall in a month

How supervised learning works?

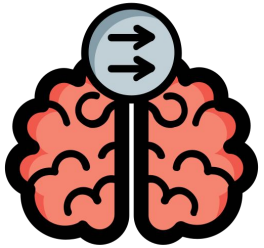


During training, the model is presented with a set of input-output pairs, called a training set. The model tries to learn the underlying pattern or relationship between the inputs and outputs in the training set, which can then be used to predict the output for new, unseen inputs.

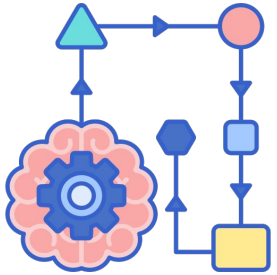
Main types of machine learning

Supervised

Unsupervised



Transfer learning

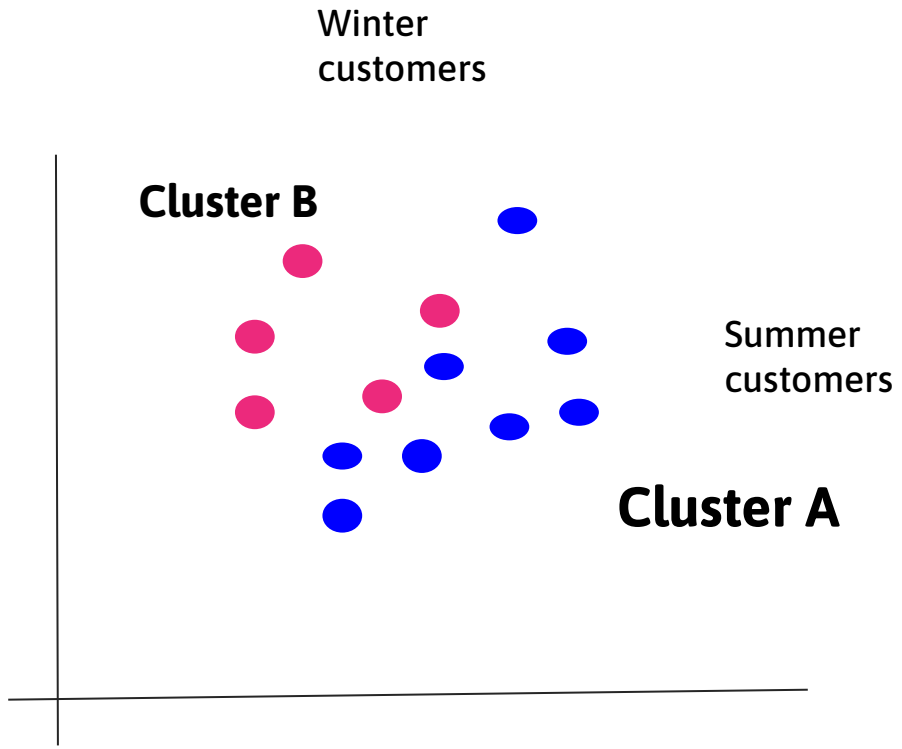


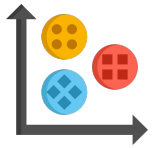
Reinforcement learning

Unsupervised learning

Data but no labels

	Purchase 1	Purchase 1
Customer 1	Ski boots	Jackets
Customer 2	Sunscreen	Beach towel
Customer 3	Sunglasses	Sunscreen
Customer 4	Wool hat	Heave jacket



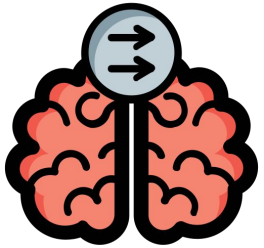


An example of unsupervised learning is **clustering**. Clustering is a type of machine learning algorithm in which a model is trained to group similar data points together based on their similarities or patterns.

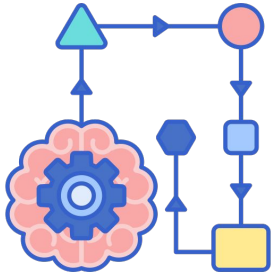
Transfer learning

Supervised

Unsupervised

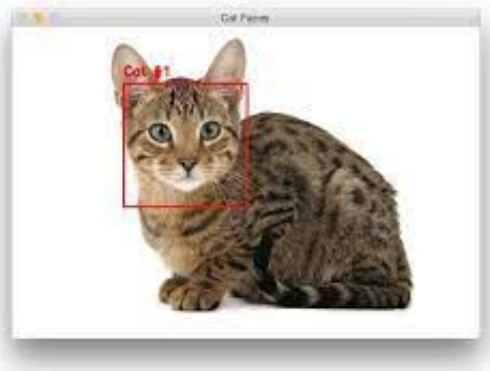


Transfer learning



Reinforcement learning

Transfer learning



Model is trained to detect cats

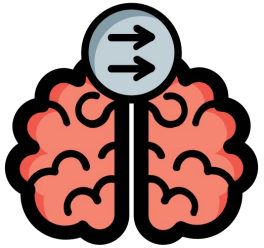
What breed is this dog?

