The Magic of Machine Learning

Agenda

- Demystifying Machine Learning
- Python setup
- The Art of Data Viz.
- Fundamentals of Data prep.
- Supervised Learning
- Clustering
- Image Recognition
- MLOps
- Career Guidance

Know the Presenter

Abilash B
Software Engineer – Data Scientist

Know the Presenter

Section - 1

Demystifying Machine Learning

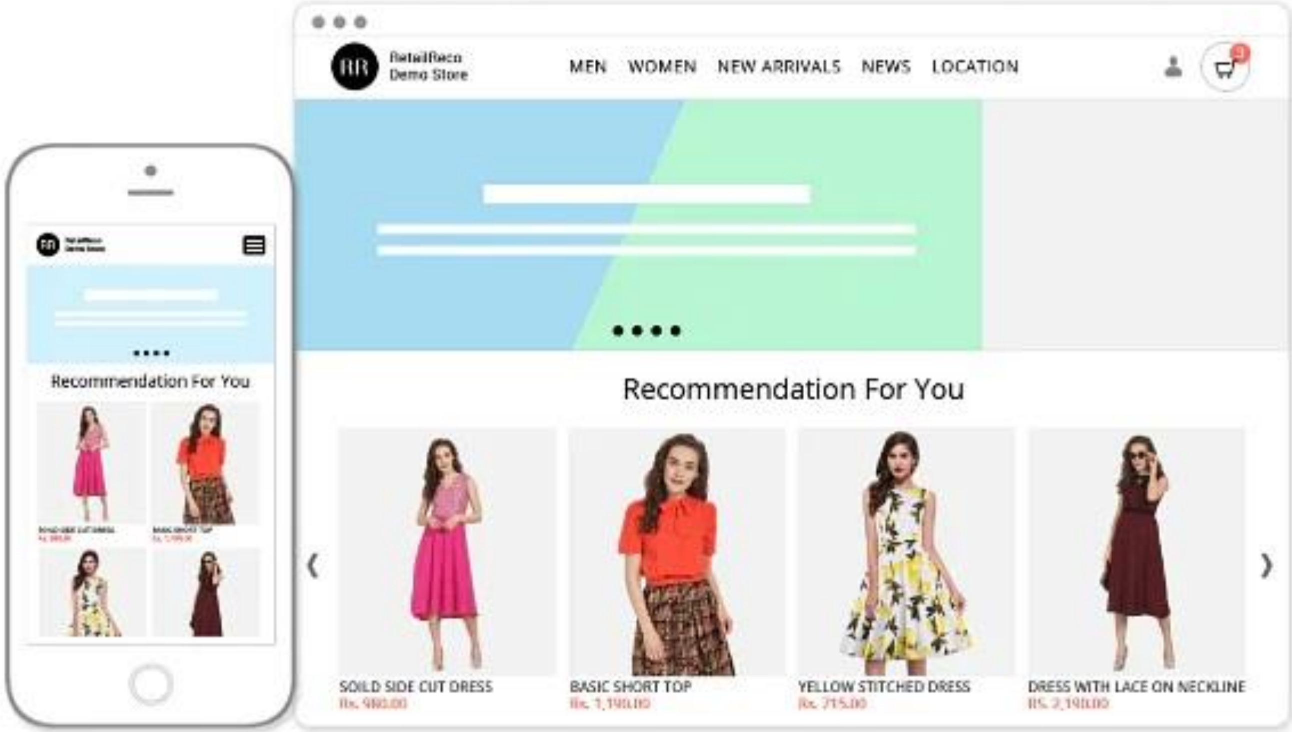
Section - 1.1

Where is Machine Learning used in Real life ?

Everywhere....!!

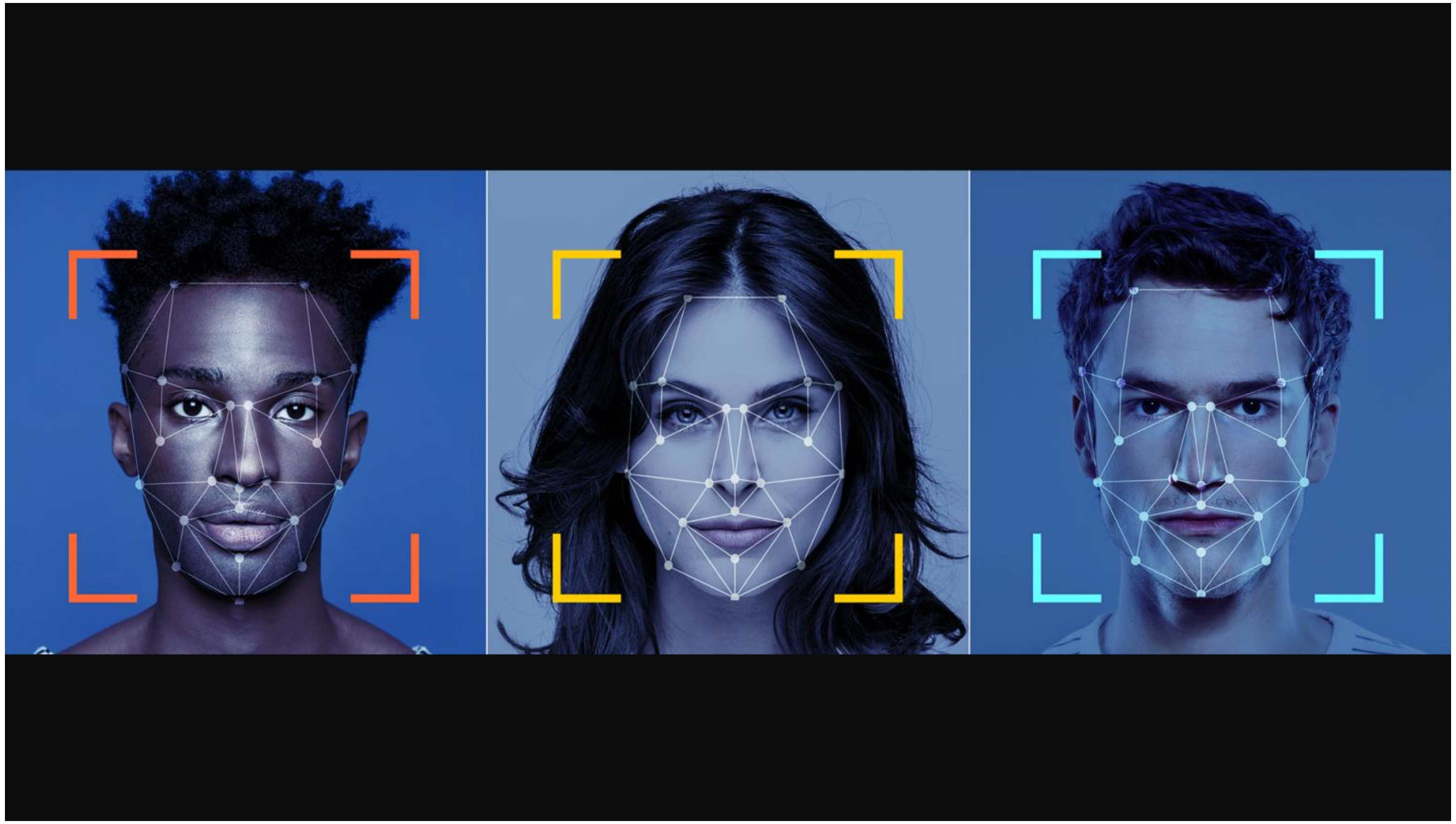
Applications

Product Recommendation



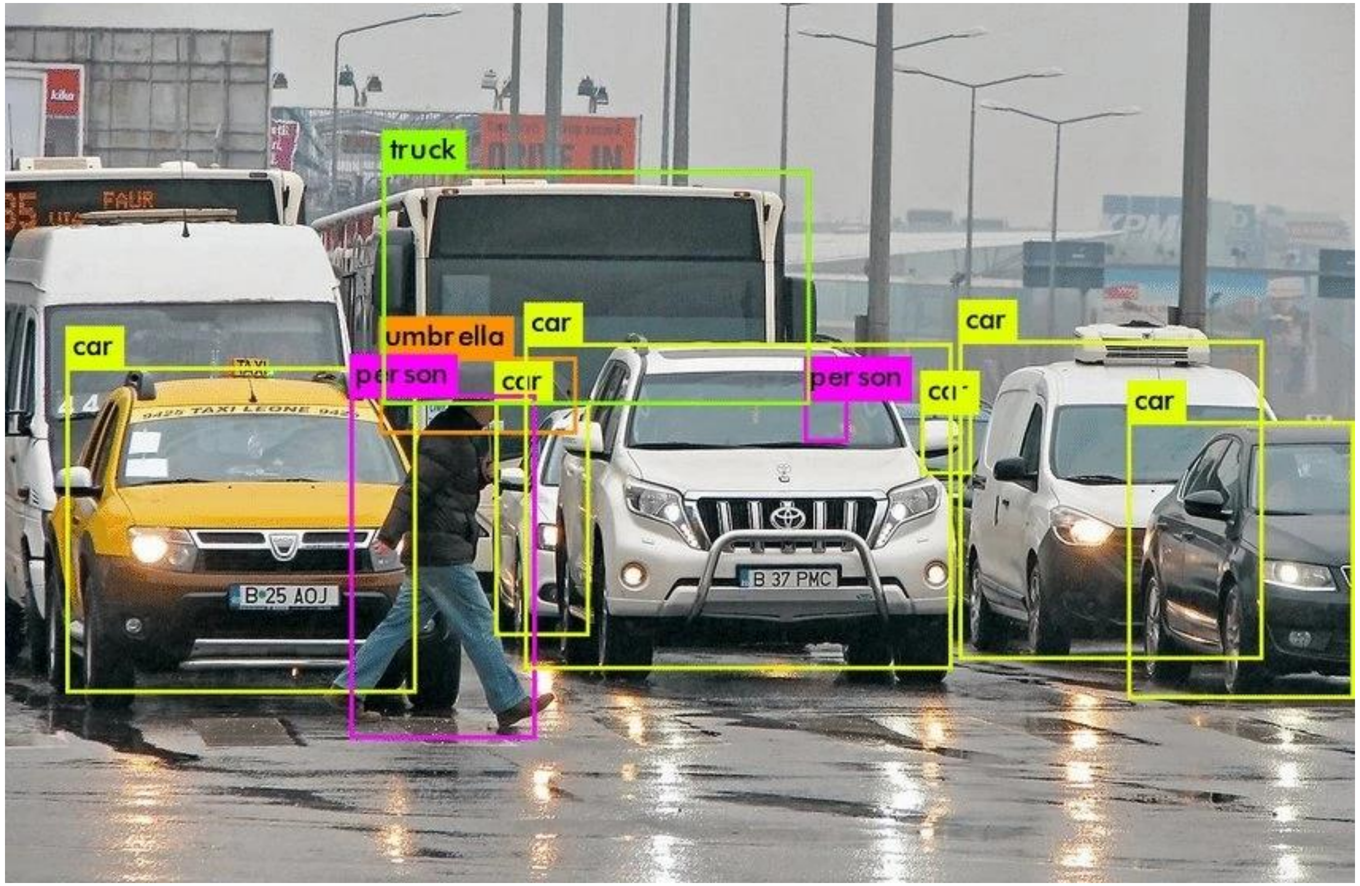
Applications

Image Recognition



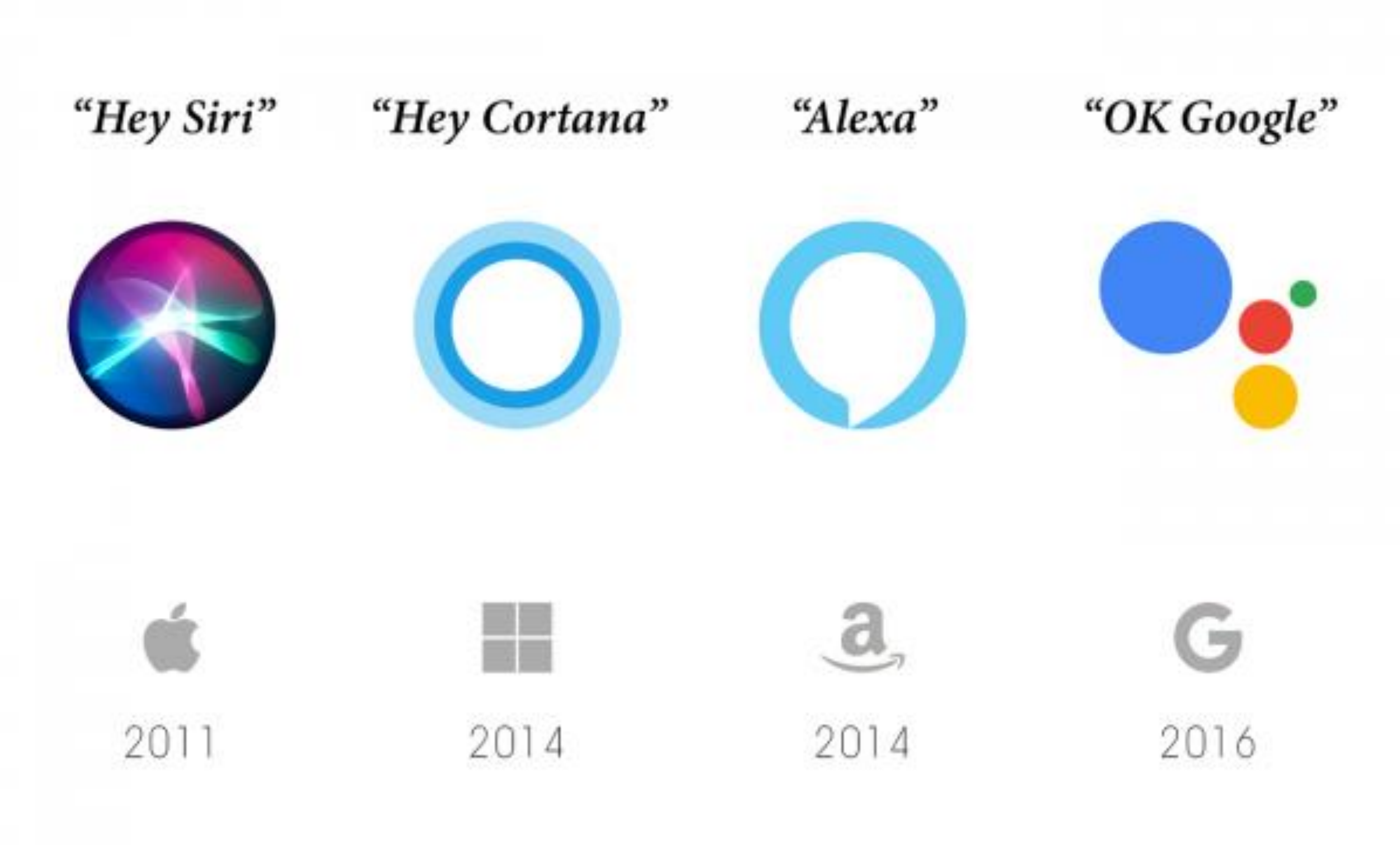
Applications

Image Recognition



Applications

Voice Assistants



Applications

Banking Industries

Personalized Customer Services

Automation & Productivity

Fraud Detection

Chatbots

StarShip Campus Delivery

Robot Graduation

a japanese university used remote-controlled robots and zoom to hold a virtual
duation

Robot Spot Dog

Section - 1.2

Why Data Science and Machine Learning?

The Global Data Science Market

Average Data scientist Salary in US = $ 102,312 per year

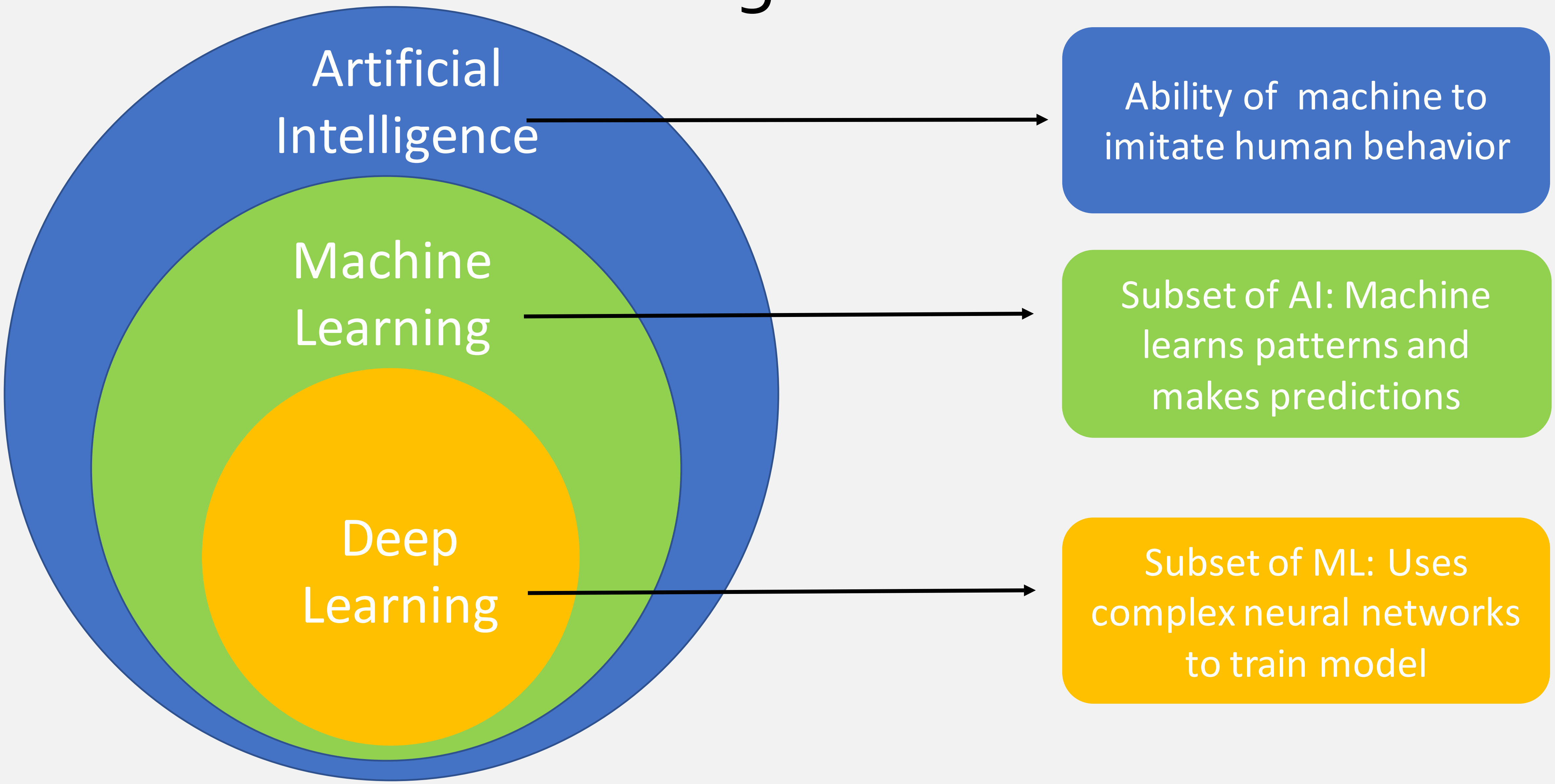
Ranks 3rd in US & 7th UK in the LinkedIn Emerging Job Market Report

Data science market is expected to grow from $37.9 billion in 2019 to $230.80 billion in 2026

Section - 1.3

What is Artificial Intelligence?

The Big Picture



History of A.I.

1950 | 1955 | 1964 | 1989 | 2000's | 2010's to now

1950

Alan Turing proposes a test for Machine Intelligence

1955

John McCarthy coins the term "Artificial Intelligence"

1964

MIT Artificial Intelligence Laboratory creates a chatbot named "Eliza"

1989

CMU pioneers the use of neural networks to steer autonomous vehicles

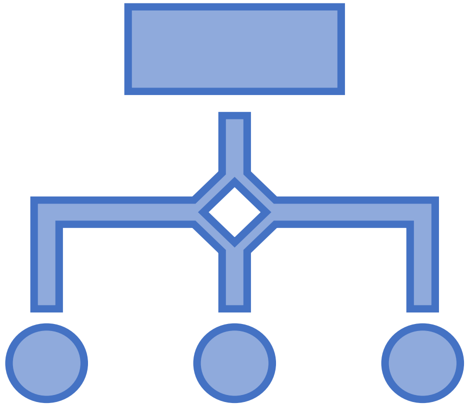
2000's

The Era of Big Data powering algorithms to learn more patterns

2010's - now

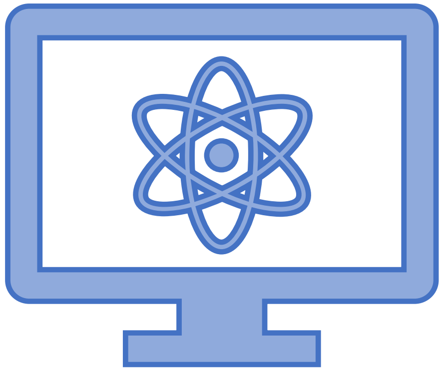
Era of deep learning from Facial recognition to autonomous cars

Types of AI

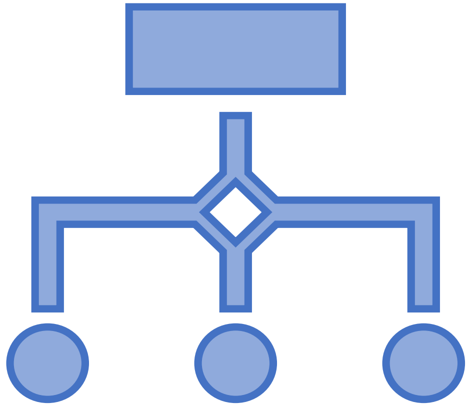


Logic & Rule based AI

Pattern based AI



Hybrid AI

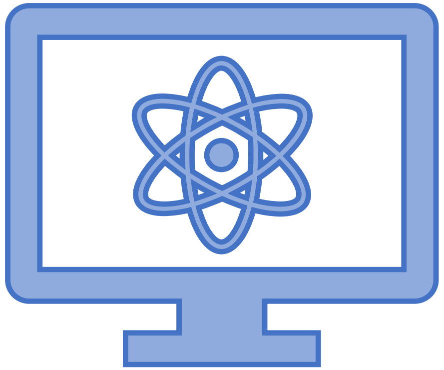


Logic & Rule based AI

- Represents processes or systems using logic or rules
- Small amount of data, simple and straight forward rules
- Rules-based systems are great for repetitive processes that require little-to-no human decision making
- Use cases : Fault Analysis, Search & Retrieval, traditional Chatbots

Pattern based AI

- Algorithm finds pattern in data and learns on his own
- Works with huge quantities of data
- Finds hidden relationships or patterns
- Makes intelligent suggestions
- Use cases : Facial recognition & Product recommendation



Hybrid AI

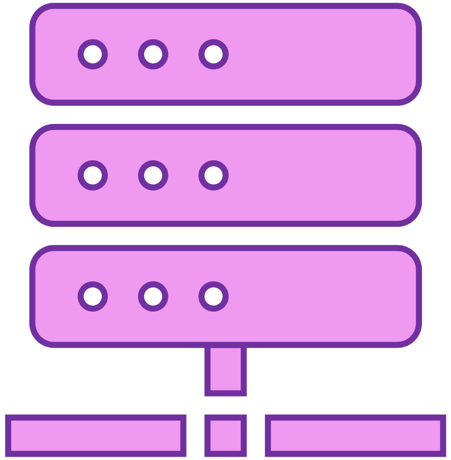
- Many successful AI systems are hybrid
- Combination of rule-based, learning based and human intelligence
- Use cases :
  - ❑ Self-driving cars
  - ❑ Rescue Drones

Section - 1.4

What is Machine Learning

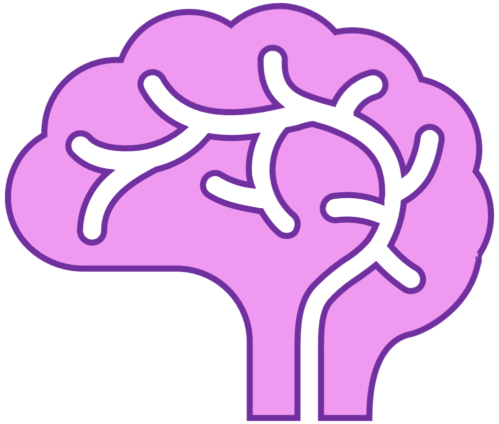
A machine learning algorithm is an algorithm that learns from data to perform tasks improve the performance overtime without being explicitly programmed

Machine Learning



Process huge
volume of data

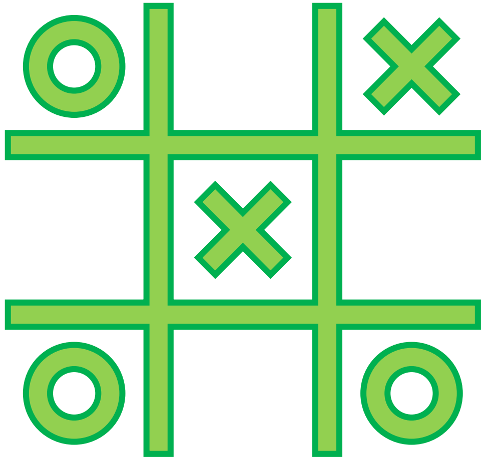
Identifies
hidden patterns



Makes insight
based decisions

Types of Machine Learning

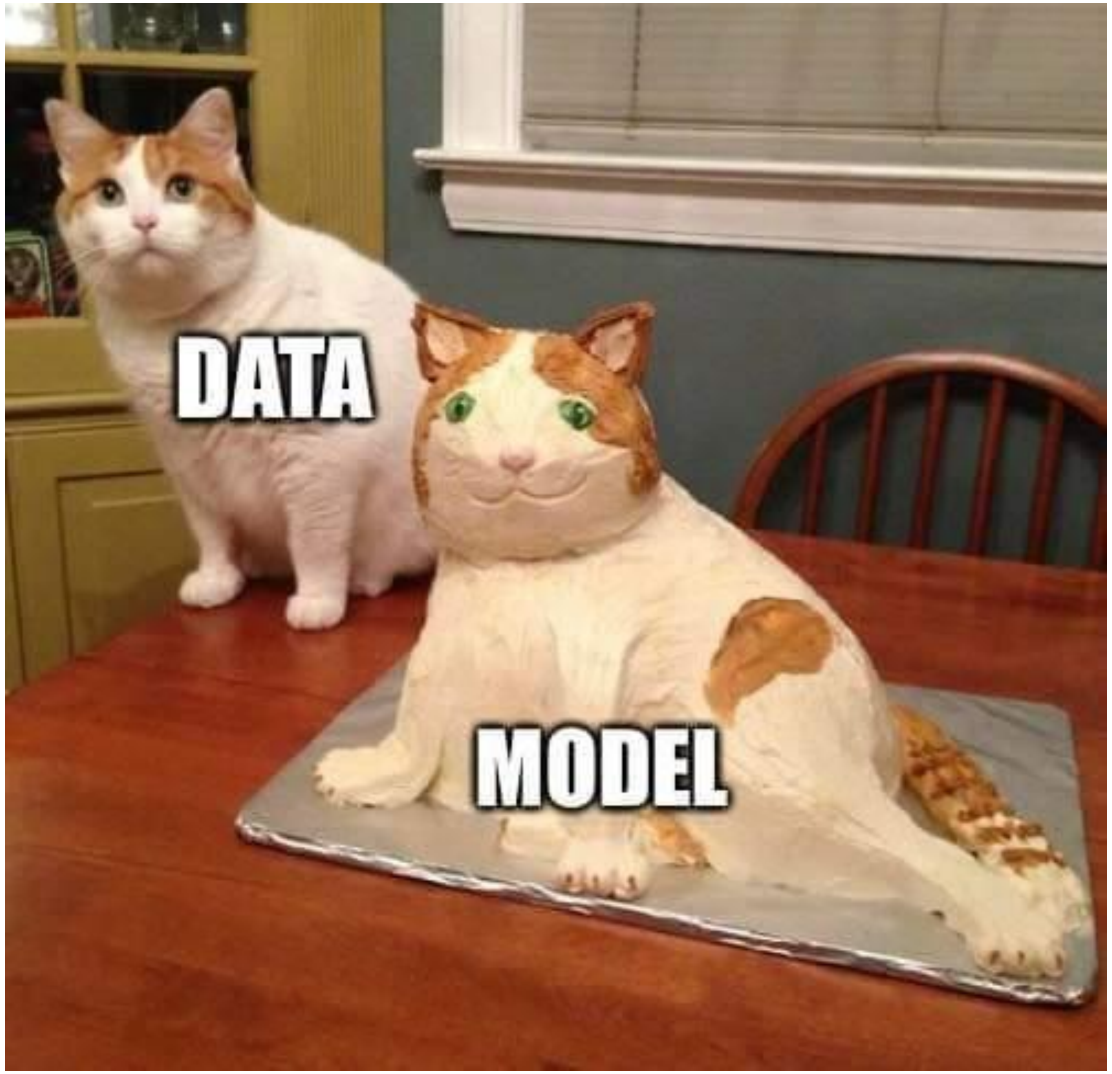
Supervised
Learning



Unsupervised
Learning

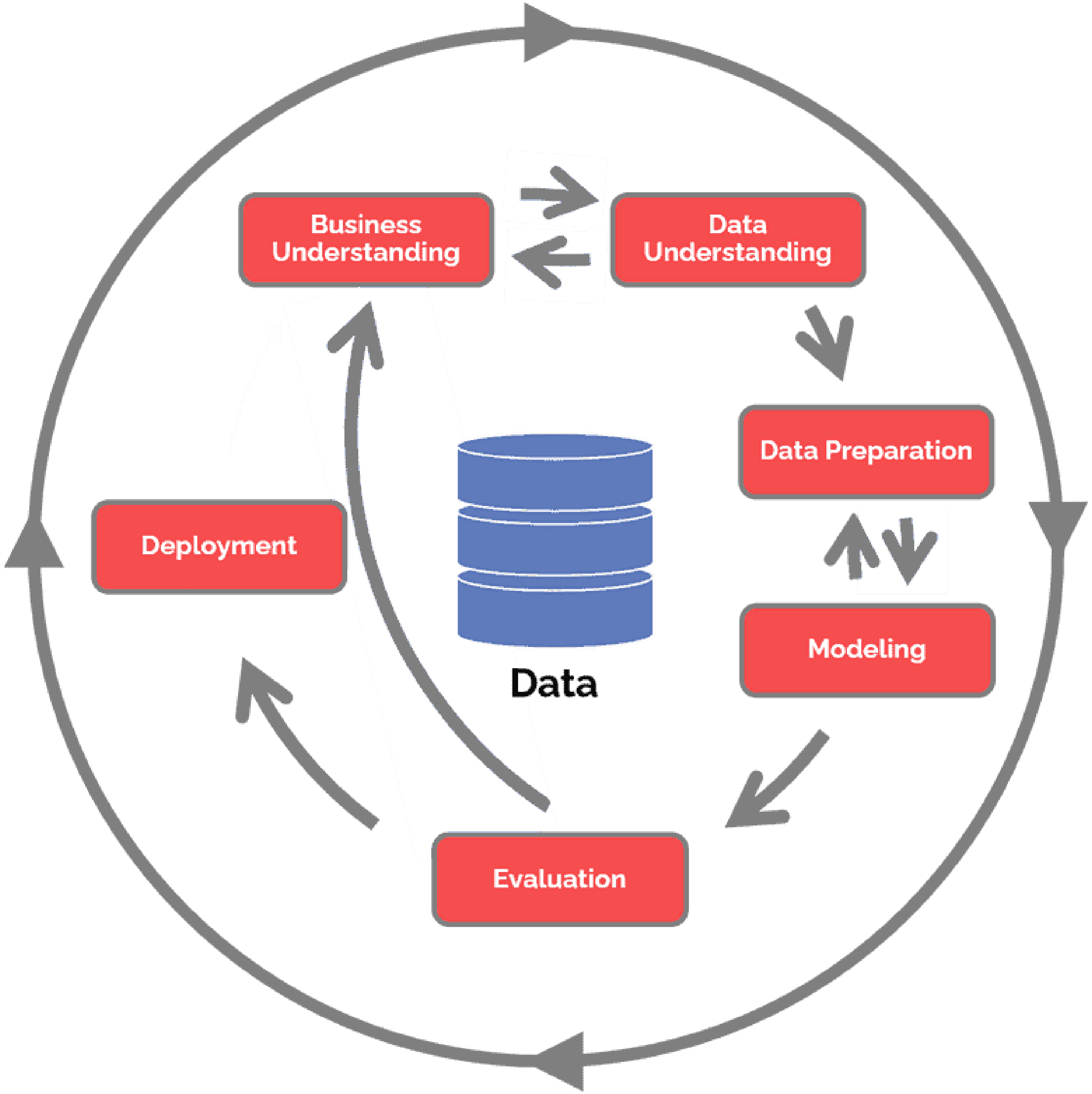
Reinforcement
Learning

Machine Learning



CRISP-DM

CRoss **I**ndustry **S**tandard **P**rocess for **D**ata **M**ining



Section - 2

Environment setup

Tools

Python 3.9

ANACONDA

Section - 2.2

Python

Libraries

NumPy

pandas

Errors

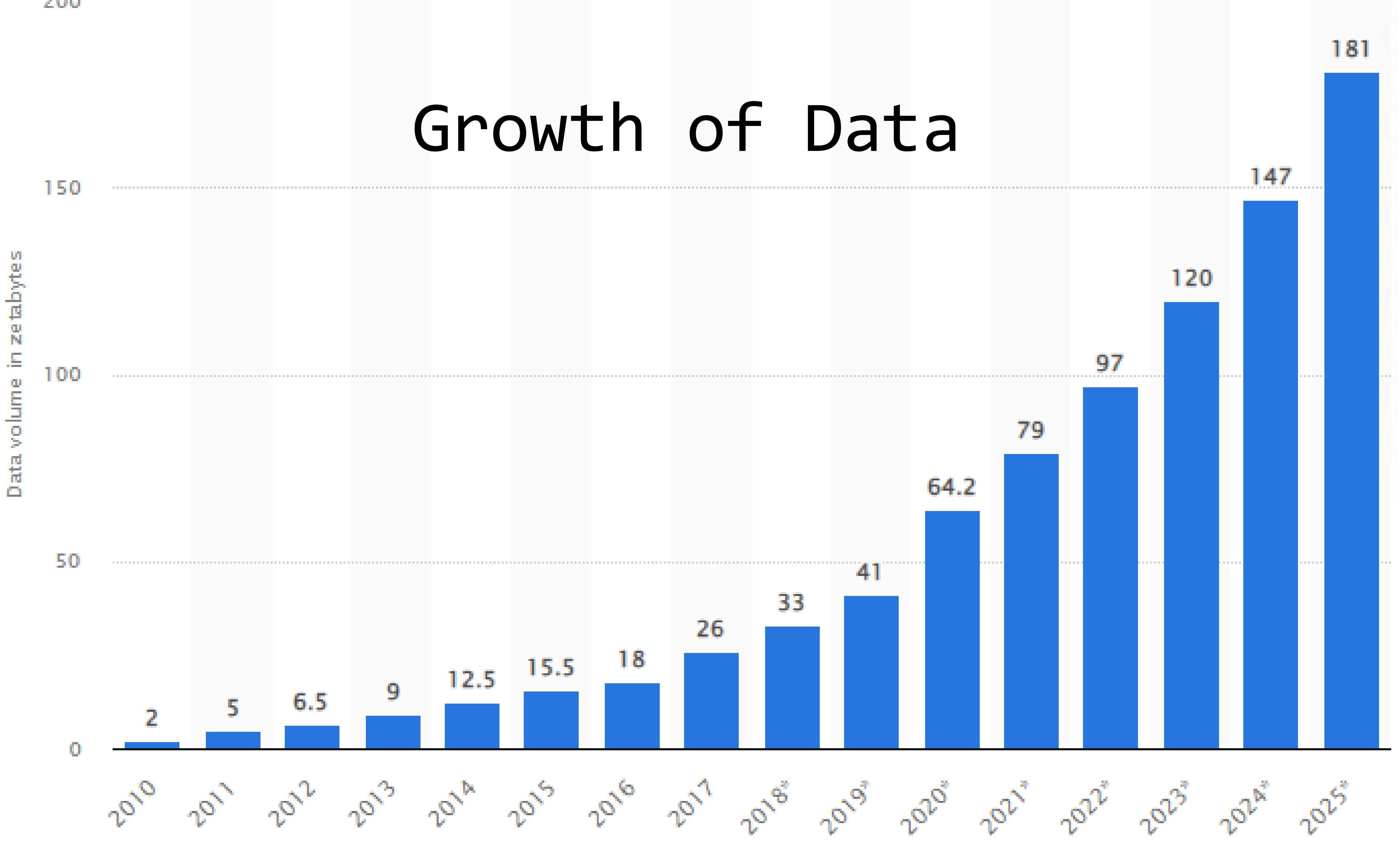

Wow, a different error message... Finally some progress!
ENERGY
ENERGY
ENERGY
AGENT-X COMICS
WWW.AGENT-X.COM.AU

Section - 3

The Art of Data Visualization

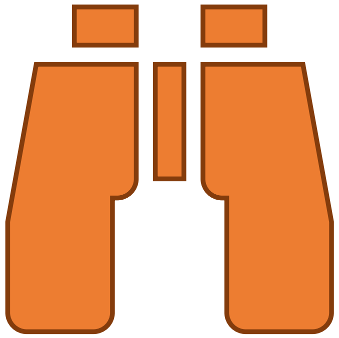
Data visualization is the process of graphical representation of data in the form of charts, infographics, statistical graphs and such for easily understanding the patterns.

Growth of Data

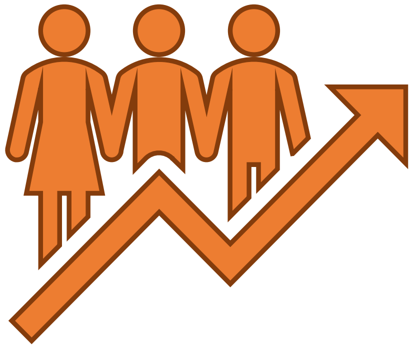


Why Data Viz. is important?