Use a pre-trained model (which has already learned how to solve a related problem and has developed a set of weights and biases) that can be used as a starting point for the new task. This approach can save a lot of time and computational resources compared to training a new model from scratch. Example is the Resnet model that can do object detection. You train it on your task

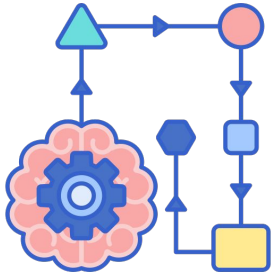
Main types of machine learning

Supervised

Unsupervised

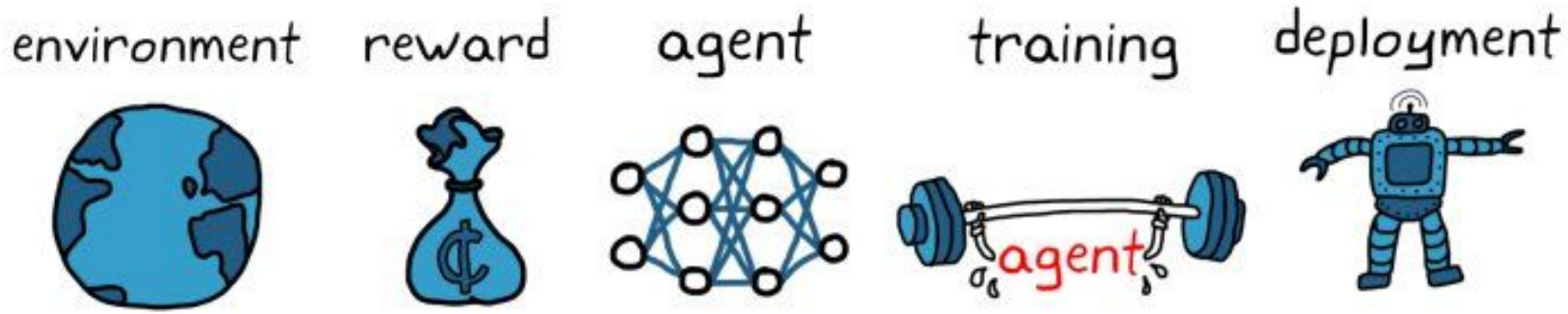


Transfer learning



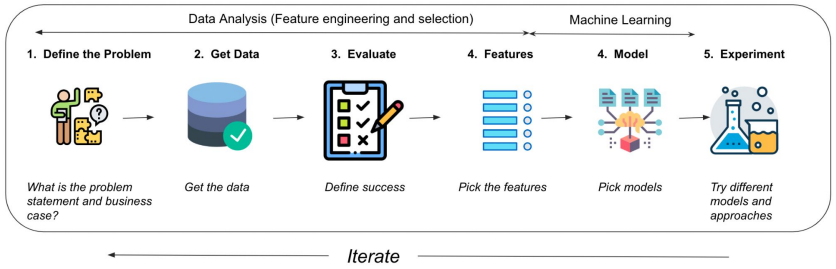
Reinforcement learning

Reinforcement learning



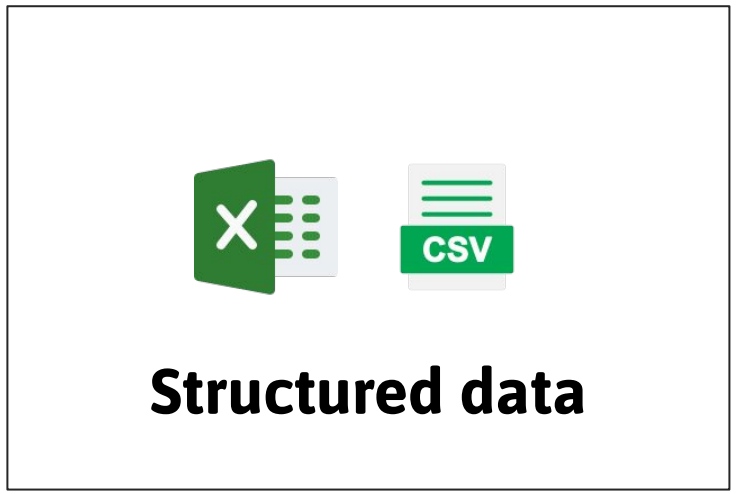
In reinforcement learning, the agent (algo) interacts with the environment and decides what is to be done and is given penalties or reward to improve the next decision. There is no initial dataset (as in supervised learning)

Source: KDNuggets



2. What data do we have?

Types of Data - structured



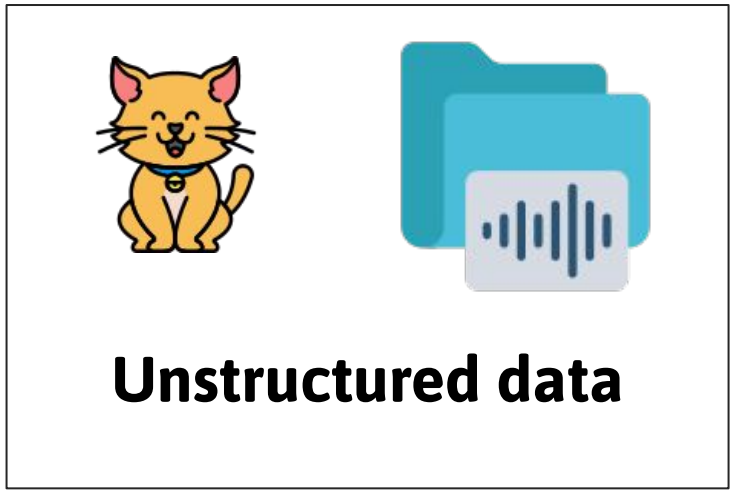
All records are in the same format and fit in a CSV or Excel