Understand & Analyse the data



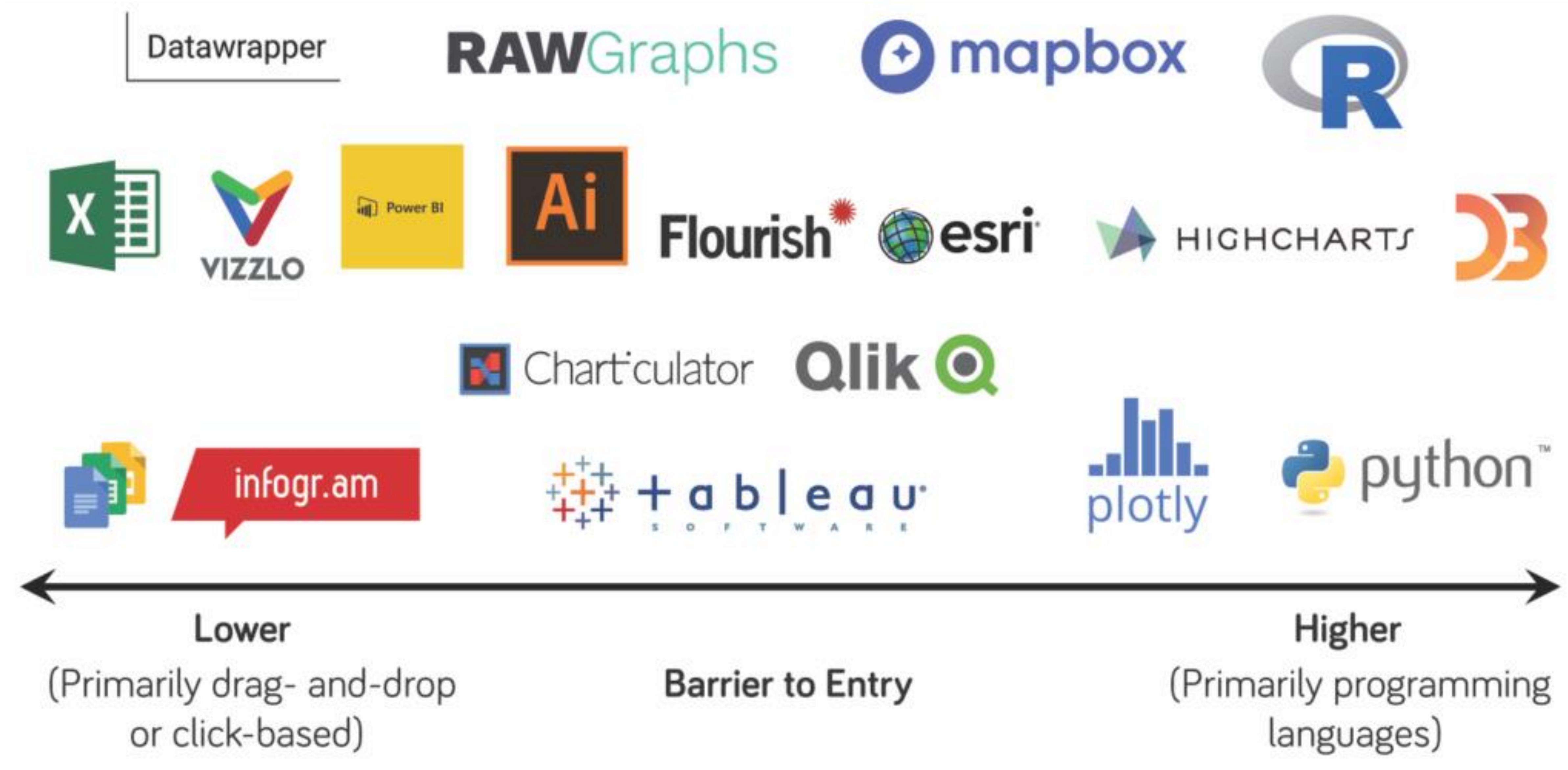
Find trends and patterns quicker



Derive actionable insights

Ability to tell a story

Data Viz. tools



Python Data Viz. Libraries

Visualisation Library

matplotlib

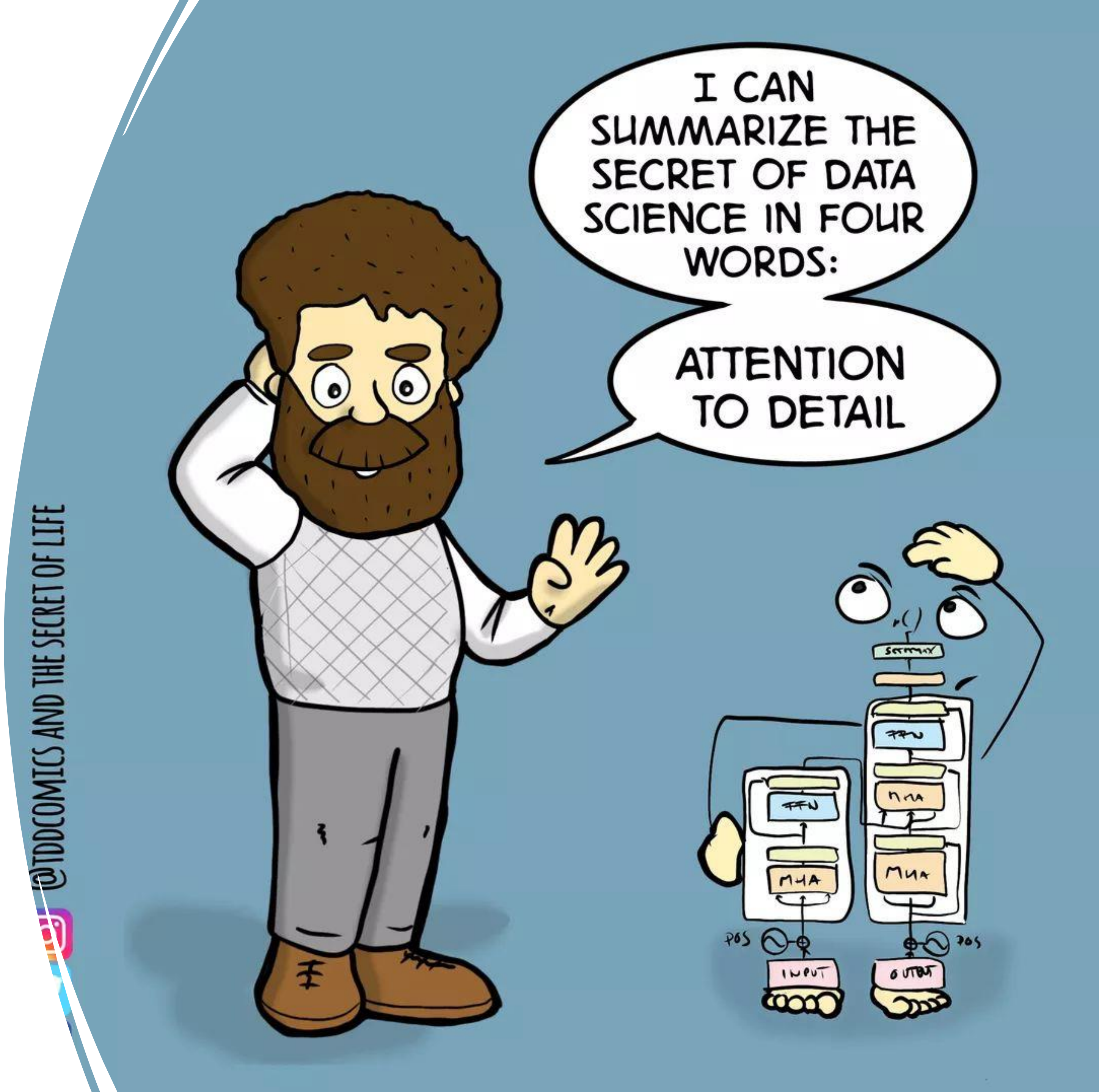
ggplot2

seaborn

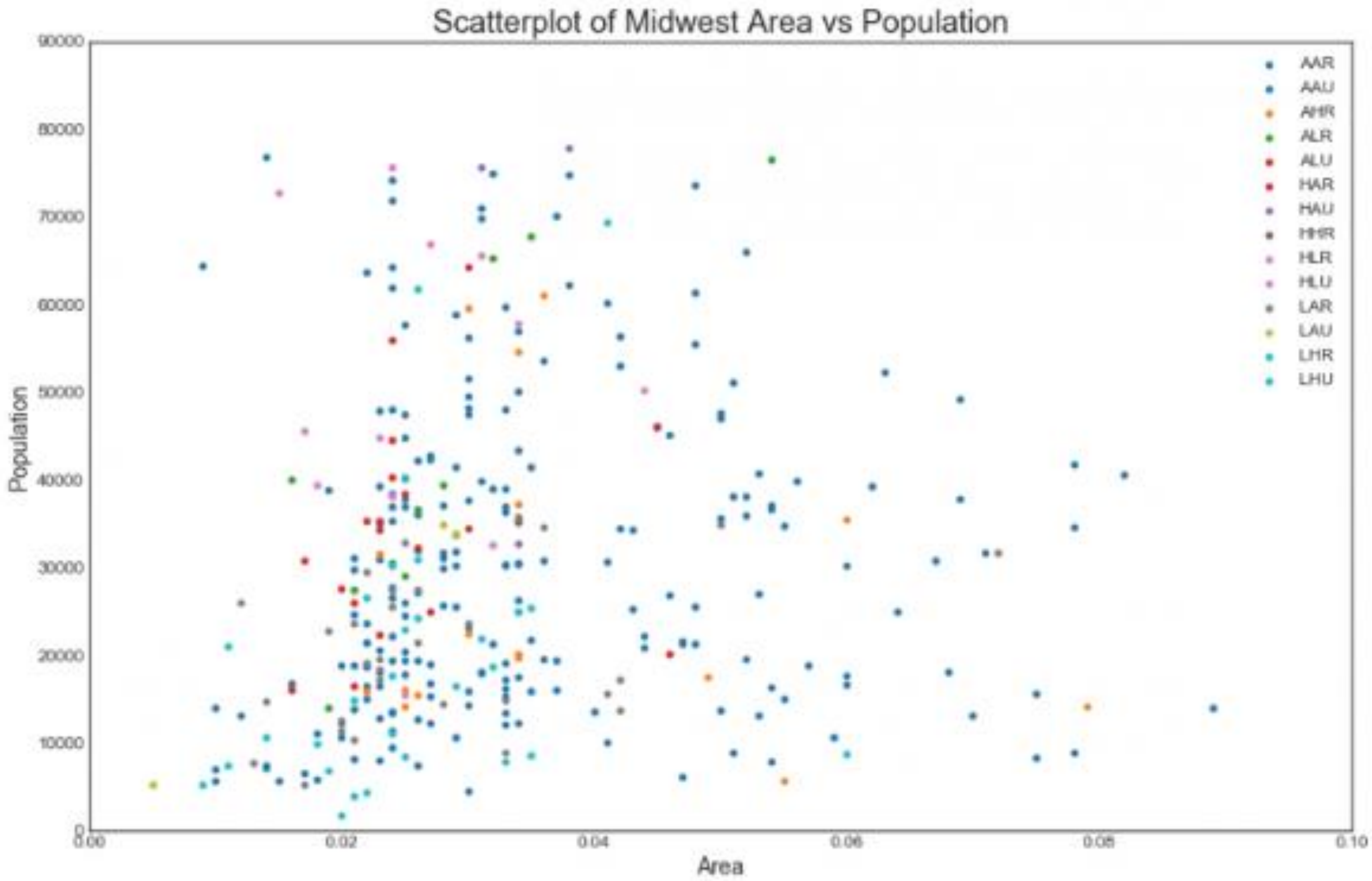
bokeh

plotly

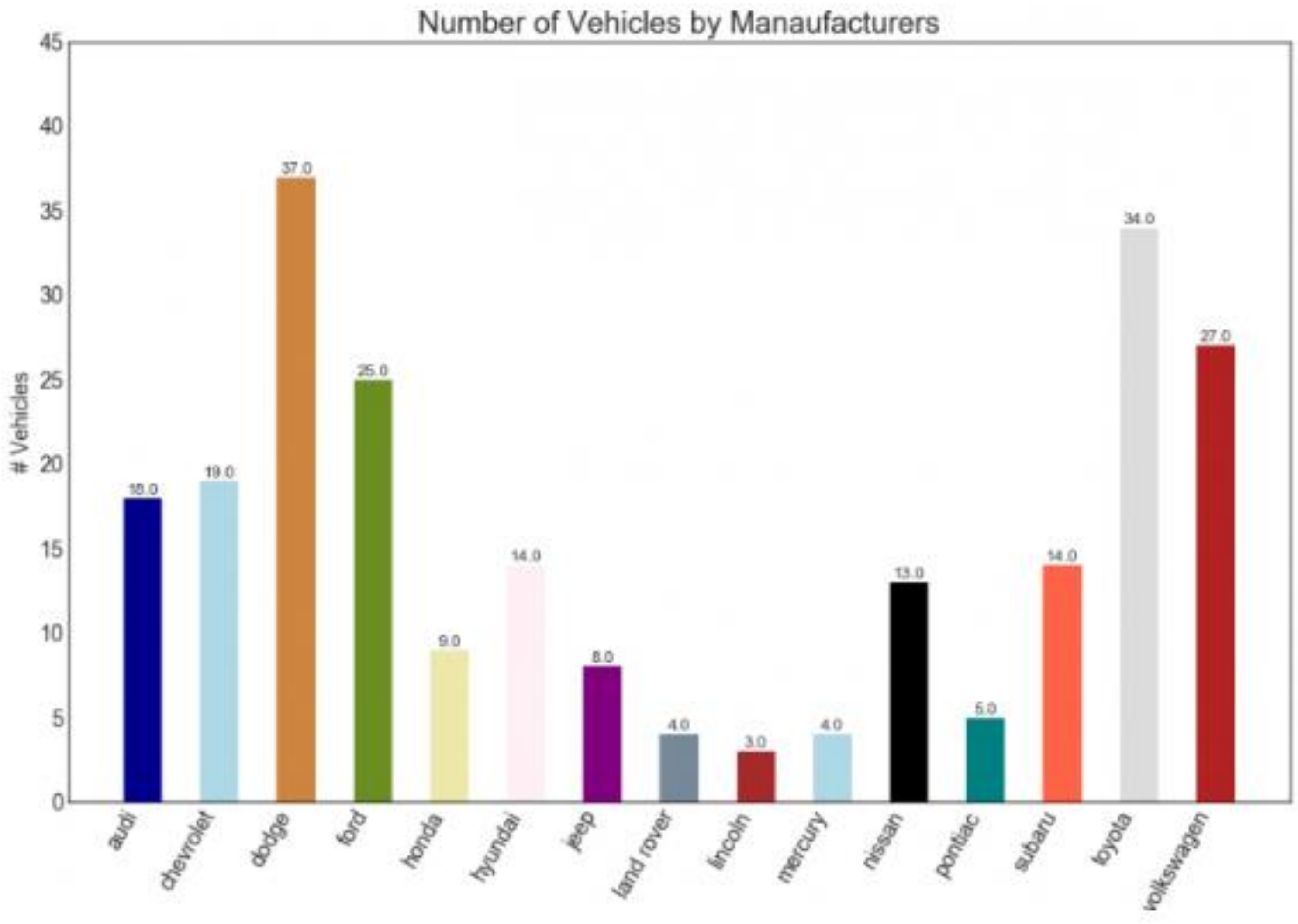
Data Visualization



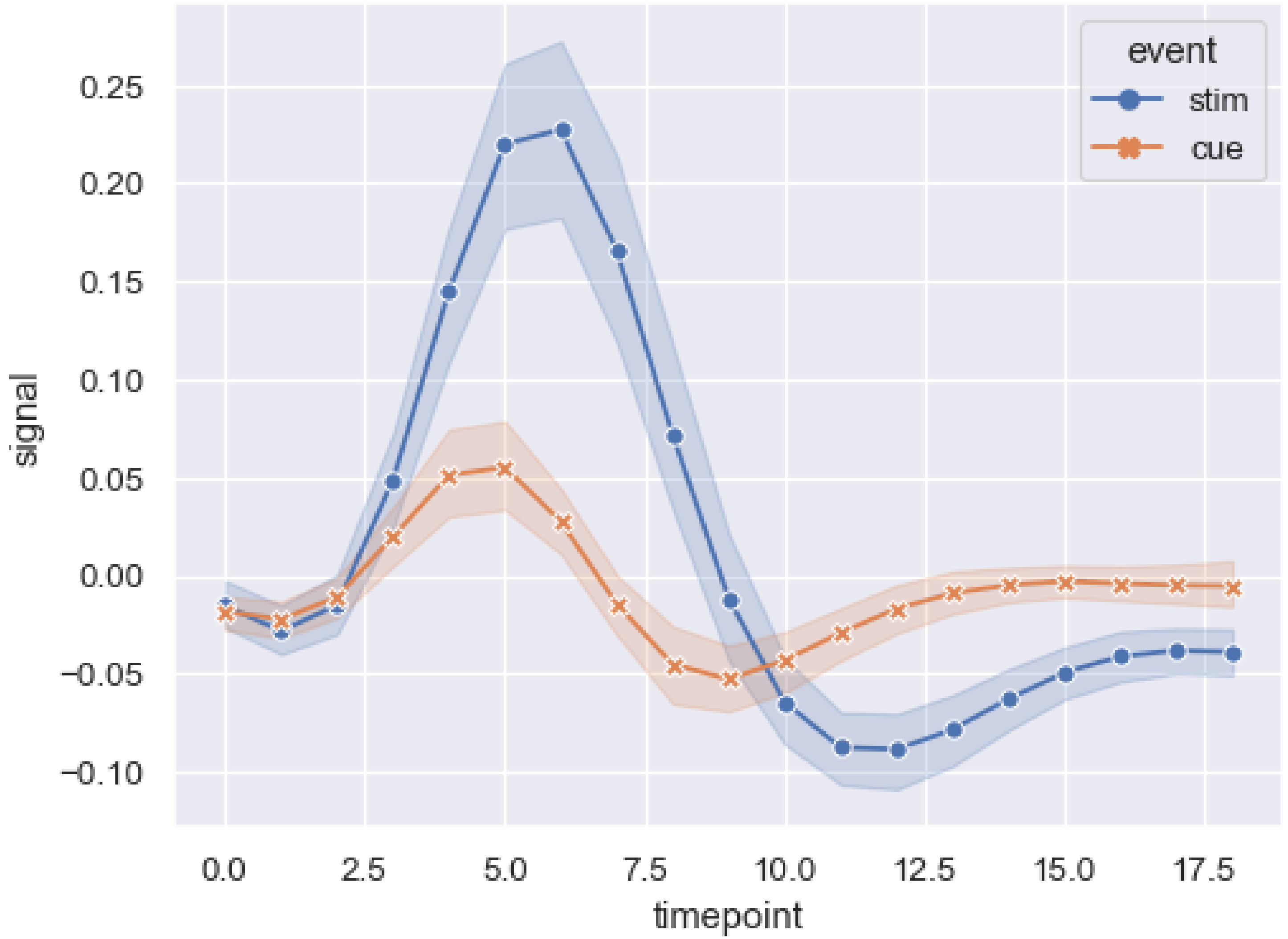
Scatter Plot



Bar plot

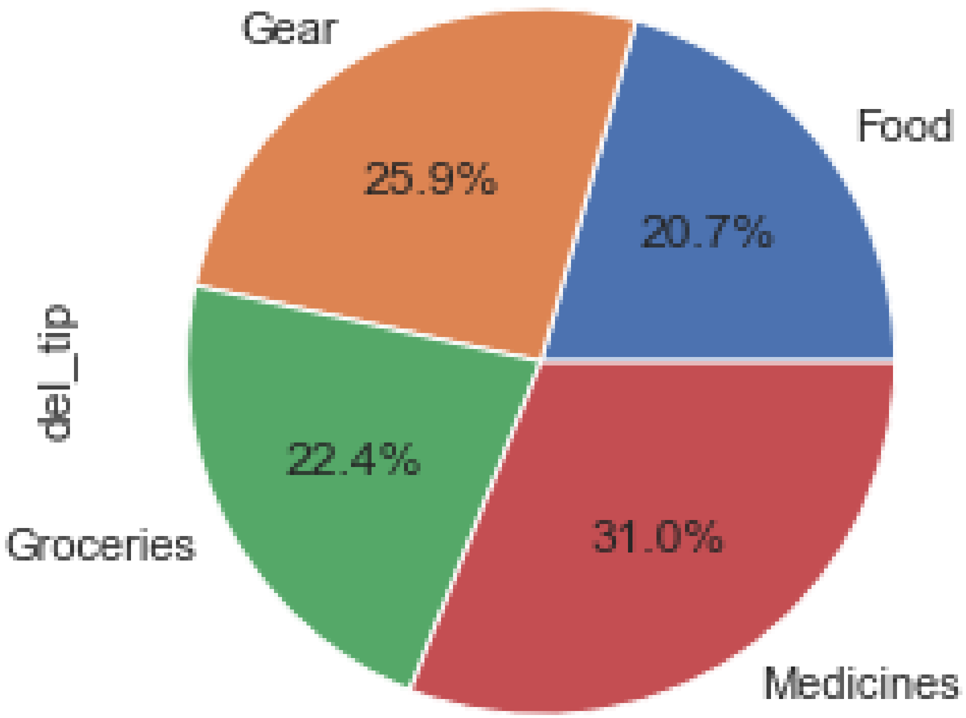

Number of Vehicles by Manaufacturers
Vehicles
0
5
10
15
20
25
30
35
40
45
18.0
19.0
37.0
25.0
9.0
14.0
8.0
4.0
3.0
4.0
13.0
5.0
14.0
34.0
27.0
audi
chevrolet
dodge
ford
honda
hyundai
jeep
land rover
lincoln
mercury
nissan
pontiac
subaru
toyota
volkswagen

Line plot

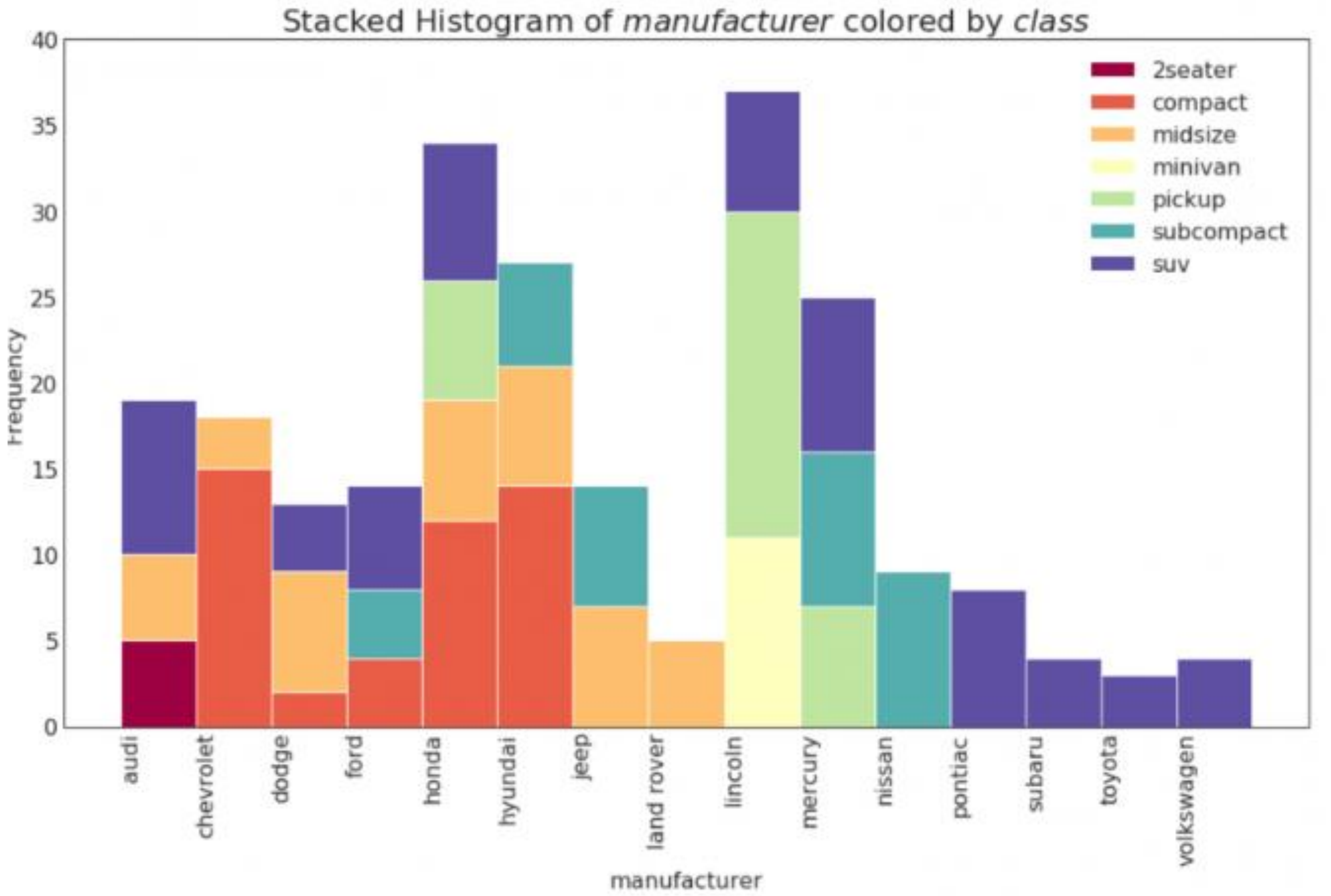


event
stim
cue
0.25
0.20
0.15
0.10
0.05
0.00
−0.05
−0.10
signal
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
timepoint

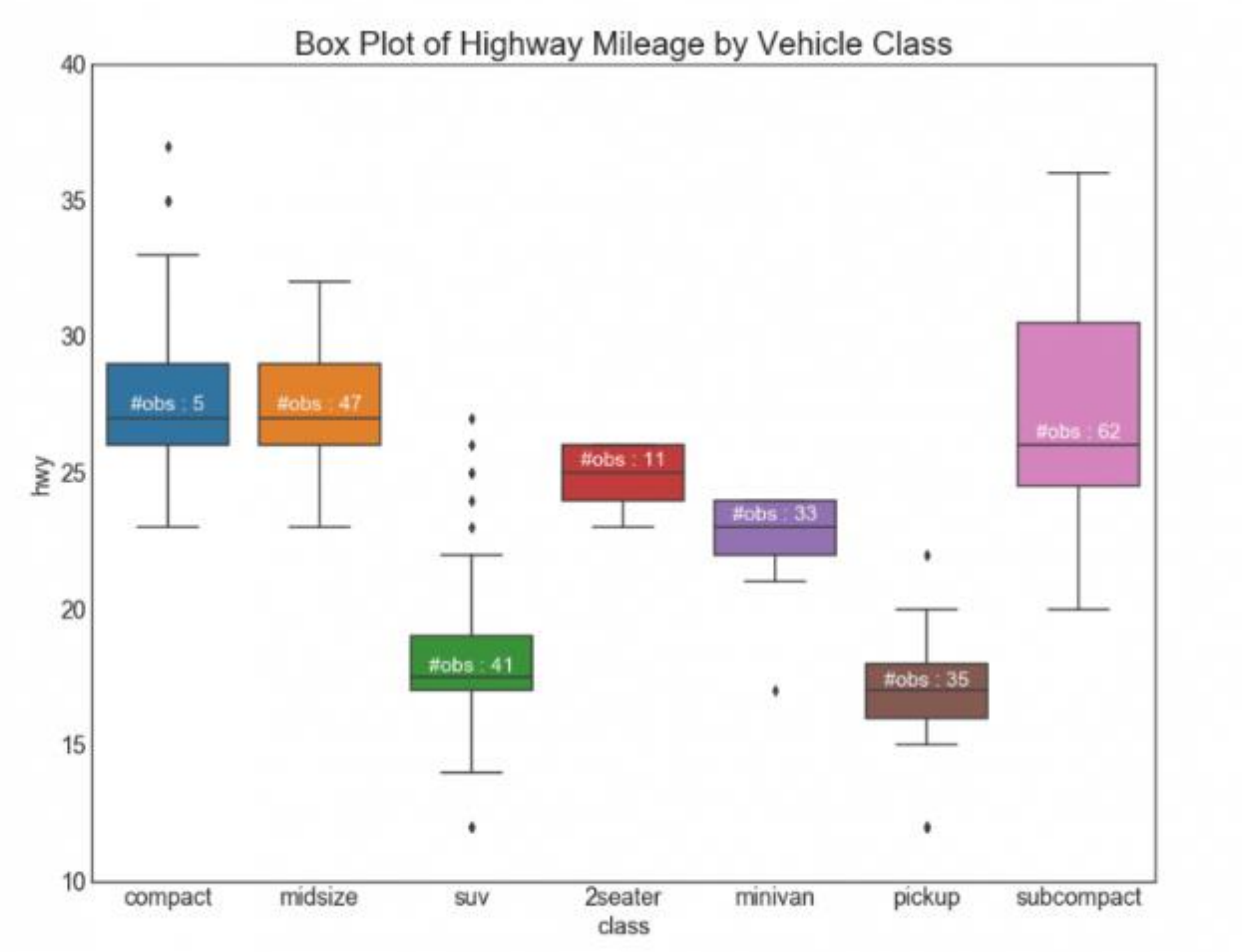
Pie chart



Histogram plot

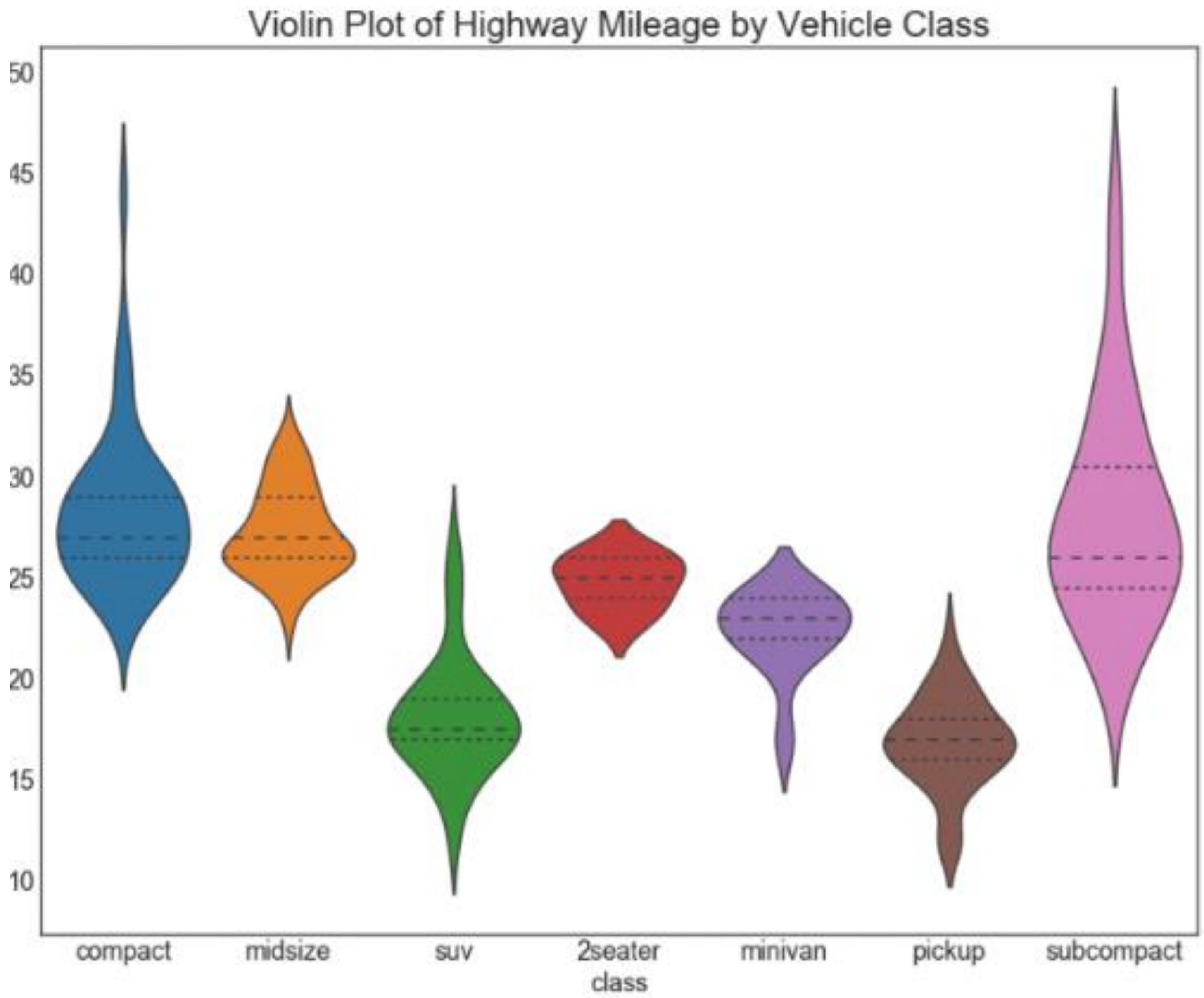


Box plot



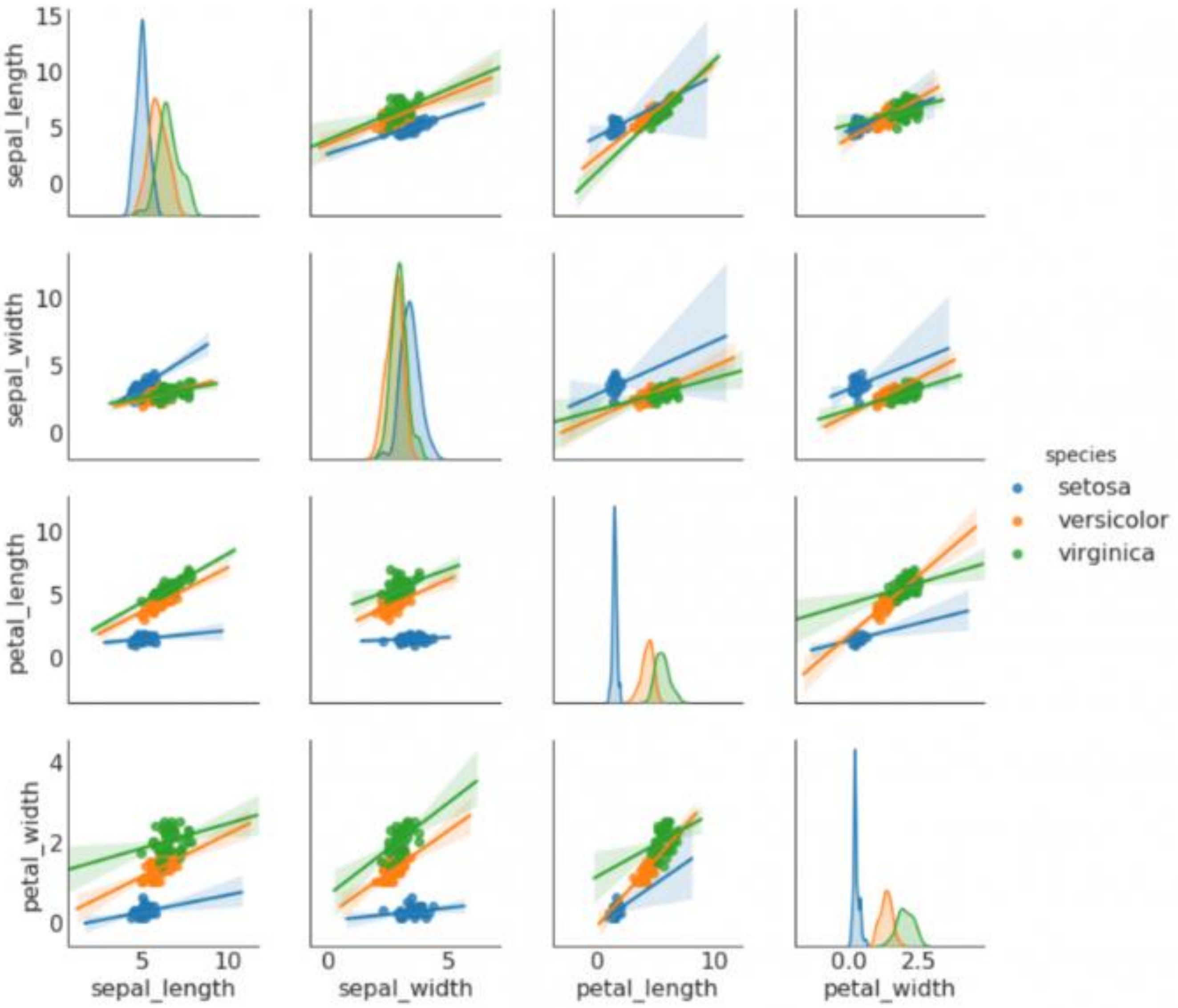
Box Plot of Highway Mileage by Vehicle Class
hwy
class
40
35
30
25
20
15
10
compact
midsize
suv
2seater
minivan
pickup
subcompact
#obs : 5
#obs : 47
#obs : 41
#obs : 11
#obs : 33
#obs : 35
#obs : 62

Violin plot

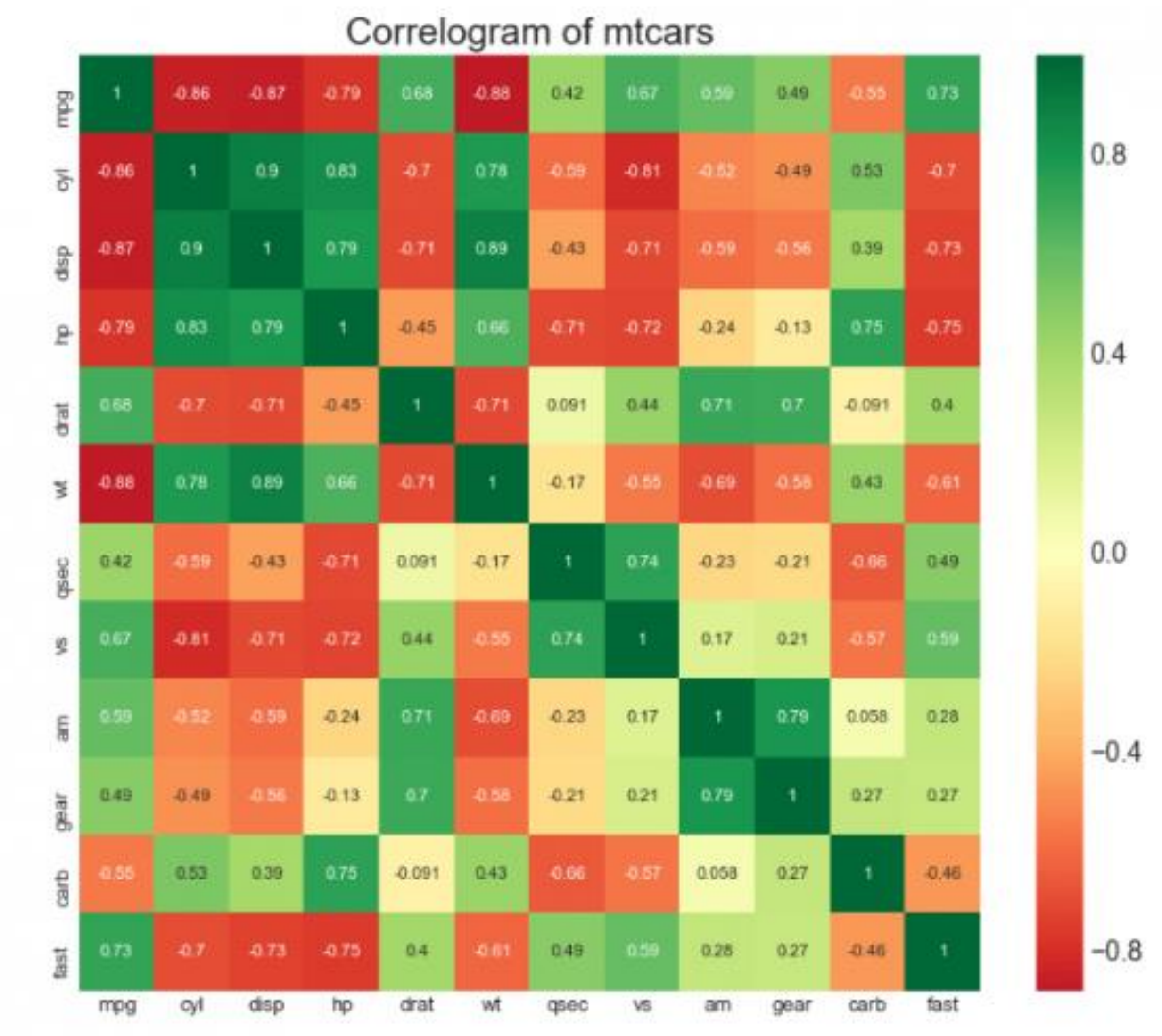


Violin Plot of Highway Mileage by Vehicle Class
50
45
40
35
30
25
20
15
10
compact
midsize
suv
2seater
minivan
pickup
subcompact
class

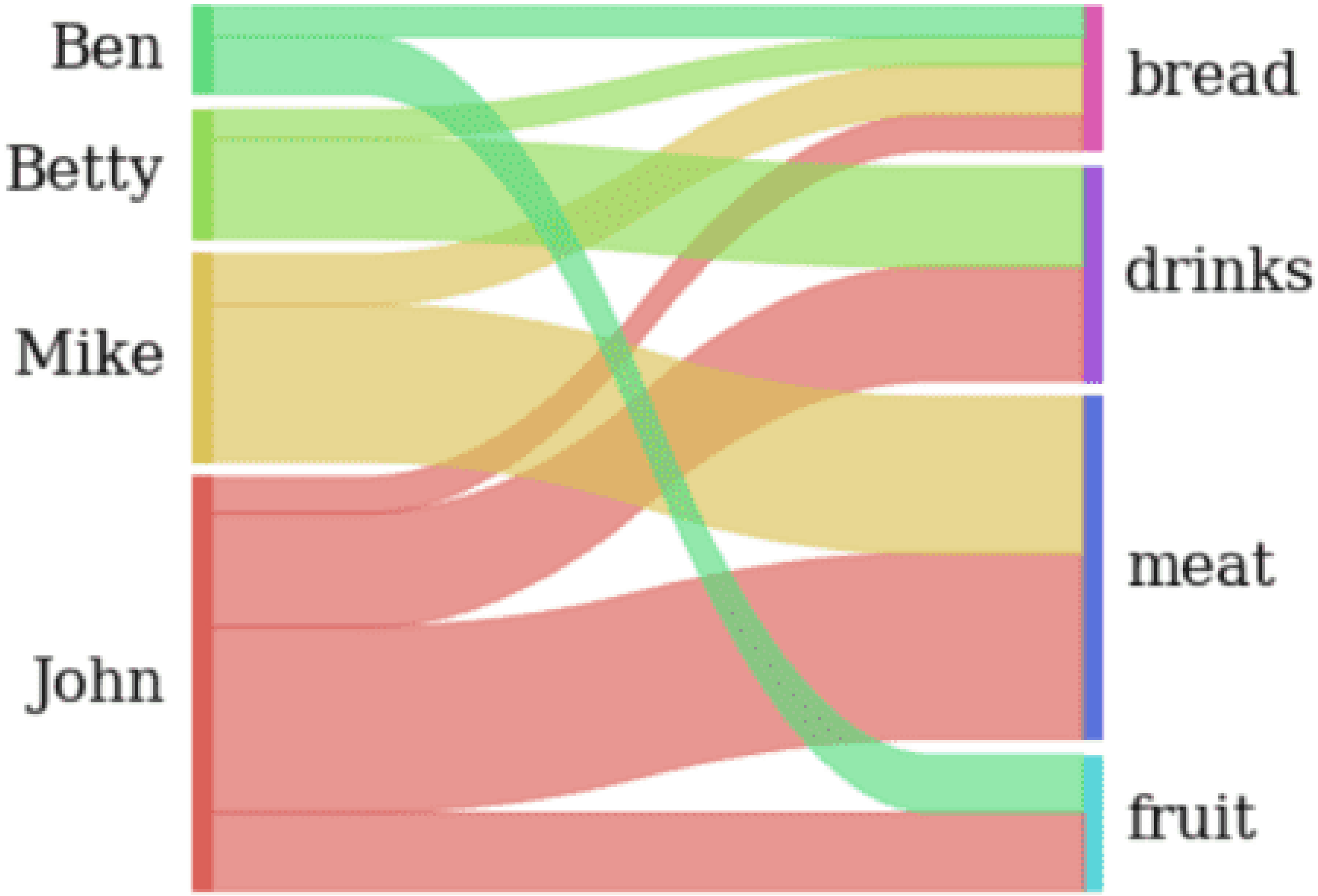
Pairwise plot



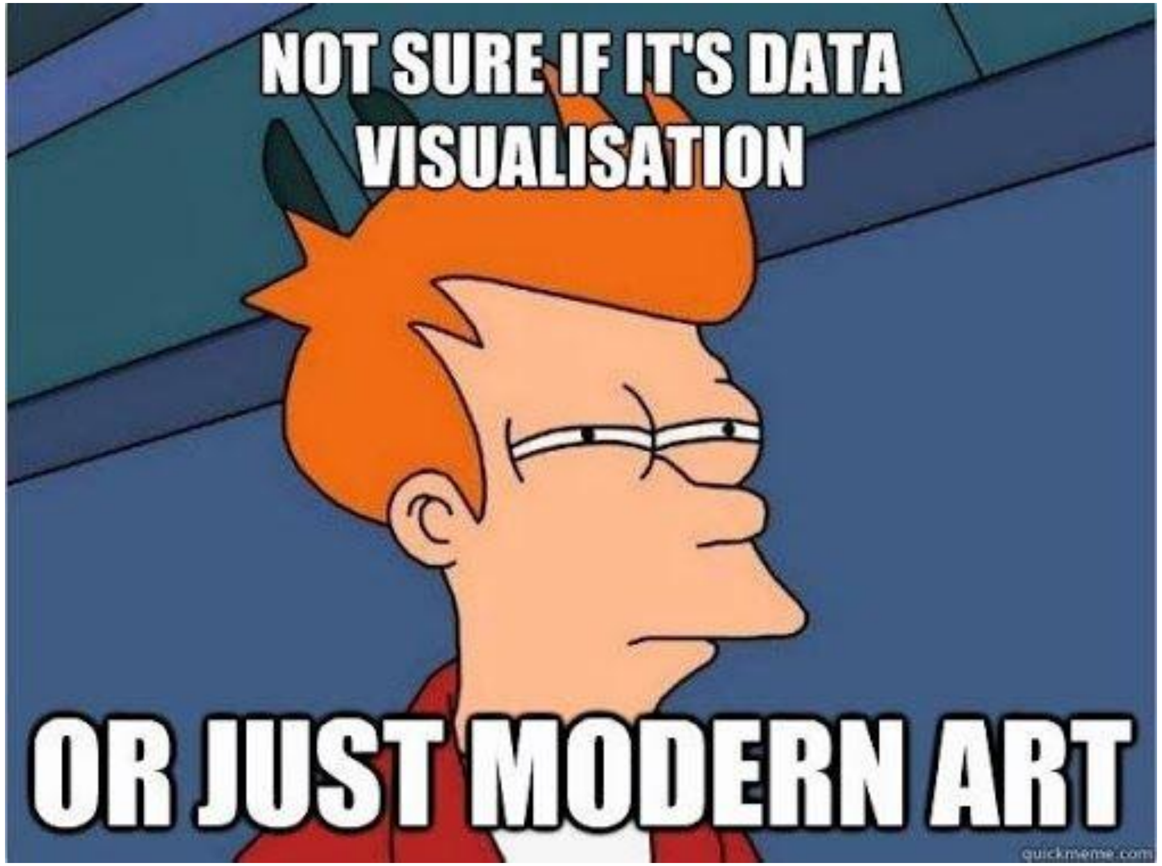
Heat Map


Correlogram of mtcars
mpg
cyl
disp
hp
drat
wt
qsec
vs
am
gear
carb
fast
0.8
0.4
0.0
−0.4
−0.8