Columns

Customer Id	Purchase	Amount (Rs)	Date
1	Ski boots	5000	01/03/2020
2	Sunscreen	3500	01/05/2020
3	Sunglasses	2000	01/07/2020
4	Wool hat	4500	01/09/2020

Rows

Types of Data - unstructured



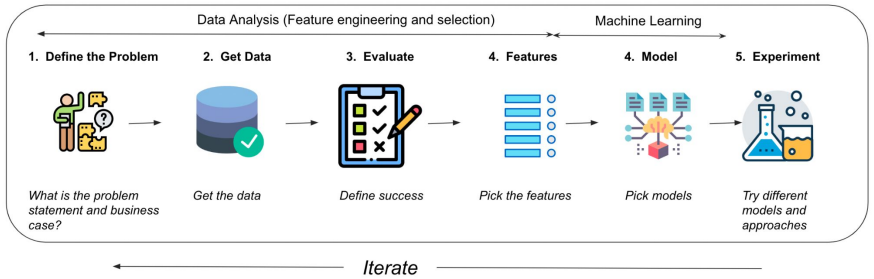
Text, audio, video, image

The data doesn't fit into any specific model or schema. Cannot be easily searched using traditional tools.

Examples, could be twitter feed, youtube videos, customer reviews etc.

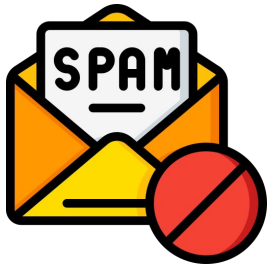
Sources of data

1. Company specific
2. Public datasets
3. IoT
4. Web scraping/APIs
5. Social media etc



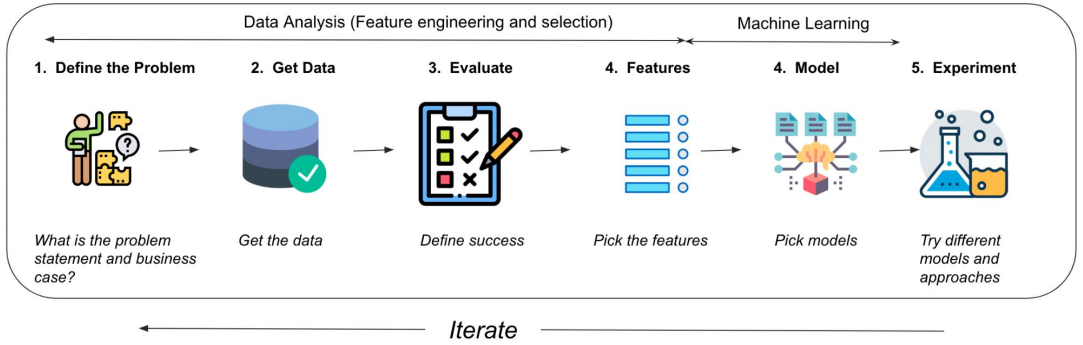
3. How to evaluate our model

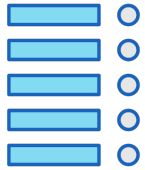
Is it okay for our model to predict with 60% accuracy for predicting heart disease?
What about spam detection?
What about a self driving car?



Different types of metrics

Classification	Regression	Recommendation
Accuracy	Mean absolute error (MAE)	Precision at K
Precision	Mean squared error (MSE)	
Recall	Root mean squared error (RSME)	





4. What are the features of the data?

Features in Data

Features are attributes or variables that are used to build a model or analyze data

Feature variables

Target variable

House	Bedrooms	Sq Feet	PIN code	Bus connection	Price
1001	2	900	560076	56	50 lacs
1004	4	1200	560076	75	75 lacs
1002	5	1500	560078	37	80 lacs

Numerical variable

Categorical variable

For example, if you were building a model to predict housing prices, some features might include the number of bedrooms, the size of the house, and the location/PIN

Feature engineering - transforming the data

New feature

House	Bedrooms	Sq Feet	PIN code	Bus connection	Price	Metro line
1001	2	900	560076	56	50 lacs	Y
1004	4	1200	560076	75	75 lacs	Y
1002	5	1500	560078	37	80 lacs	N

Feature engineering is the process of selecting/transforming raw data into features that are suitable for use in a machine learning model - Adding, transforming or normalizing the variables e.g. if we knew that the metro line was in pin code 560075 we could do feature engineering by adding a new feature Metro.