Sankey

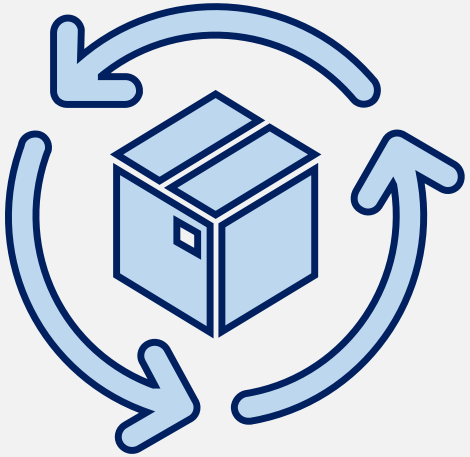


Data Visualization



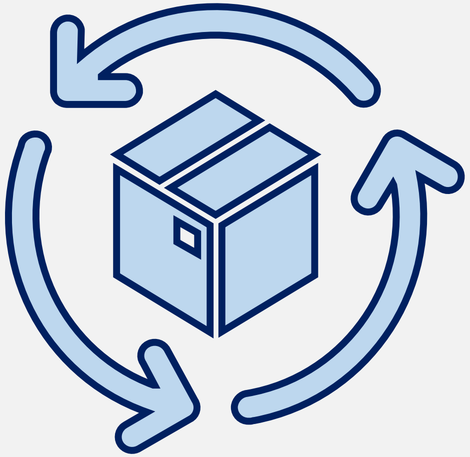
Section - 4

Fundamental Techniques of Data Preparation & Feature Engineering

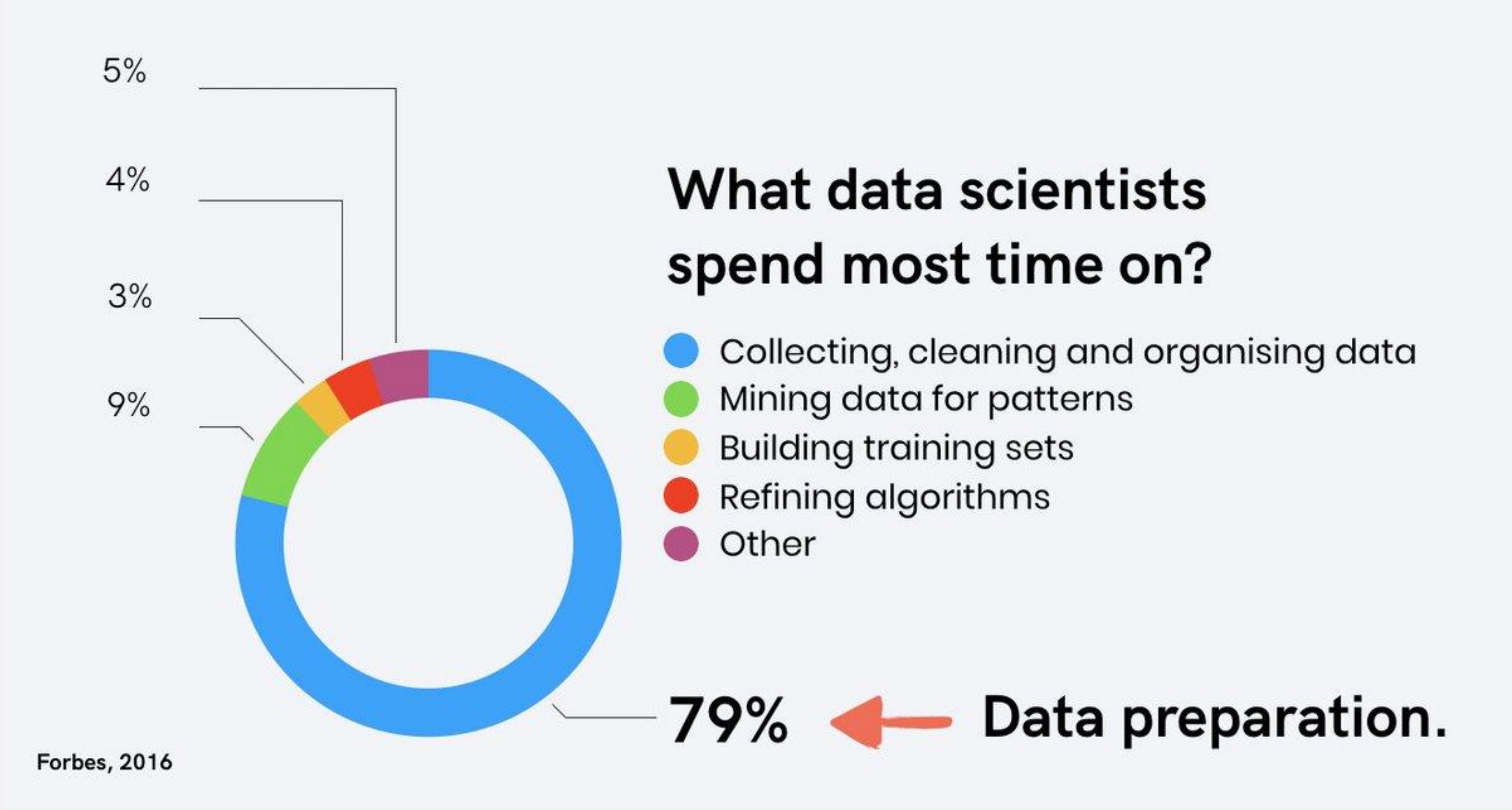


Benefits of Data Preparation

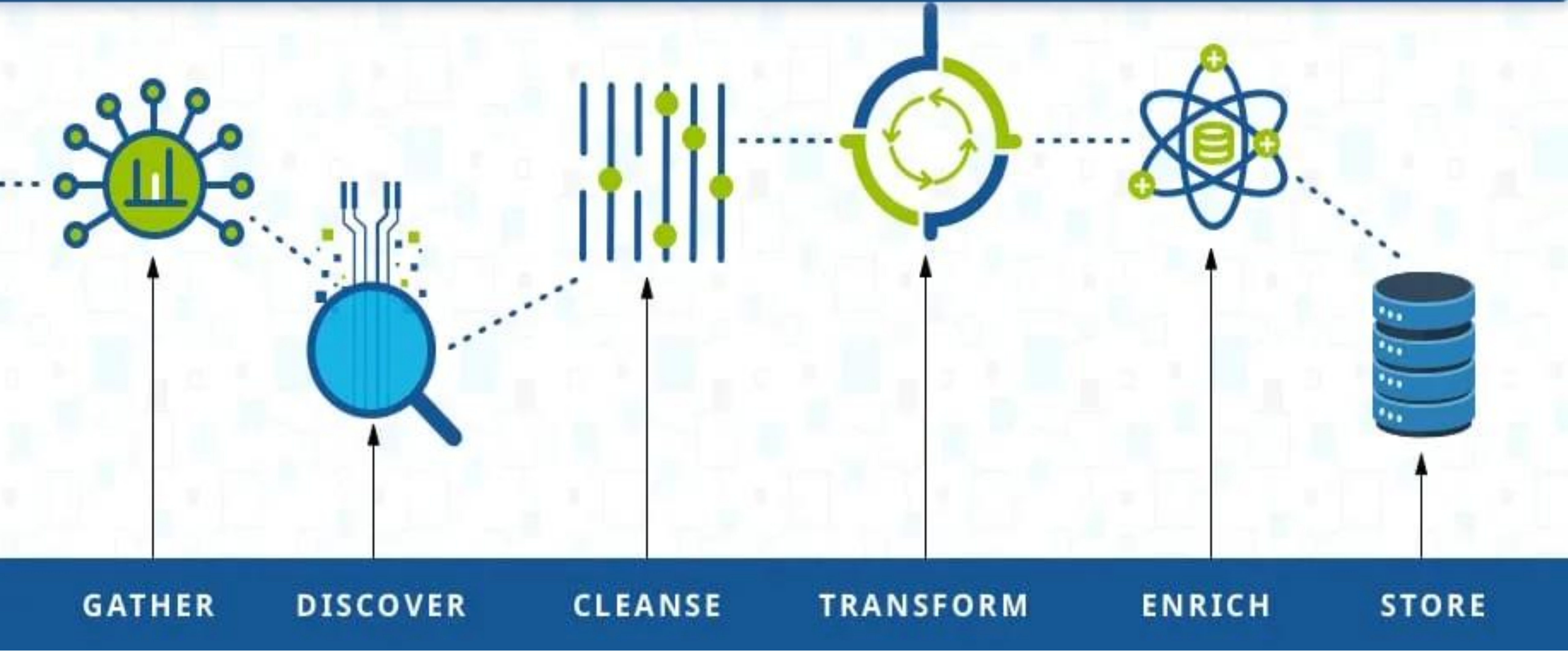
- ❑ Produces reliable results
- ❑ Enables more informed decision making
- ❑ Reusability
- ❑ Detect and fix errors quickly
- ❑ Produces top quality data

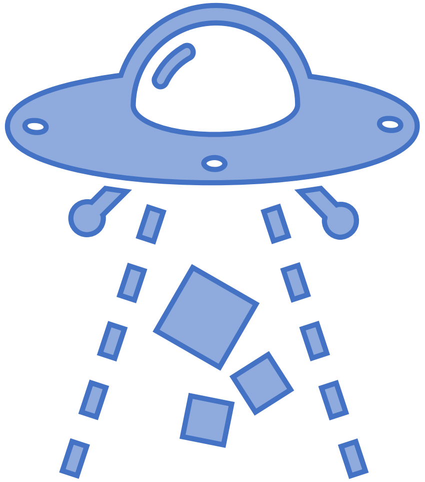


Data Preparation

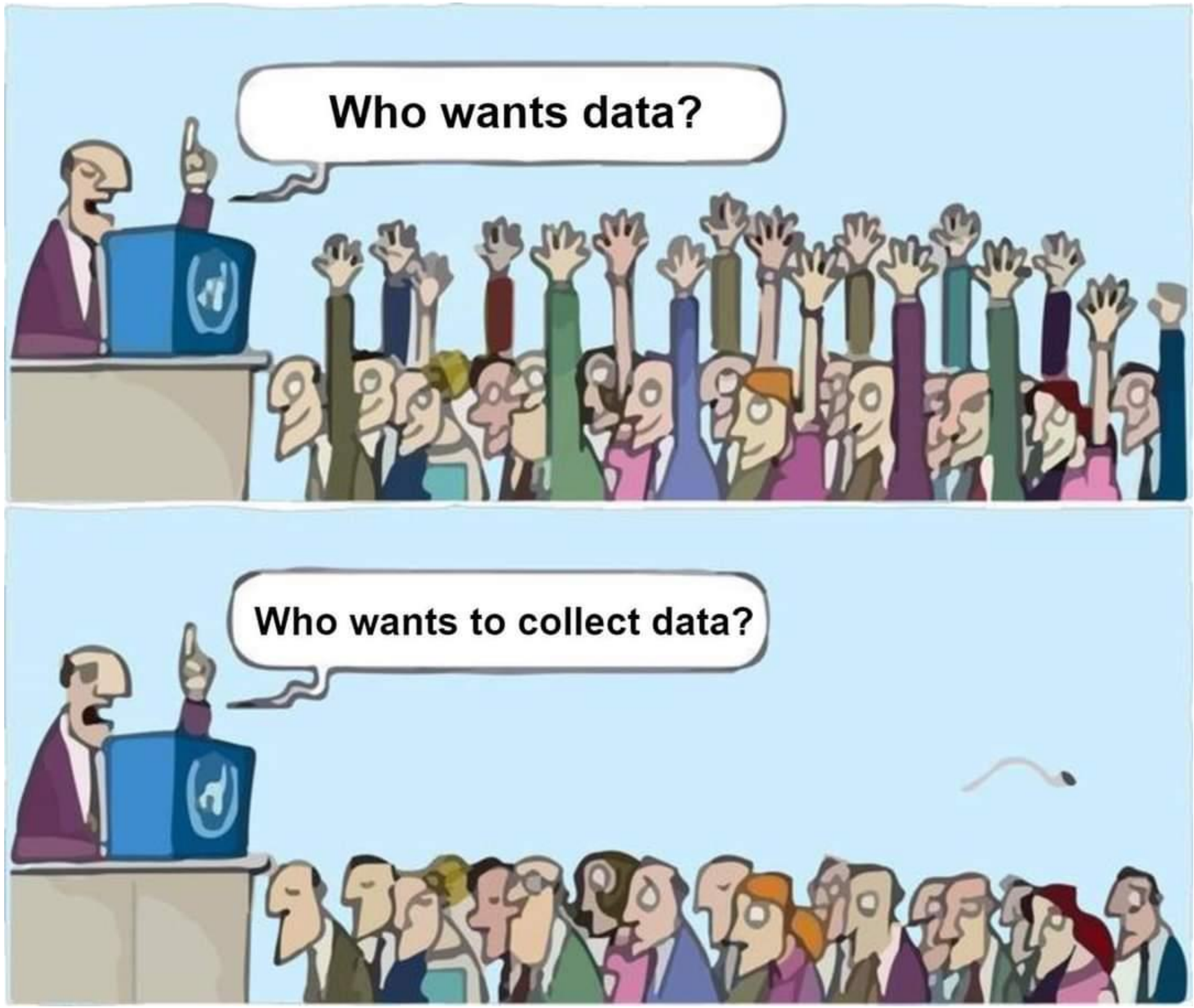


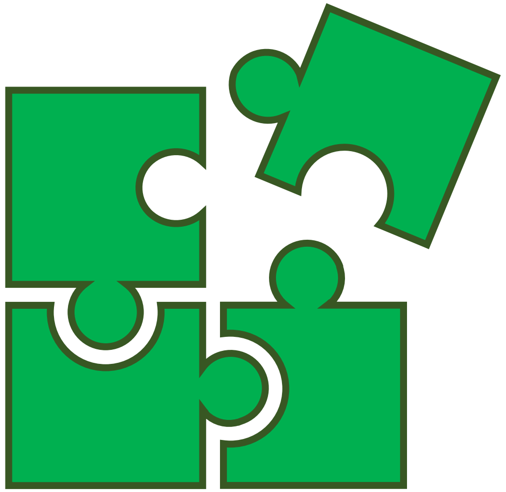
DATA PREPARATION





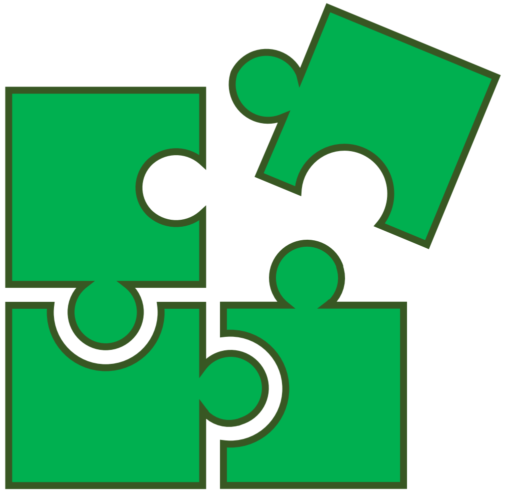
Data Collection



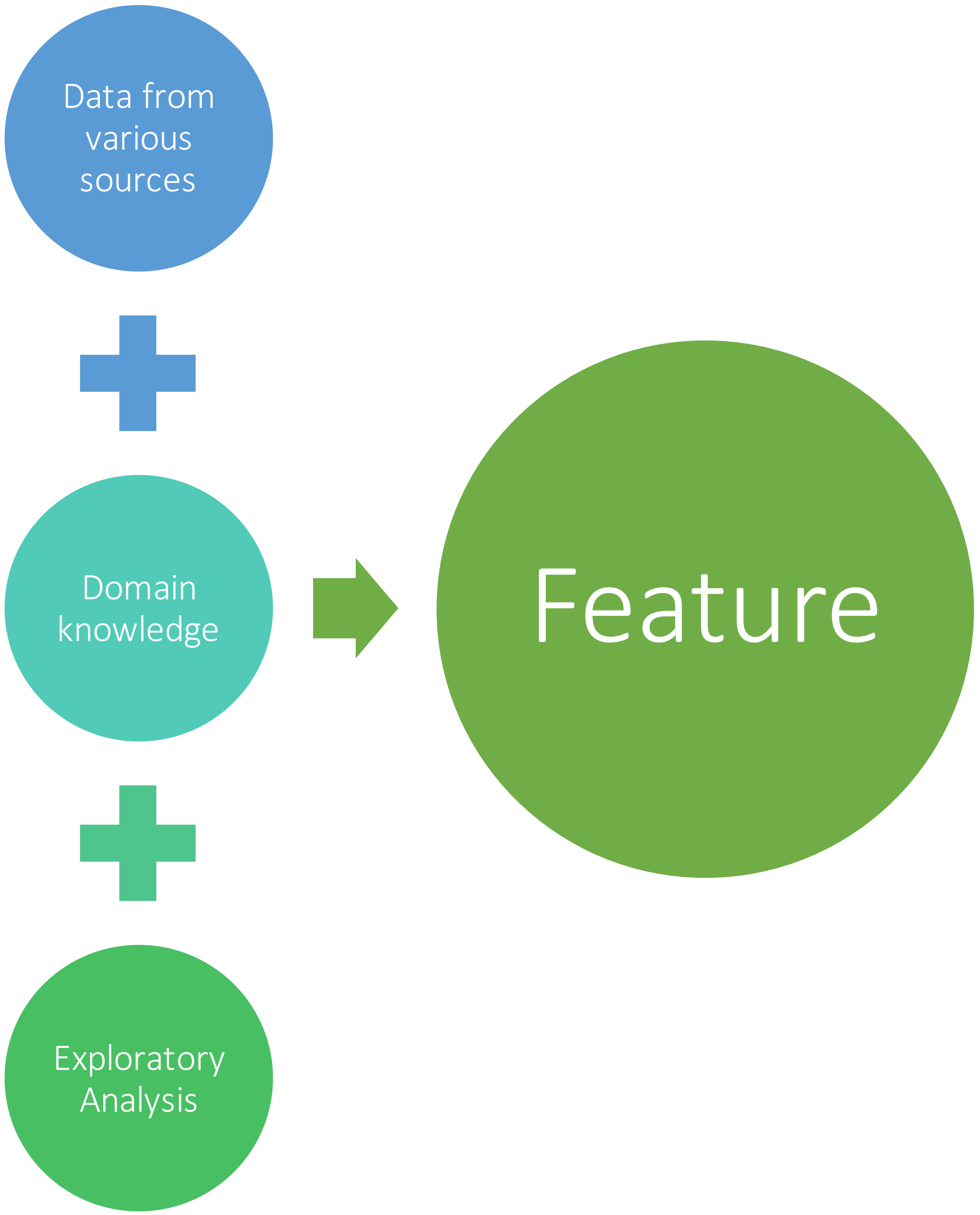


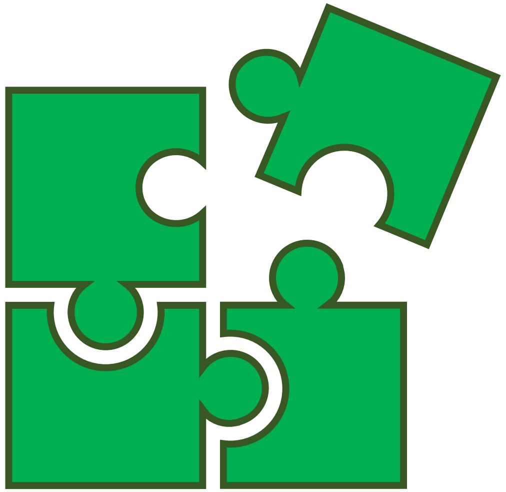
Feature

- A numerical representation of data
- Building block of a dataset
- Quality of features affect the quality of the model
- Different business problems even within same industry need not require same features.



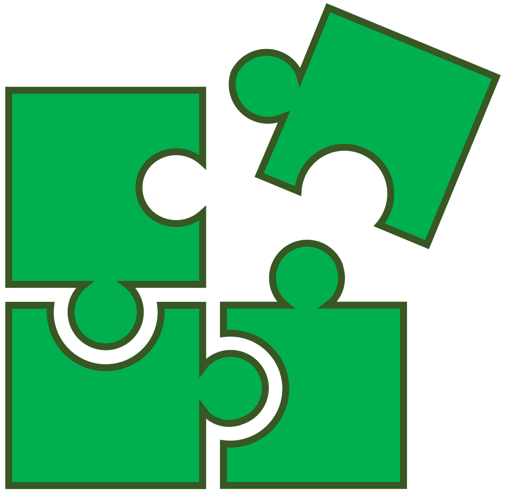
Feature


Data from various sources
Domain knowledge
Exploratory Analysis
Feature

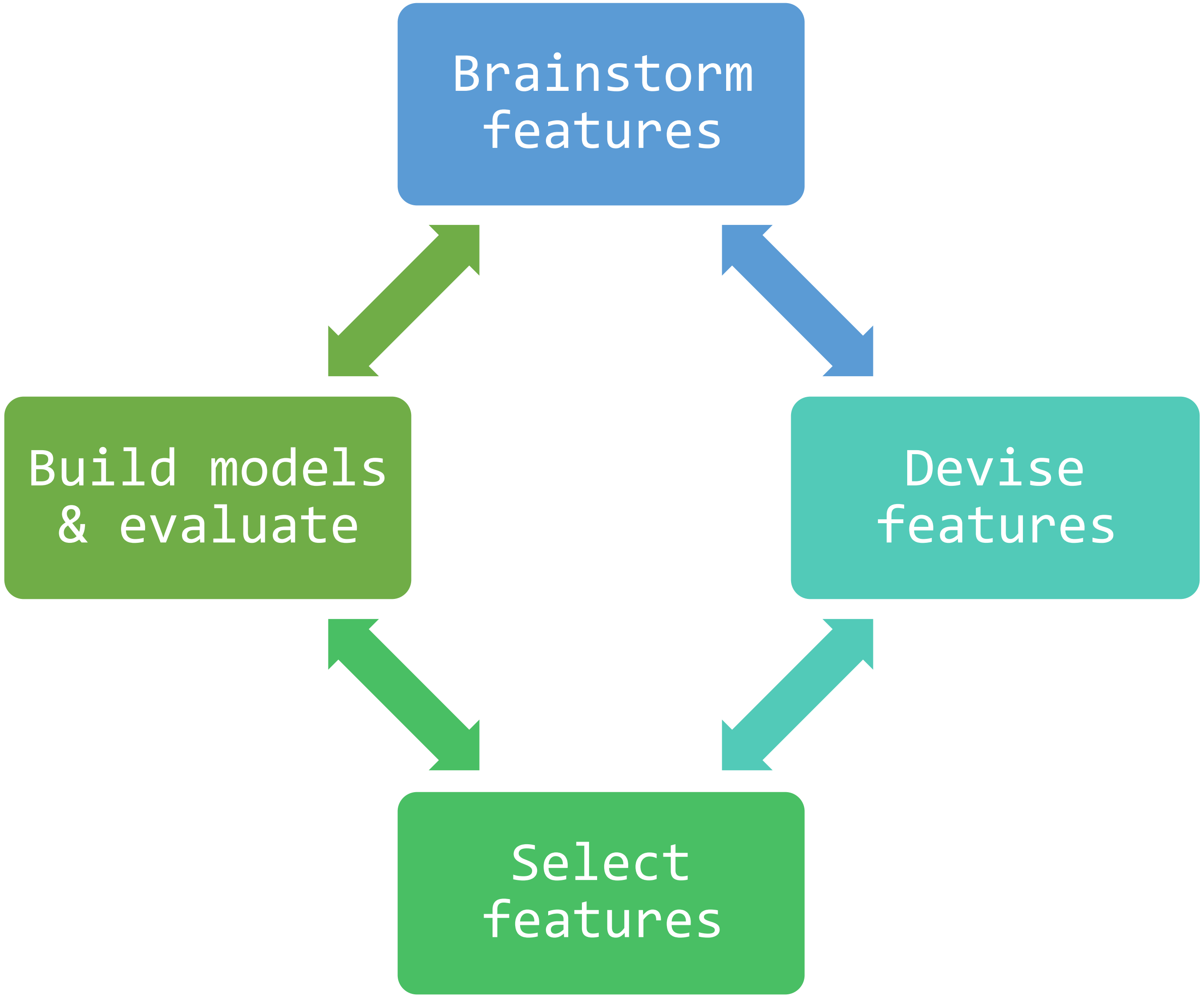


Feature Engineering

- The act of extracting features from raw data by transforming into formats that are suitable for machine learning models.
- Proper representation of sample data to learn a solution to the problem

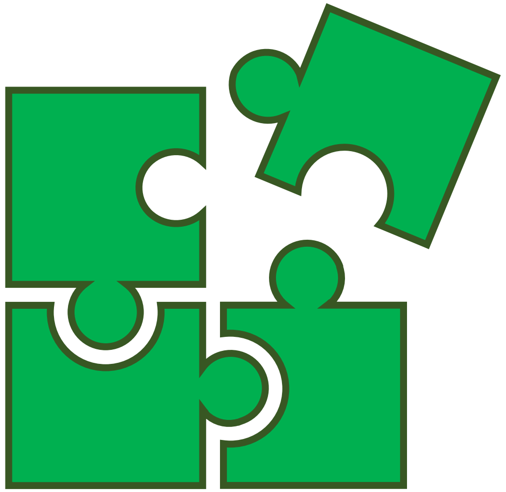


Feature Engineering



Feature Engineering Techniques

- Data Imputation
- Binning
- Feature Transforms
- Feature Encodings
- Feature Interactions
- Vectorizations



Data Imputation

- Missing values are one of the most common problems

- Why do we have missing values ?

Ans: Interruption in data flow /human errors

Data Imputation

Missing Data

1. Impute Zeroes
2. Impute Mean
3. Impute nearest value
4. Drop the record

	col1	col2	col3	col4	col5
0	2	5.0	3.0	6	NaN
1	9	NaN	9.0	0	7.0
2	19	17.0	NaN	9	NaN

Data Imputation

Missing Data

1. **Impute Zeroes**
2. Impute Mean
3. Impute nearest value
4. Drop the record

	col1	col2	col3	col4	col5
0	2	5.0	3.0	6	0.0
1	9	0.0	9.0	0	7.0
2	19	17.0	0.0	9	0.0

Missing Data

1. **Impute Zeroes**
2. Impute Mean
3. Impute nearest value
4. Drop the record

Data Imputation

Pros:

- ➢ Works well with categorical features

Cons:

- ➢ Does not factor in correlations
- ➢ It can introduce bias in data

Data Imputation

Missing Data

1. **Impute frequent**
2. Impute Mean
3. Impute nearest value
4. Drop the record

Feature-1	*Feature-2*
Male	23
Male	24
Female	25
Male	26

Data Imputation

Missing Data

1. Impute Zeroes
2. **Impute Mean**
3. Impute nearest value
4. Drop the record

	col1	col2	col3	col4	col5
0	2	5.0	3.0	6	NaN
1	9	NaN	9.0	0	7.0
2	19	17.0	NaN	9	NaN

Data Imputation

Missing Data

1. Impute Zeroes
2. **Impute Mean**
3. Impute nearest value
4. Drop the record

	col1	col2	col3	col4	col5
0	2.0	5.0	3.0	6.0	7.0
1	9.0	11.0	9.0	0.0	7.0
2	19.0	17.0	6.0	9.0	7.0

Missing Data

1. Impute Zeroes
2. **Impute Mean**
3. Impute nearest value
4. Drop the record

Data Imputation

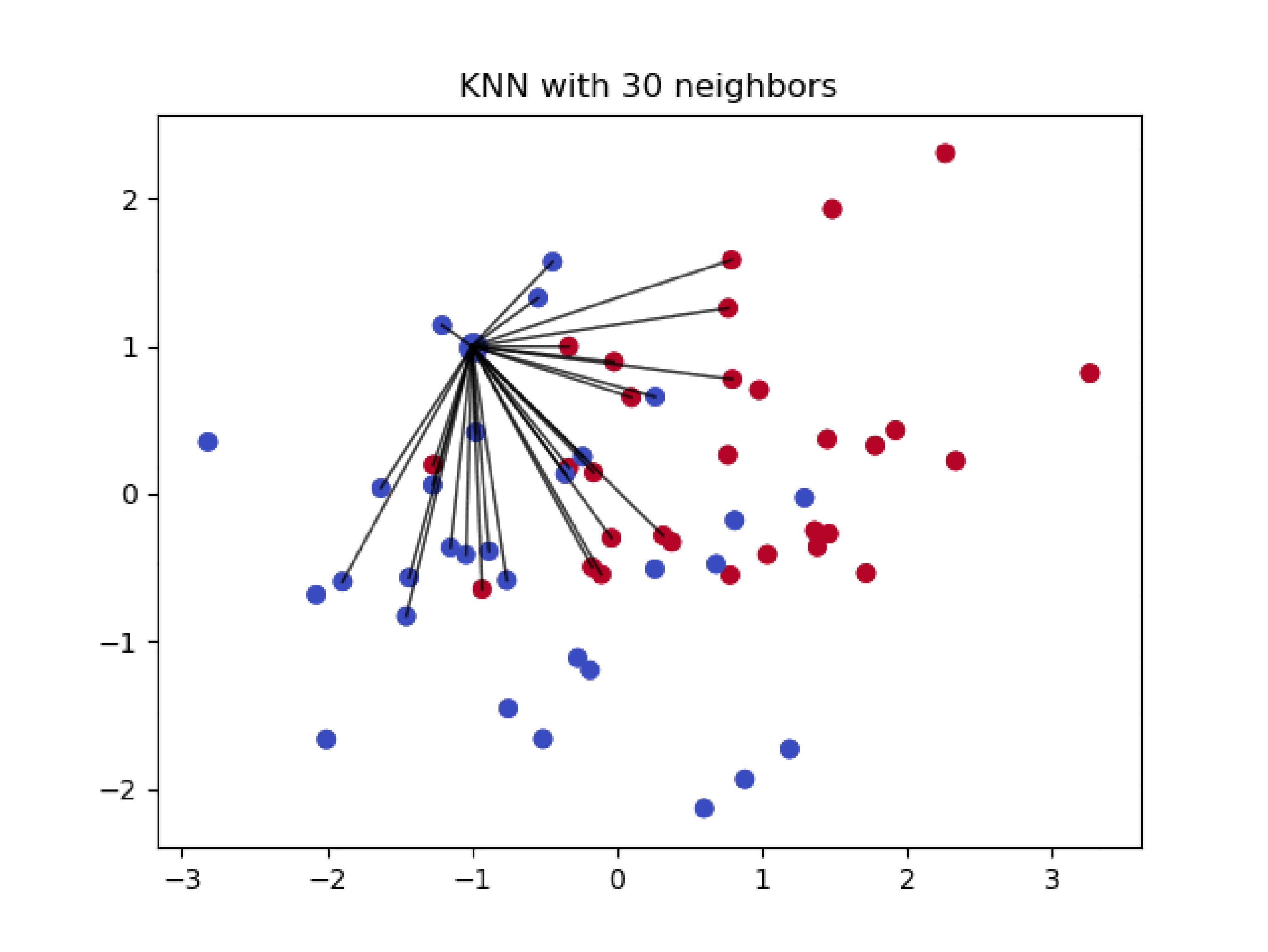
Pros:

- Easy & Fast
- Work well with small numerical data

Cons:

- Doesn't factor correlation between features
- It only works on column level

Data Imputation



Missing Data

1. Impute Zeroes
2. Impute Mean
3. **Impute nearest value**
4. Drop the record

Missing Data

1. Impute Zeroes
2. Impute Mean
3. **Impute nearest value**
4. Drop the record

Data Imputation

- The algorithm uses '**feature similarity**' to predict the values of any new data points
- Creates a a basic mean impute then uses the resulting complete list to construct a KDTree
- The resulting KDTree to compute nearest neighbours (NN)
  - After it finds the k-NNs, it takes the weighted average of them.

Data Imputation

Missing Data

1. Impute Zeroes
2. Impute Mean
3. **Impute nearest value**
4. Drop the record

Pros:

- Can be much more accurate than the mean, median or most frequent imputation methods

Cons:

- Computationally expensive
- Sensitive to outliers

Missing Data

1. Impute Zeroes
2. Impute Mean
3. Impute nearest value
4. **Drop the record**

Data Imputation

- ❑ Missing values can be handled by deleting the rows or columns having null values.
- ❑ If columns have more than half of the rows as null then the entire column can be dropped.
- ❑ The rows which are having one or more columns values as null can also be dropped.

Data Imputation

Missing Data

1. Impute Zeroes
2. Impute Mean
3. Impute nearest value
4. **Drop the record**

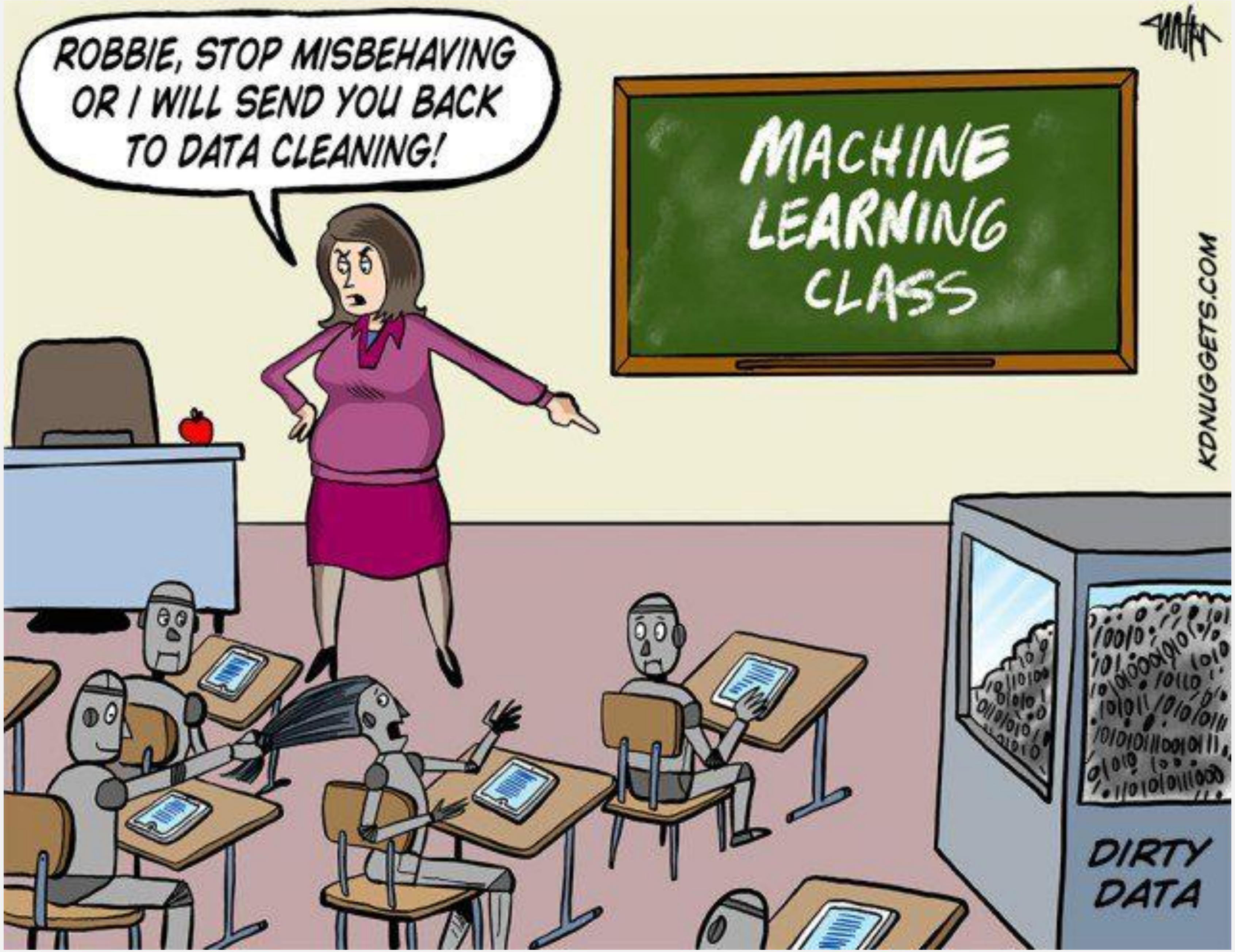
Pros:

❖ A model trained with the removal of all missing values creates a robust model.

Cons:

❖ Loss of a lot of information.

❖ Works poorly if the percentage of missing values is excessive in comparison to the complete dataset.


ROBBIE, STOP MISBEHAVING OR I WILL SEND YOU BACK TO DATA CLEANING!
MACHINE LEARNING CLASS
KDNUGGETS.COM
DIRTY DATA

Data Imputation

Outliers

1. **What is an Outlier**
2. Detection
3. Handling outliers

Outliers are those data points which differs significantly from other observations present in given dataset.

It can occur because of variability in measurement and due to misinterpretation in filling data points.

Data Imputation

Outliers

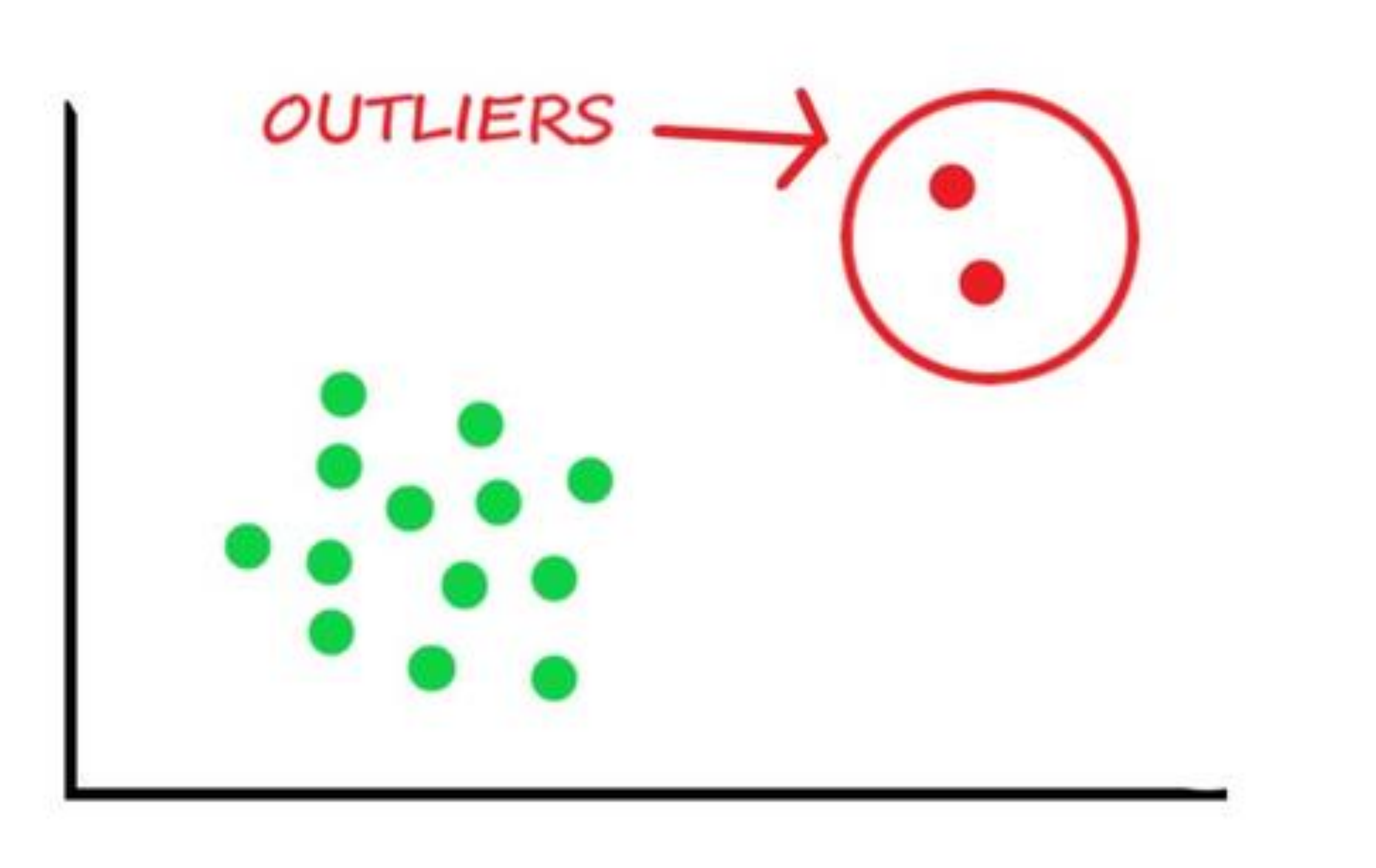
1. **What is an Outlier**
2. Detection
3. Handling outliers

1. *Data Entry Errors*
2. *Measurement Error*
3. *Experimental errors*
4. *Intentional*
5. *Data processing errors*
6. *Sampling errors*
7. *Natural Outlier*

Data Imputation

Outliers

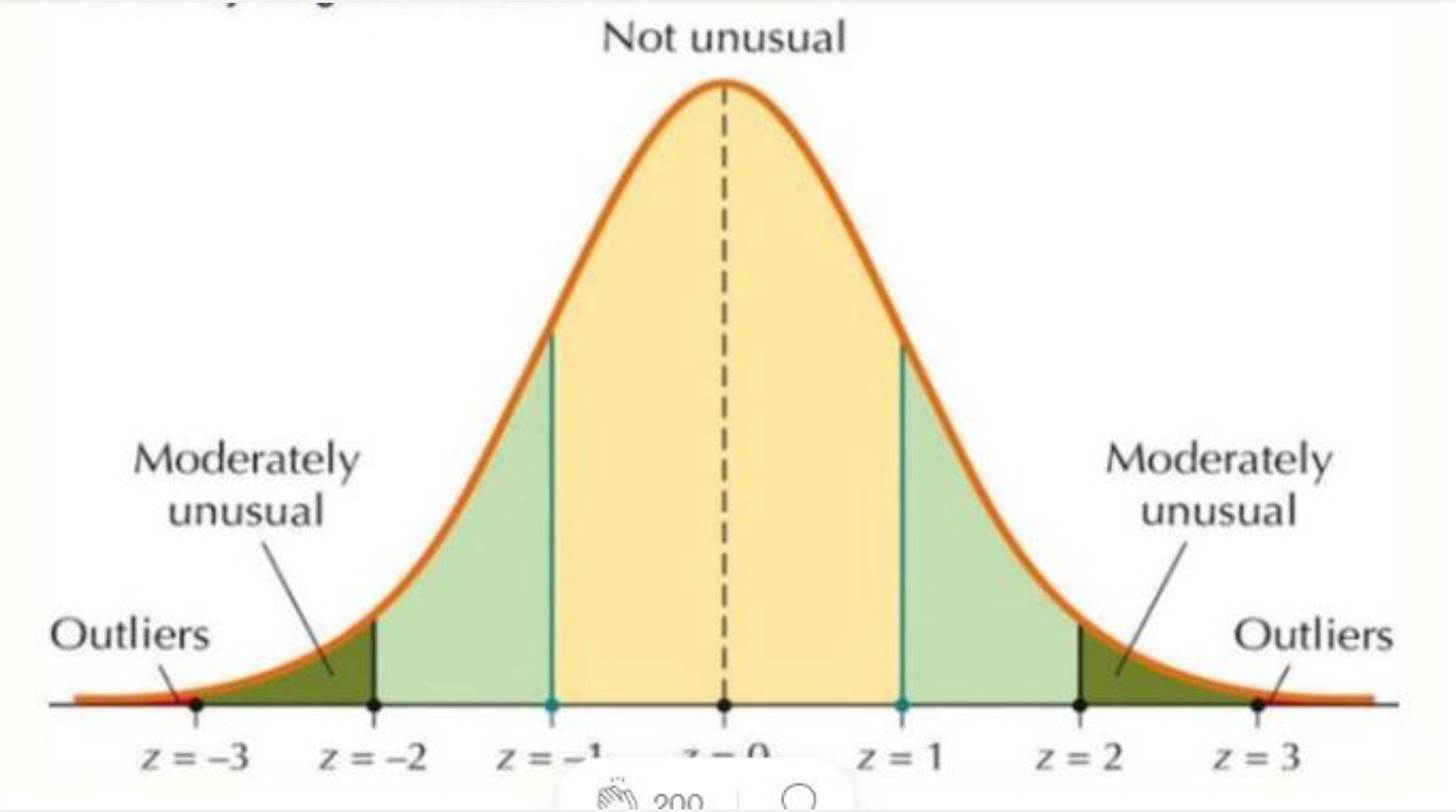
1. **What is an Outlier**
2. Detection
3. Handling outliers



Data Imputation

Outliers

Z-score detection



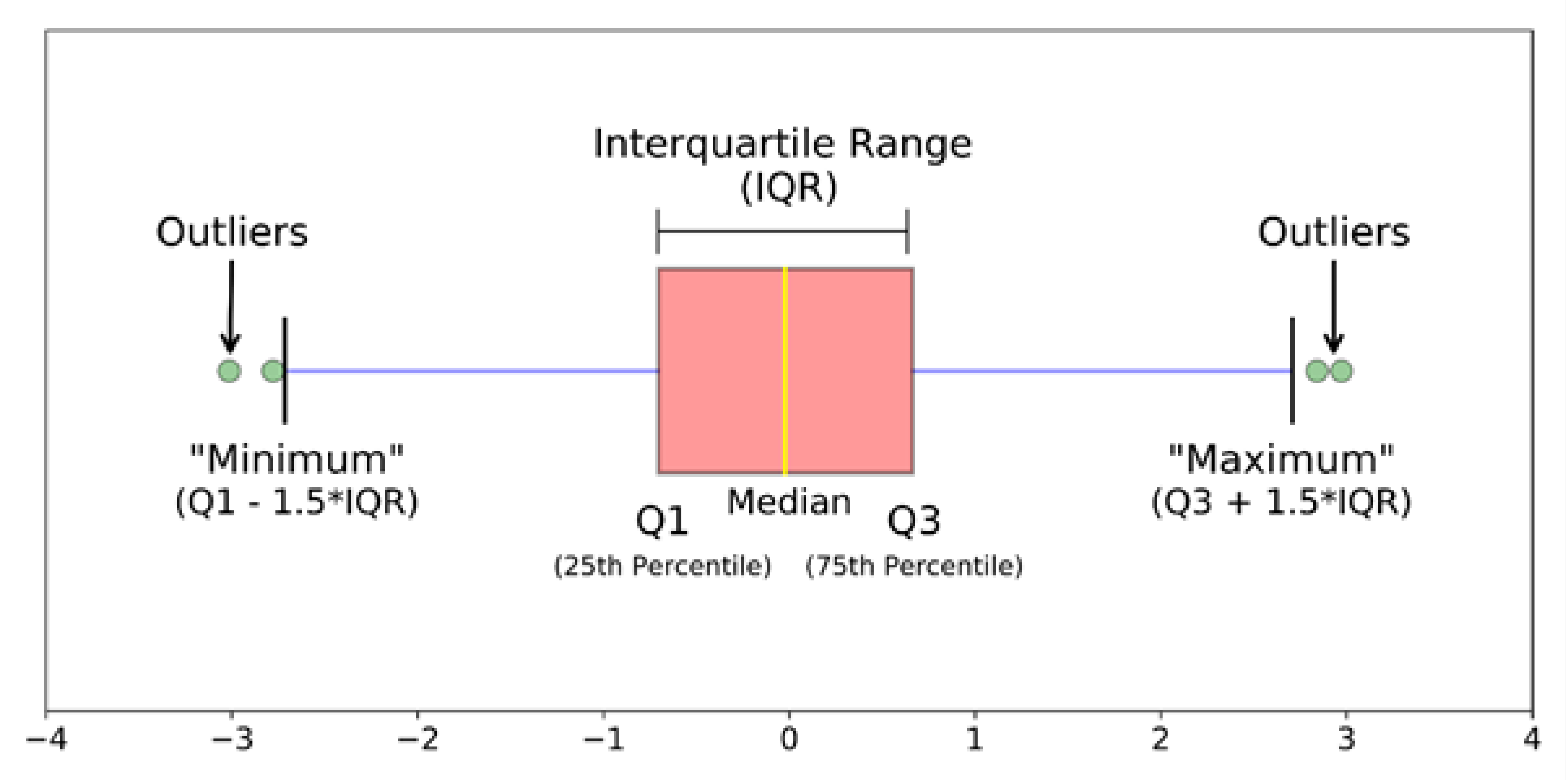
1. What is an Outlier
2. **Detection**
3. Handling outliers