Feature coverage - there should be enough rows with the feature for it to work well.e.g. we don't all the bus lines and has low coverage

Feature engineering

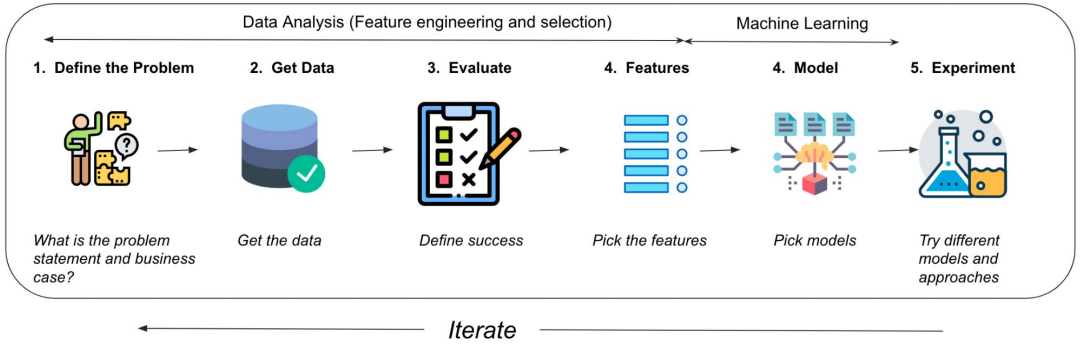
Typical issues in data

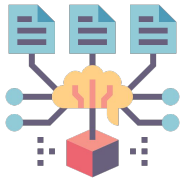
- Missing or null values,
- Duplicate rows,
- Irrelevant columns,
- Outliers,
- Correlation between columns

These conditions can affect model performance hence we can do feature engineering to address the issue

e.g. how to address null values in your dataset?

Deleting rows or columns with null values
Imputing missing values
Using a separate category (if categorical variable)
Using machine learning algorithms

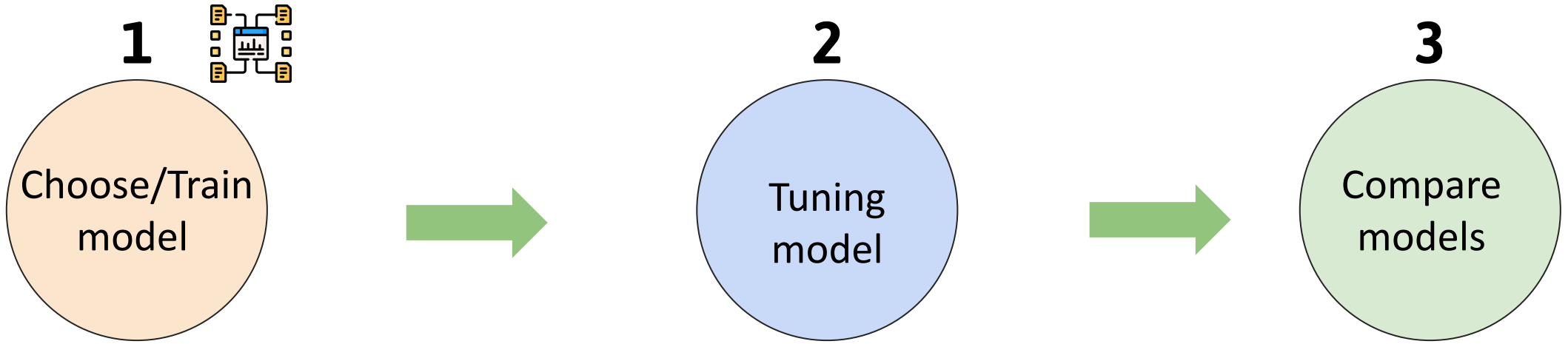




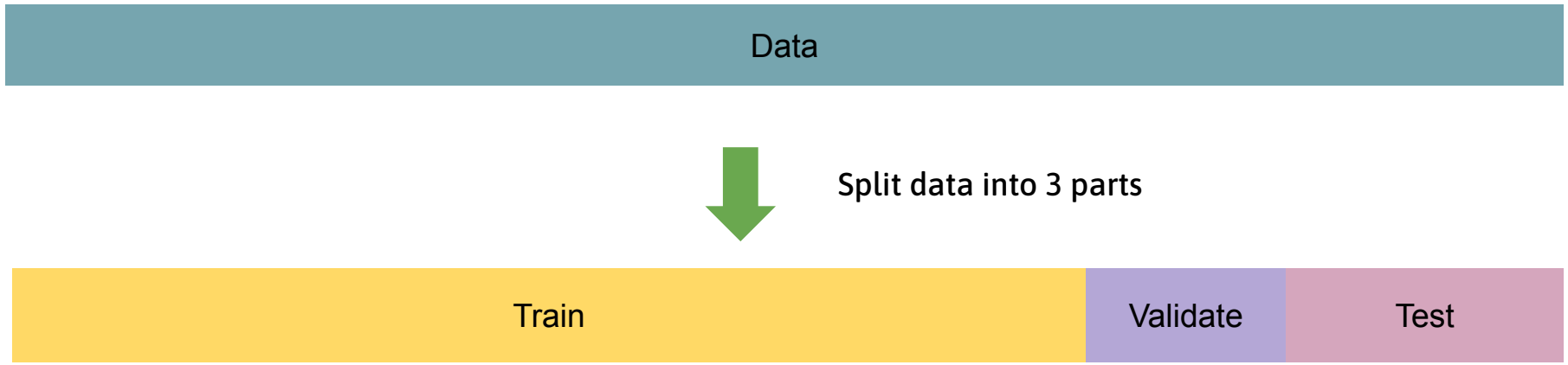
5. Pick the model

Given the problem and the data we have, what model should we choose

3 steps to modelling



Train, validate, test (3 sets split)



How well will your model do in the real world? To answer, we need 3 sets of the data

Why do we need the Train, Validate, Test data split?

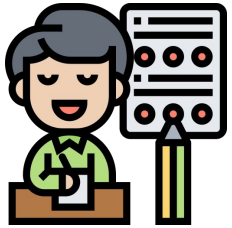
Consider a student preparing for a major exam …

Train

Study for an Exam

The training data

Validate



Take practice exams

Validation (Tuning the model)

Test

Final exam

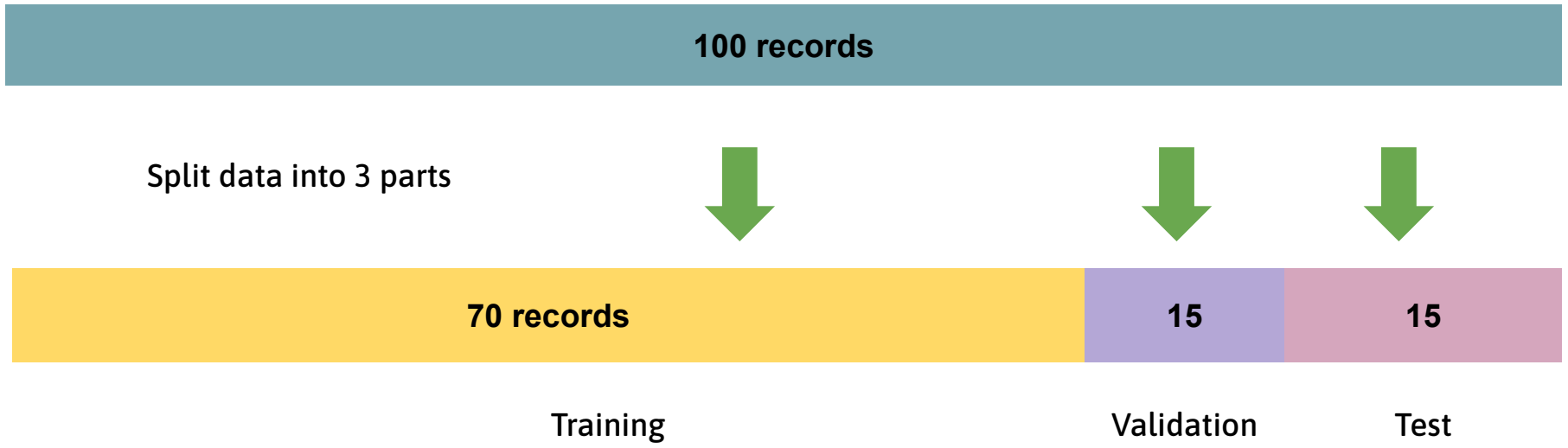
Test data

Student does well in questions not seen before

The Test

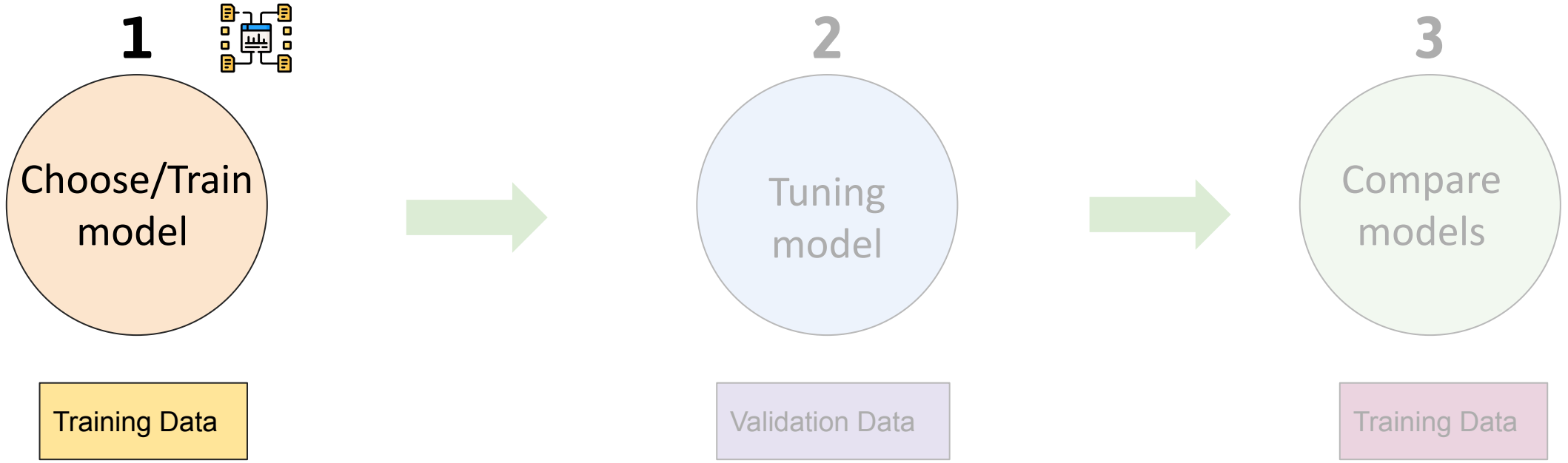
Can the algo generalized well (when it is working on <u>unseen data</u> - the exam).
Or can you do well in the final exam :)

Train, validate, test (3 sets split)



A good split for data is 70-15-15 or so. or 80-10-10 etc.