Data Imputation

Outliers

1. What is an Outlier
2. **Detection**
3. Handling outliers

Inter-Quantile Range

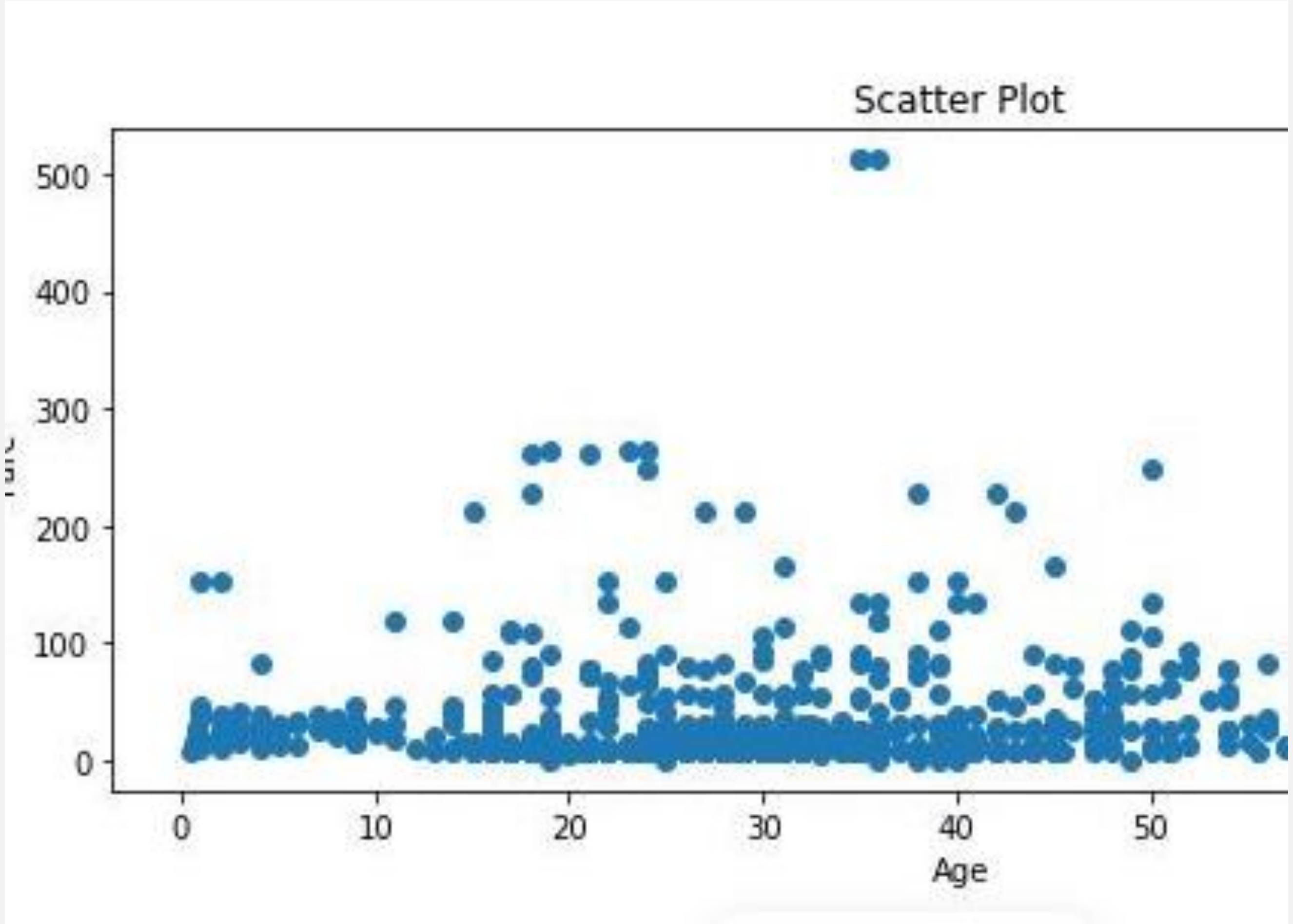


Outliers

1. What is an Outlier
2. Detection
3. Handling outliers

Data Imputation

Visualization

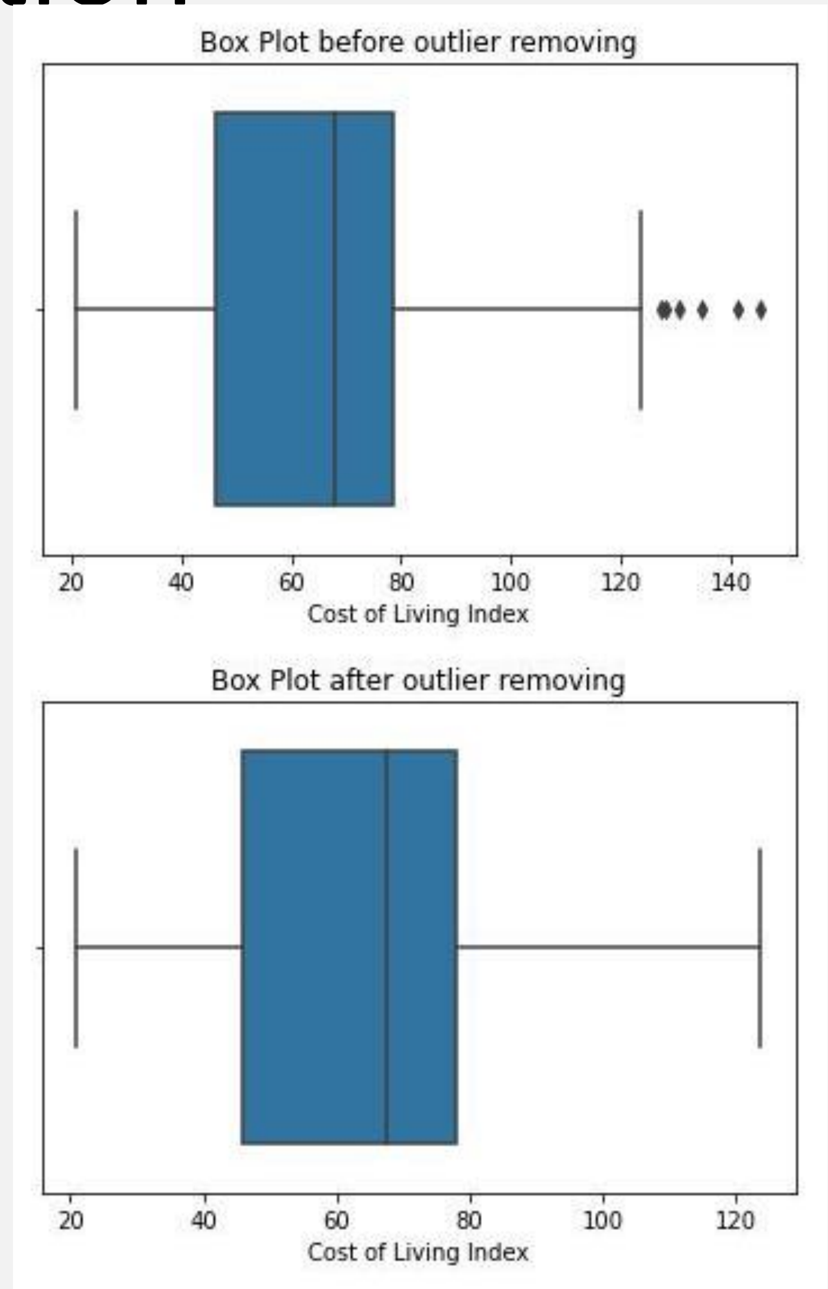


Outliers

1. What is an Outlier
2. Detection
3. **Handling outliers**

Data Imputation

1. Removing the records
2. Imputing with mean/median




How to make everything work
Keep removing strange data until it fits
HIGH IMPACT PhD MEMES

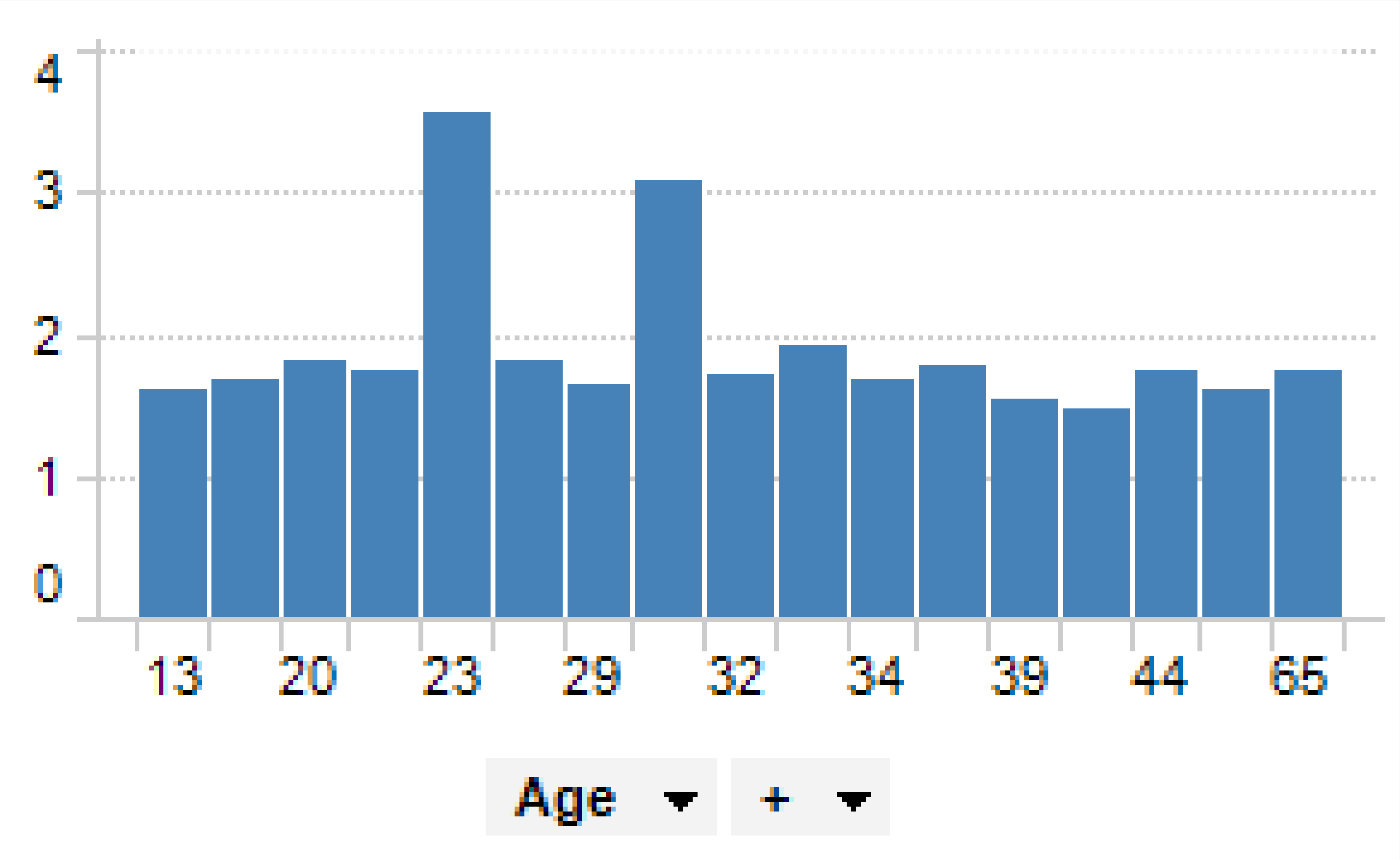
Data Binning

Binning is the process of assigning category to the numerical values

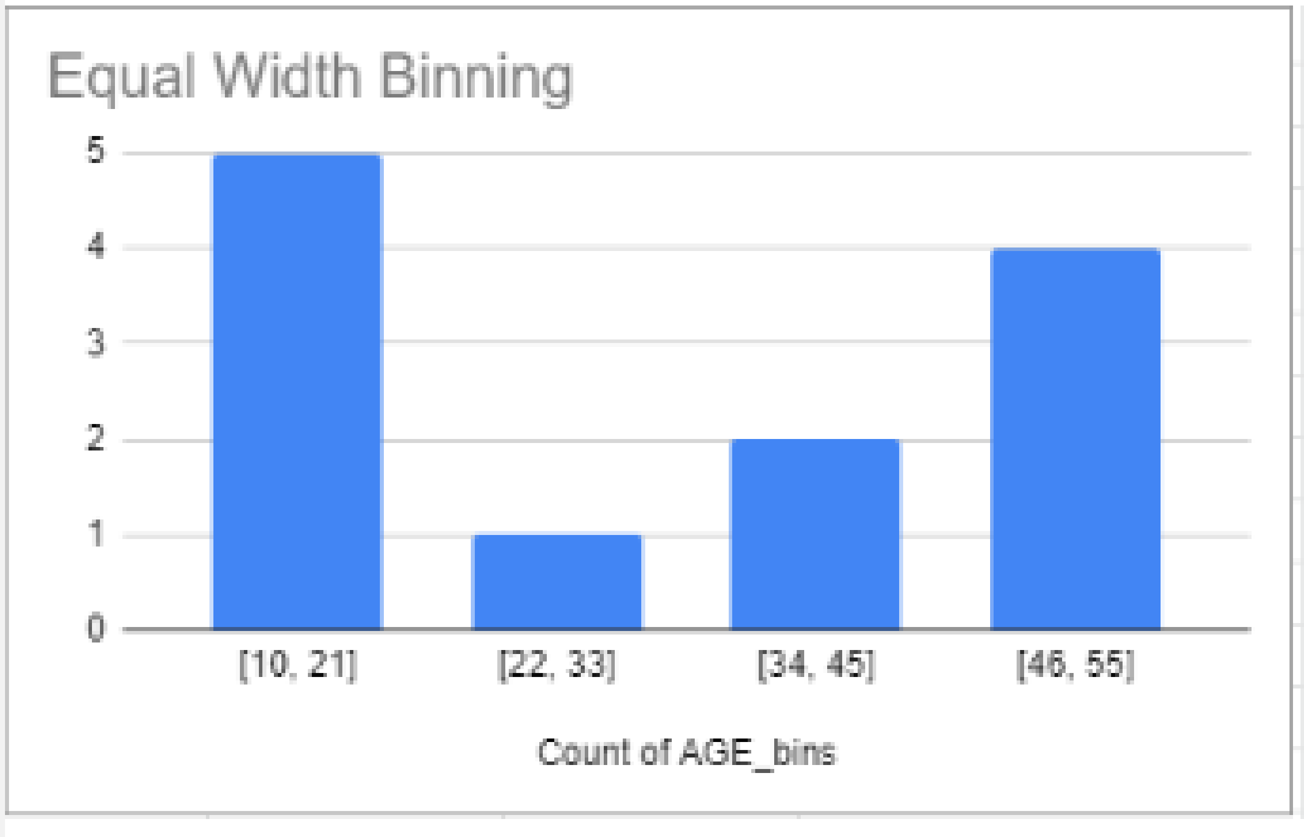
Benefits:

1. Makes the model more robust
2. Prevents overfitting

Data Binning



Data Binning



Feature Encoding

Feature encoding

Why ?

A machine can only understand the numbers. Any textual data needs to be converted to an appropriate numerical representation

Feature Encoding

Onehot encoding

Creates a new features based on the number of unique values in the categorical feature.

Every unique value in the category will be added as a feature

Feature Encoding

Onehot encoding

Index	Animal
0	Dog
1	Cat
2	Sheep
3	Horse
4	Lion

One-Hot code

Index	Dog	Cat	Sheep	Lion	Horse
0	1	0	0	0	0
1	0	1	0	0	0
2	0	0	1	0	0
3	0	0	0	0	1
4	0	0	0	1	0

Feature Encoding

Label encoding

Each label is assigned a unique integer. It is applied when the data is ordinal.

Retaining the order is important. Hence, encoding should reflect the sequence.

Feature Encoding

Label encoding

	Degree
0	High school
1	Masters
2	Diploma
3	Bachelors
4	Bachelors
5	Masters
6	Phd
7	High school
8	High school

	Degree
0	1
1	4
2	2
3	3
4	3
5	4
6	5
7	1
8	1

Feature Encoding

Target encoding

A Bayesian encoding technique

Bayesian encoders use information from dependent/target variables to encode categorical data

Feature Encoding

Target encoding

	class	Marks
0	A,	50
1	B	30
2	C	70
3	B	80
4	C	45
5	A	97
6	A	80
7	A	68

	class
0	65.000000
1	57.689414
2	59.517061
3	57.689414
4	59.517061
5	79.679951
6	79.679951
7	79.679951

Feature Encoding

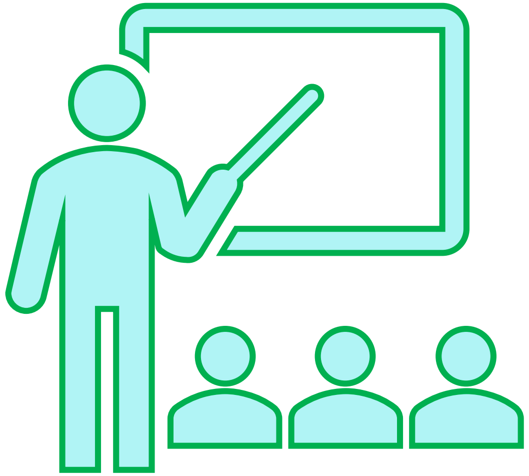
Other Encoders

1. Dummy encoding
2. Binary Encoding
3. Base N Encoding
4. Hash Encoding

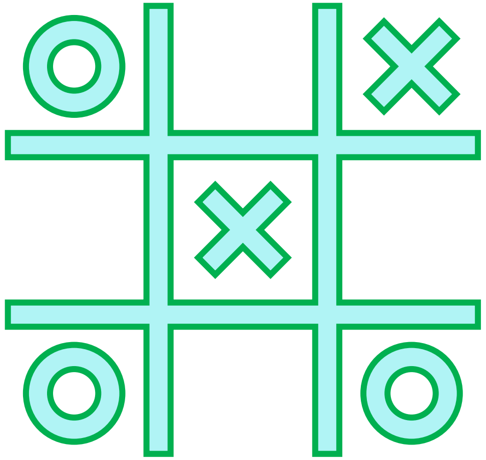
Section - 5

Supervised Learning

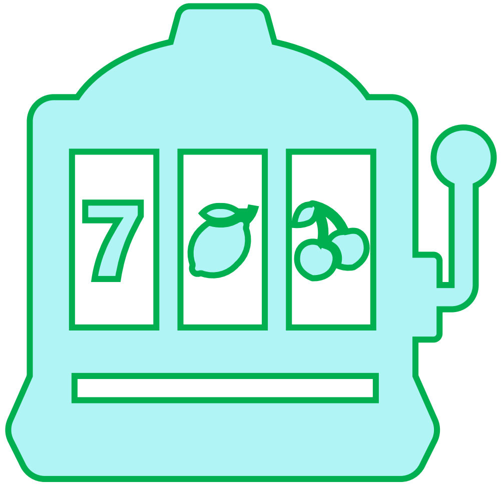
Types of Machine Learning



Supervised Learning



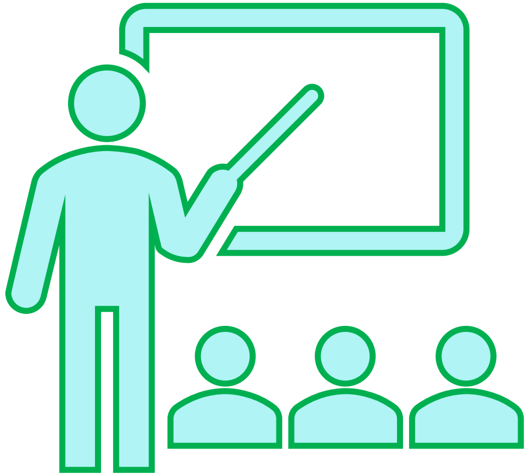
Unsupervised Learning



Reinforcement Learning

Supervised Learning

Supervised learning uses a training set to teach models to yield the desired output. This training dataset includes inputs and correct outputs, which allow the model to learn over time.



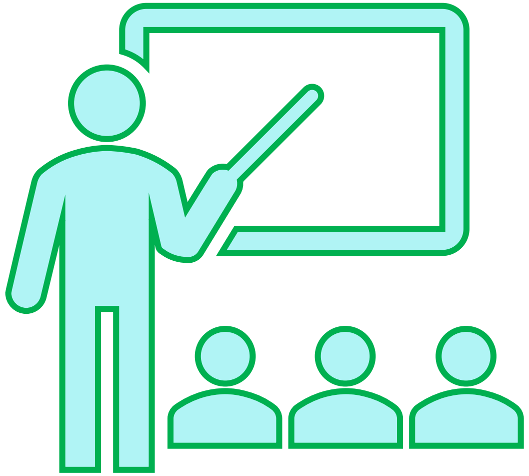
Supervised Learning

Input variable **x** and output variable **y**

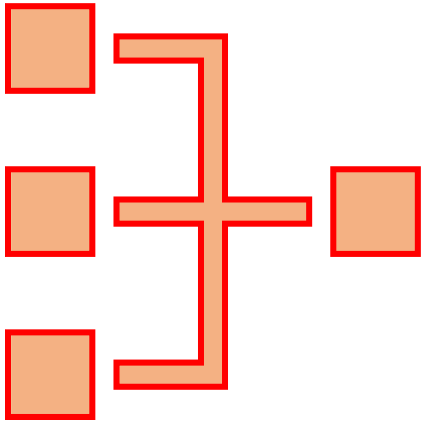
Algorithm learns the mapping function **$y = f(x)$**

Algorithm approximates the relationship between x and y such that it can predict y for new values of x

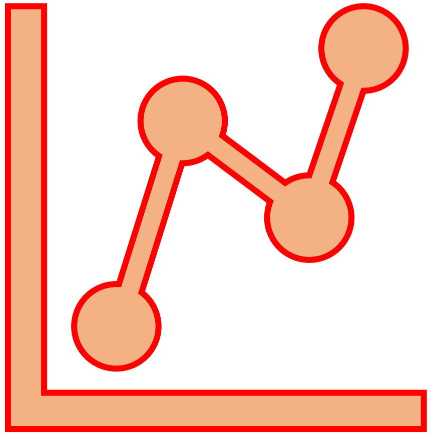
Types



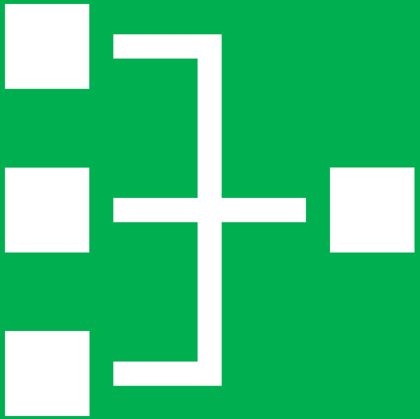
Supervised Learning



Classification



Regression



CLASSIFICATION PREDICTS THE CATEGORY THE DATA BELONGS TO

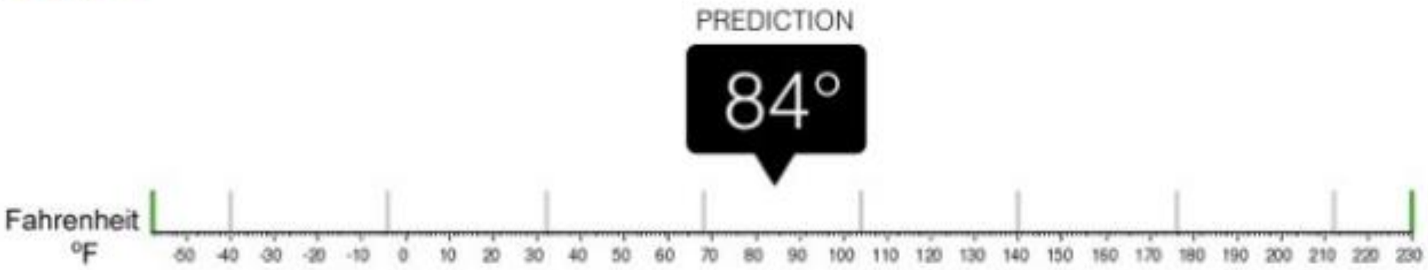
REGRESSION PREDICTS A NUMERICAL VALUE BASED ON PREVIOUSLY OBSERVED DATA.