3 steps to modelling



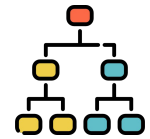
1. Choose the model based on problem at hand

Structured Data

Unstructured Data

dmlc XGBoost

Gradient boosting



Random forest

TensorFlow

PyTorch

Deep learning and transfer learning

1. Train the model

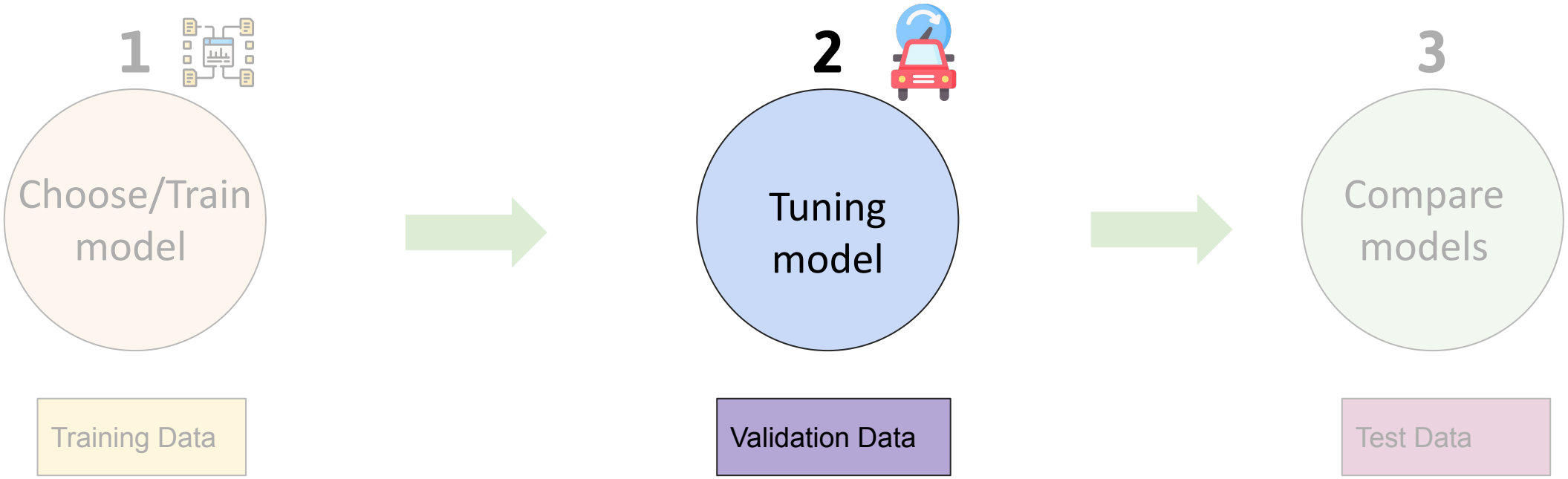
Training Data

X (data) Y (label)

House	Bedrooms	Sq Feet	PIN code	Bus connection	Price
1001	2	900	560076	56	50 lacs
1004	4	1200	560076	75	75 lacs
1002	5	1500	560078	37	80 lacs

Try to minimize the time it takes to train the model as you have to iterate
(may be pick a smaller dataset or choose a lighter model and then add complexity)

3 steps to modelling



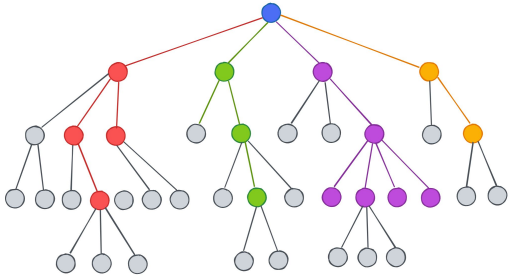
Tuning your model is like tuning your car

Validation - tune the model

It’s like tuning your car…adjust the ignition time, coolant, T-belt, shocks etc.

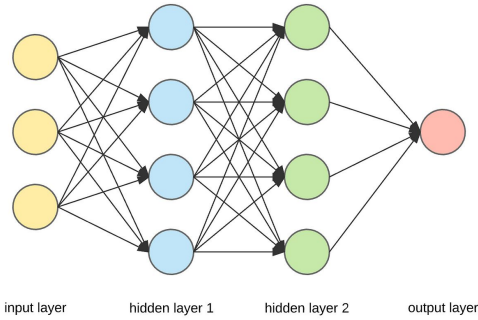
Tuning your model (using hyperparameters

Random forest



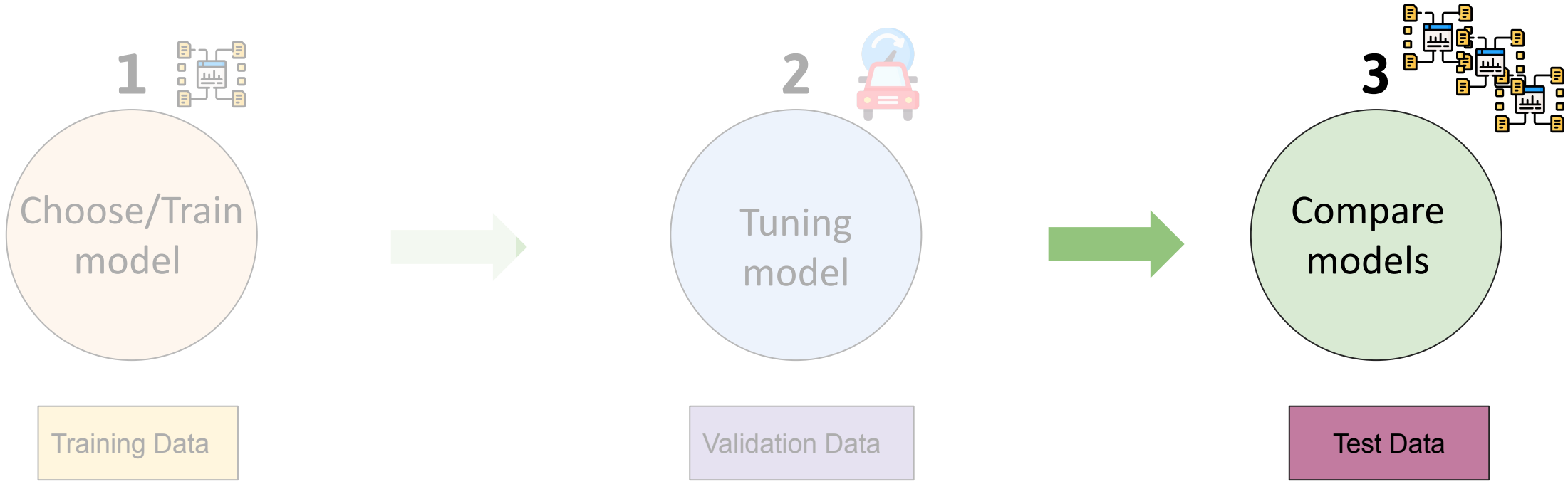
Adjust the depth of the tree (estimators, max features etc.)

Deep Learning



Adjust the number of layers & neuron etc

3 steps to modelling



Model performance with training data & test data

Model v1

Model	Performance
Training Data	94%
Test Data	63%

Underfitting 

Test data performance is well below training data

Model v2

Model	Performance
Training Data	94%
Test Data	92%

Good model 

Test data performance is close to training data

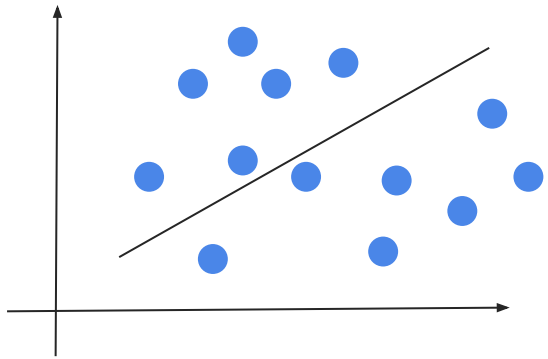
Model v3

Model	Performance
Training Data	94%
Test Data	99%

Overfitting 

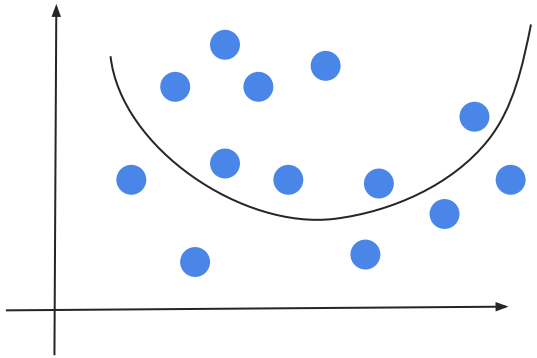
Test data performance is well above training data

Overfitting and Underfitting

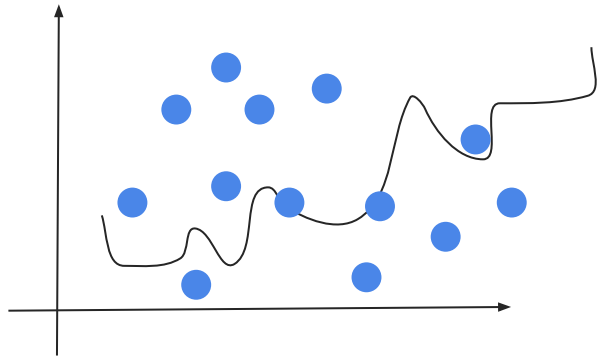


Doing well with training data but poorly with test data implies underfitting

Fix:
A more complex model, more data, add features, reduce bias

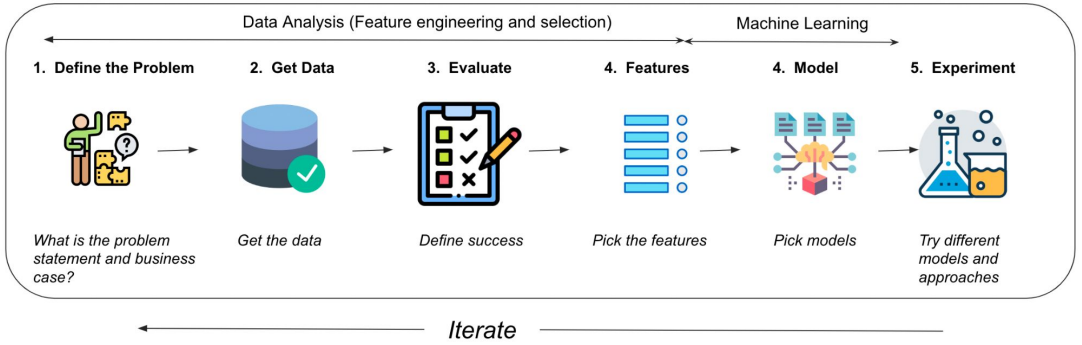


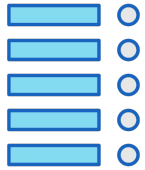
Model is just right



Doing well with training data but poorly with test data implies underfitting. Model is capturing noise

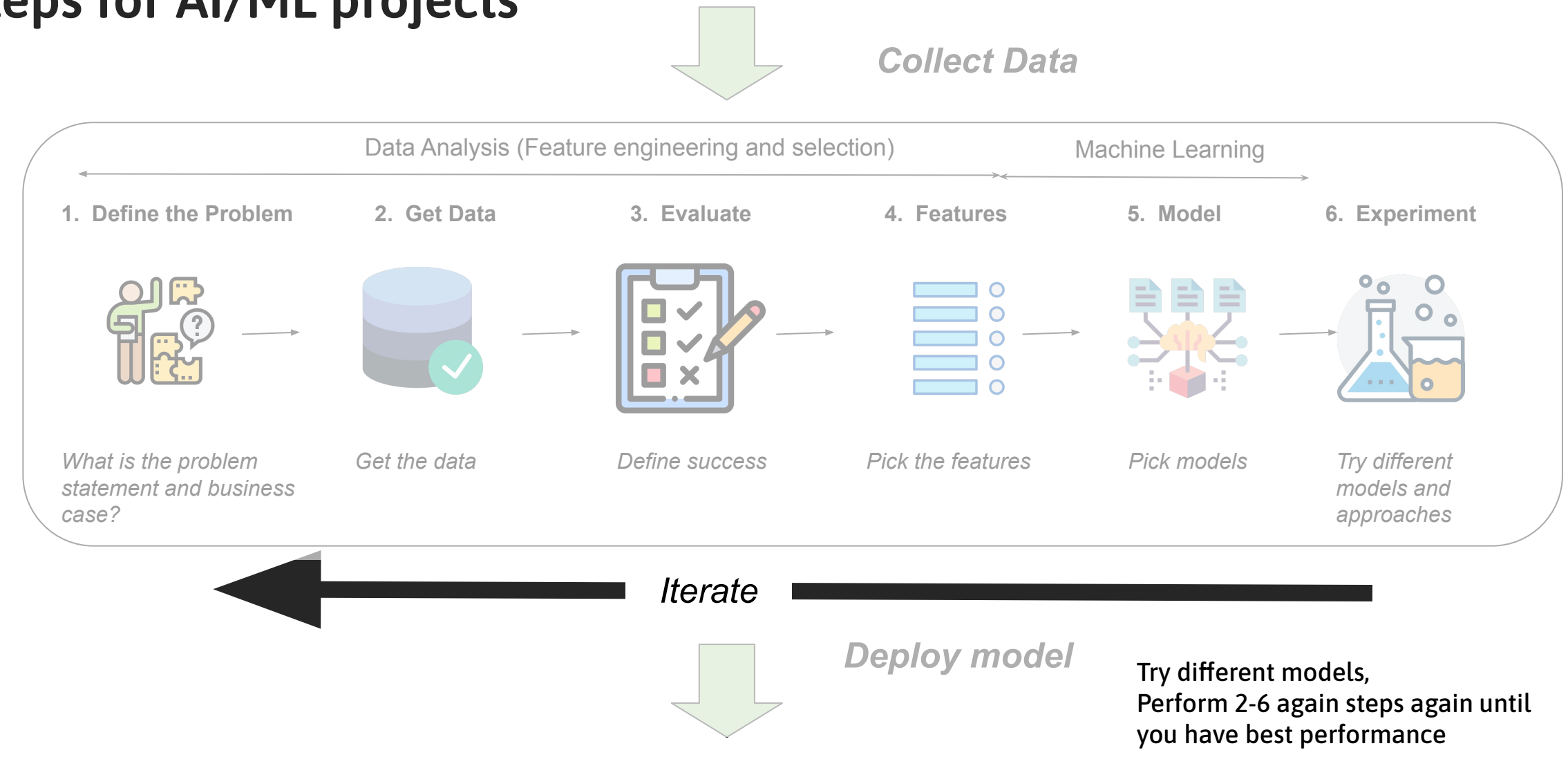
Fix:
Less advanced model, more data, data augmentation, regularization





6. Experiment

Steps for AI/ML projects



Try different models,
Perform 2-6 again steps again until you have best performance

Key points to build a great model

1. Avoid overfitting and underfitting
2. Isolate test data, don’t mix with training data
3. Ensure the datasets are the same
4. Balance time to train model and accuracy

Assignment

1. Download the CSV for the heart disease dataset
2. Do research on the features in the dataset
3. Open a new colab notebook heart-disease.ipynb
4. Use the panda library to read the dataset
   a. import pandas as pd
   b. df = pd.read_csv('https://talentcocomedia.s3.amazonaws.com/ml-assets/heart-disease.csv')
   c. df.head()
   d. Describe the features in the data set in the notebook in the comments section. Research on the internet - a place to start is https://archive.ics.uci.edu/ml/datasets/heart+disease
5. Download the file once research is done and written down your research
6. Commit the file and share the link in our Whatsapp group