Section - 5

Supervised Learning

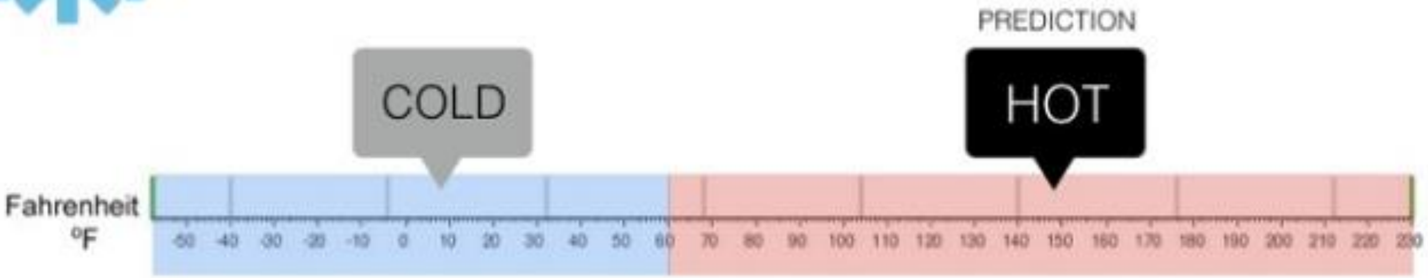
Regression

What is the temperature going to be tomorrow?



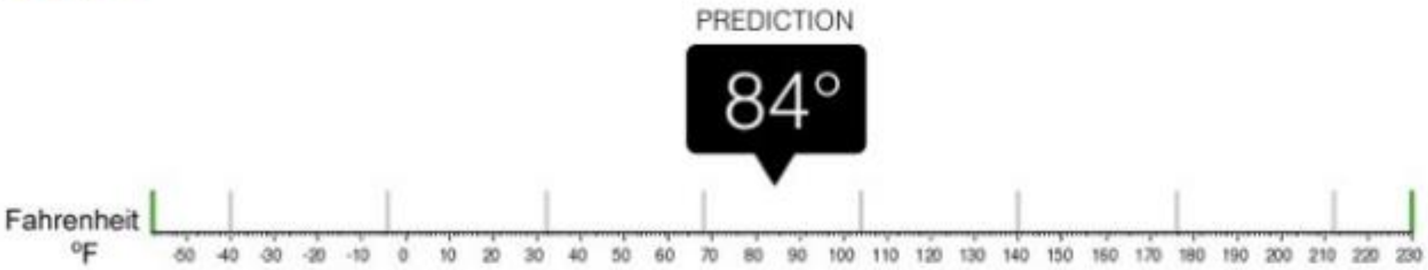
Classification

Will it be Cold or Hot tomorrow?



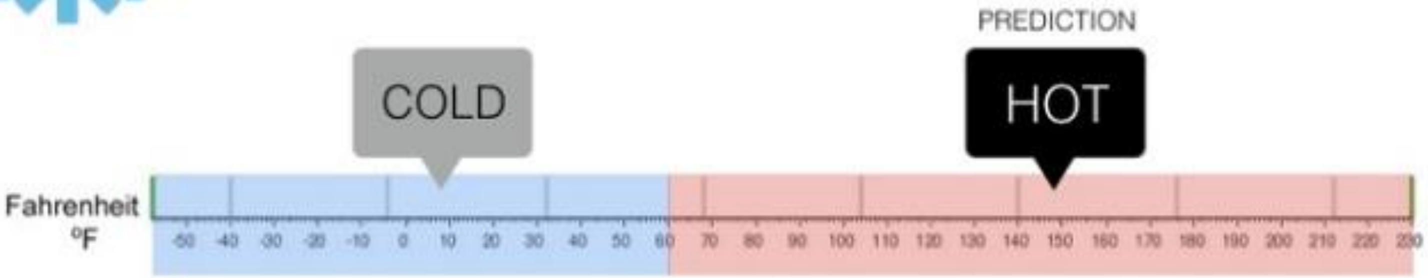
Regression

What is the temperature going to be tomorrow?



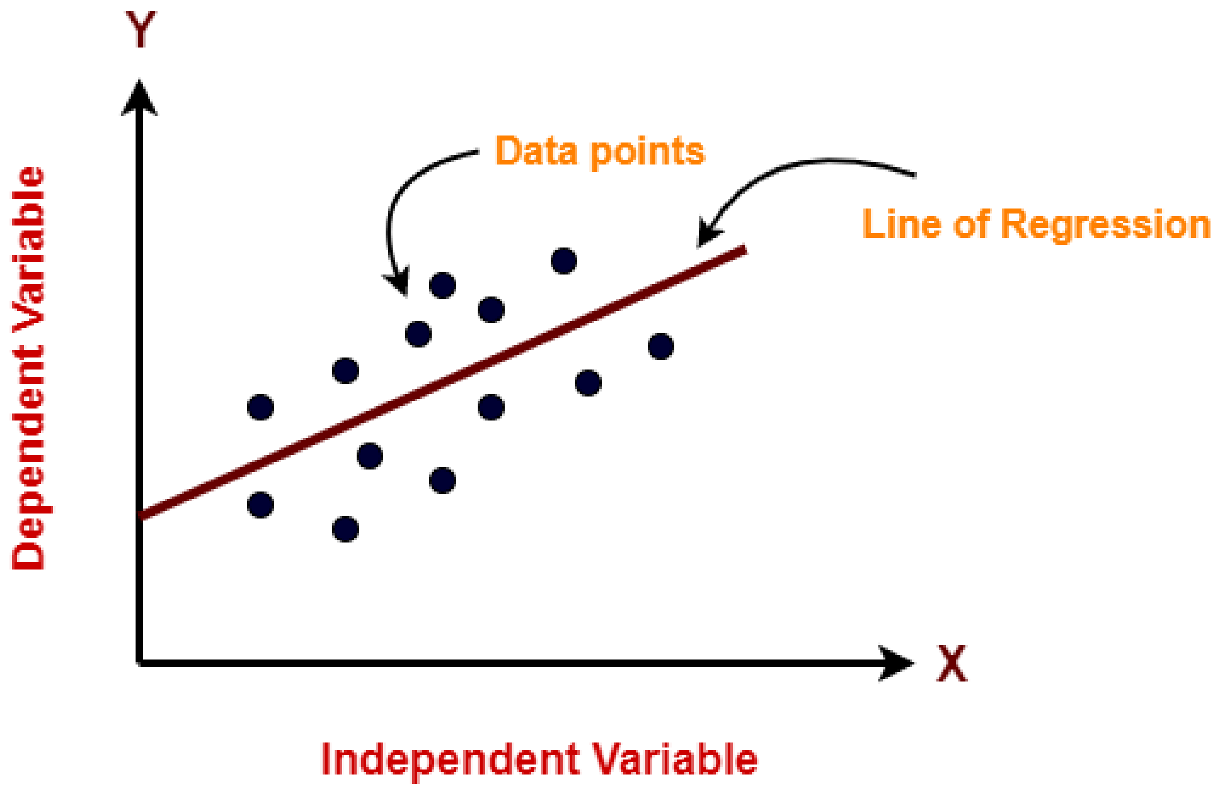
Classification

Will it be Cold or Hot tomorrow?



Linear Regression

- Linear regression performs the task to predict a dependent variable(target) based on the given independent variable(s).
- So, this regression technique finds out a linear relationship between a dependent variable and the other given independent variables.



Linear Regression

Pros:

•Linear Regression is simple to implement.

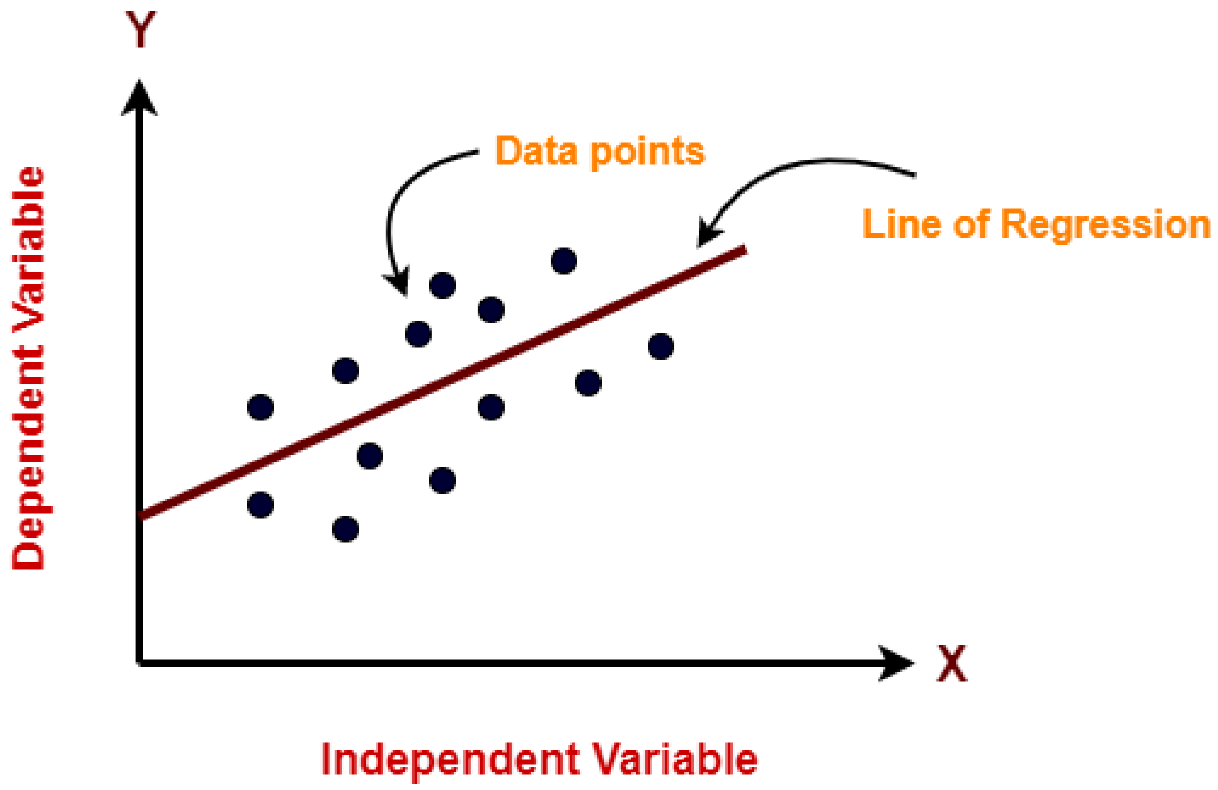
•Less complexity compared to other algorithms.

Cons:

•Outliers affect this algorithm badly.

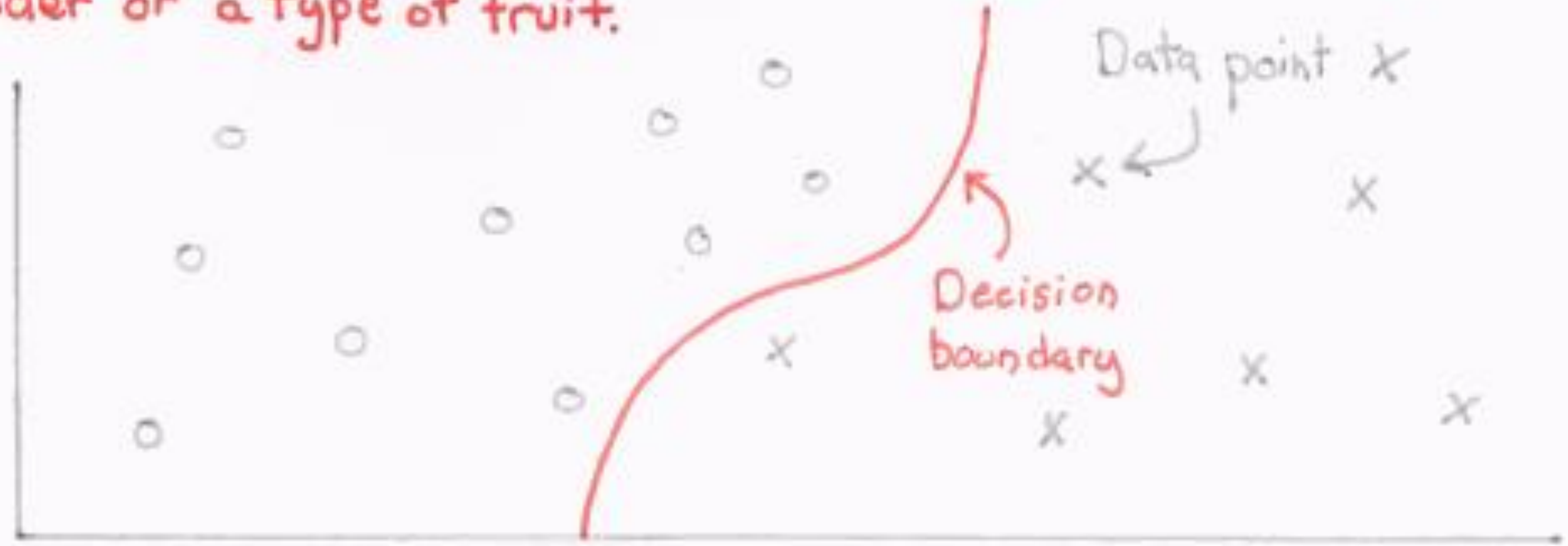
•Linear Regression may lead to over-fitting

•It over-simplifies real-world problems by assuming a linear relationship among the variables, hence not recommended for practical use-cases.



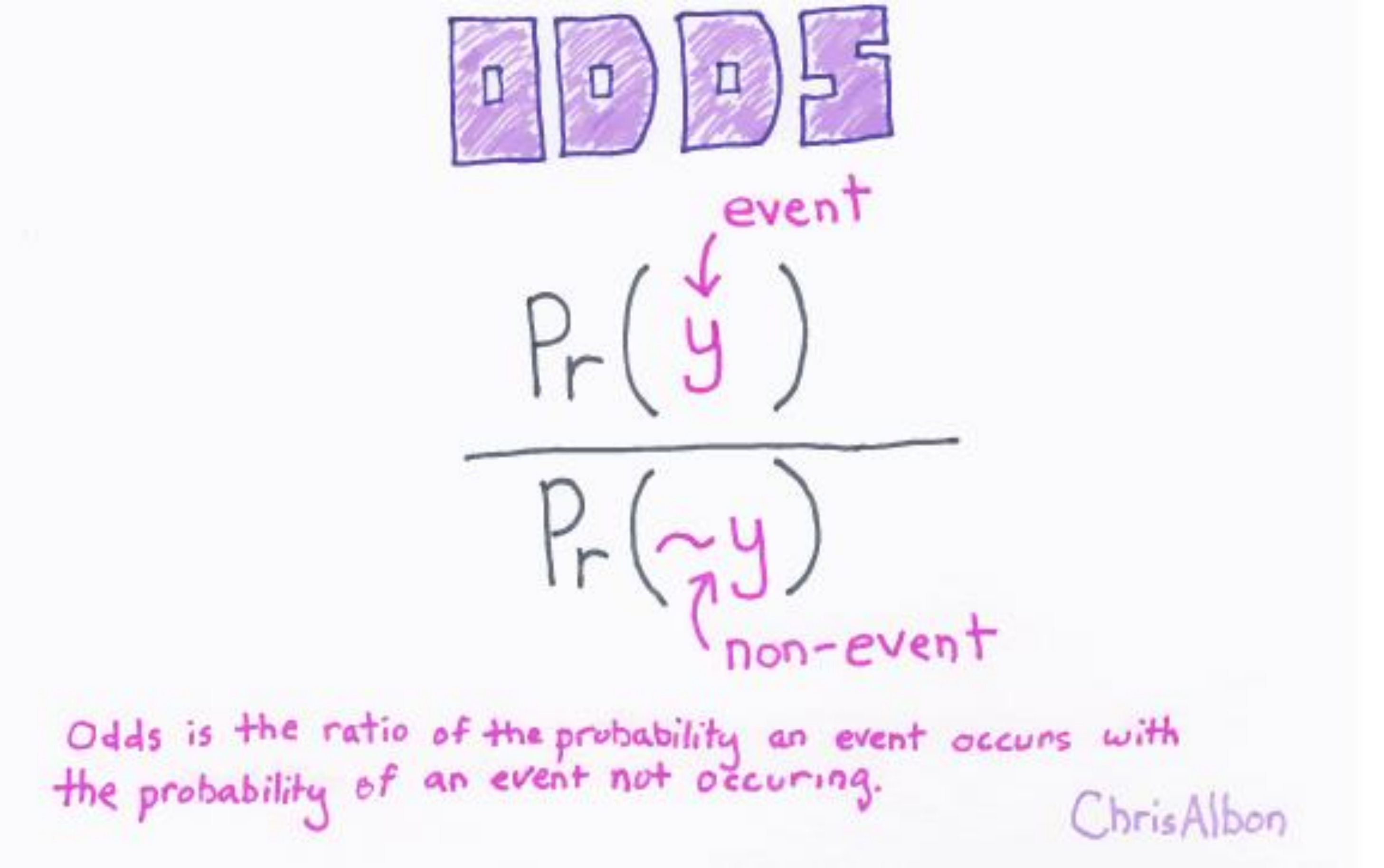
CLASSIFICATION

Classification problems are when we are training a model to predict qualitative targets. For example: gender or a type of fruit.



ChrisAlbon

Logistic Regression



Logistic Regression Assumptions

There is little to no multicollinearity between the independent variables.

The independent variables are linearly related to the log odds (log (p/(1-p)).

The data sample sizes are larger, which is integral for better results.

There are no outliers.

Logistic Regression Advantages

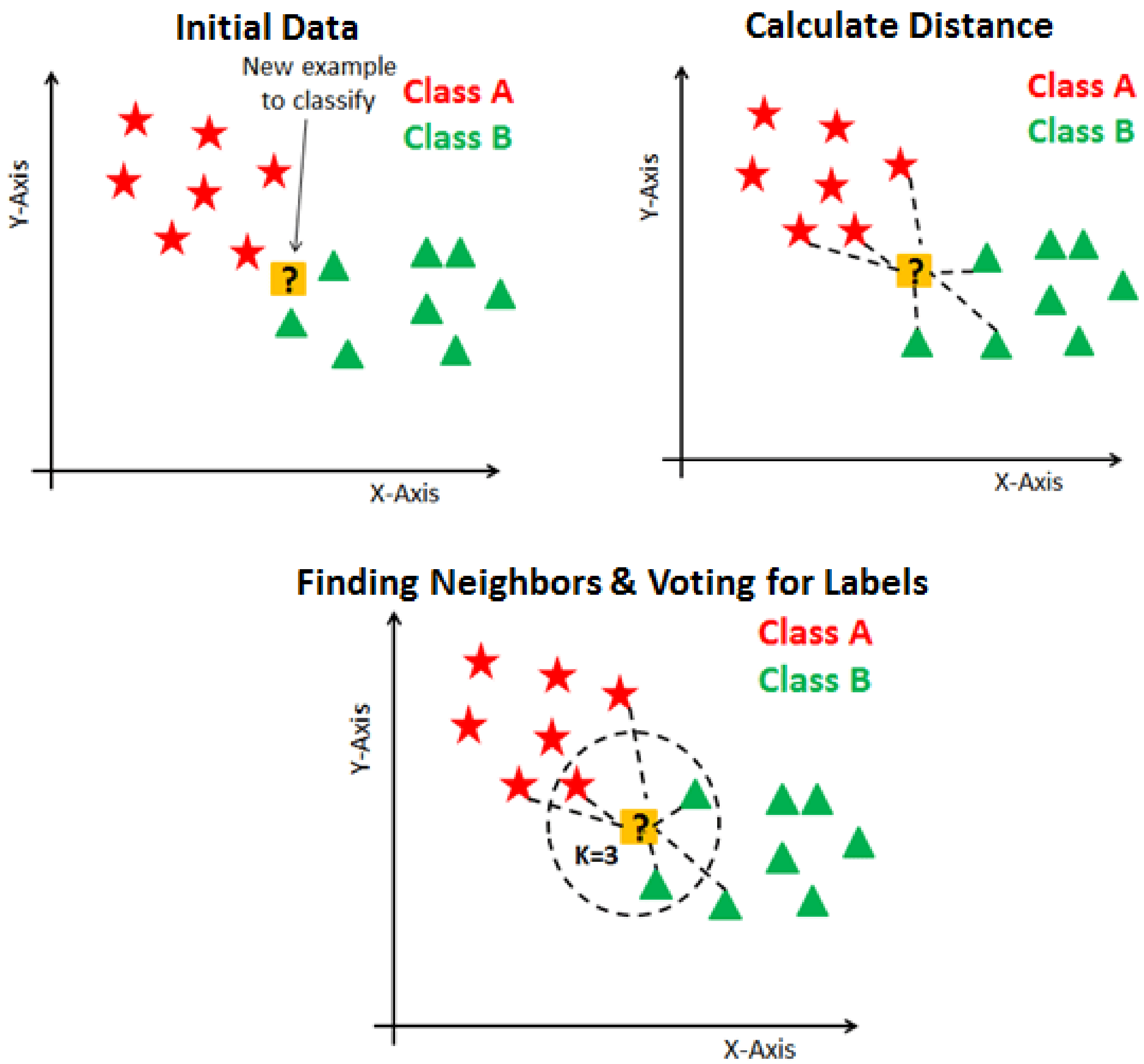
- Simple to understand, easy to implement, and efficient to train
- Performs well when the dataset is linearly separable
- Good accuracy for smaller datasets
- Doesn't make any assumptions about the distribution of classes
- Provides well-calibrated probabilities

Logistic Regression Disadvantages

- Constructs linear boundaries
- Can lead to overfitting if the number of features is more than the number of observations
- Predictors should have average or no multicollinearity
- Challenging to obtain complex relationships.
- Can't solve non-linear problems
- Sensitive to outliers

K-Nearest Neighbors

- The KNN algorithm assumes that similar things exist in close proximity.
- In other words, simila things are near to each other.



K-Nearest Neighbors algo

1. Load the data
2. Choose K value
3. For each data point in the data:
    1. Find the Euclidean distance to all training data samples
    2. Store the distances on an ordered list and sort it
    3. Choose the top K entries from the sorted list
    4. Label the test point based on the majority of classes present in the selected points
4. End

K-Nearest Neighbors Advantages

- It's easy to understand and simple to implement
- It can be used for both classification and regression problems
- It's ideal for non-linear data since there's no assumption about underlying data
- It can naturally handle multi-class cases
- It can perform well with enough representative data

K-Nearest Neighbors Disadvantages

- Associated computation cost is high as it stores all the training data
- Requires high memory storage
- Need to determine the value of K
- Prediction is slow if the value of N is high
- Sensitive to irrelevant features

Naïve Bayes Algorithm

The naive Bayes classifier is based on Bayes' theorem with the independence assumptions between predictors

it assumes the presence of a feature in a class is unrelated to any other feature

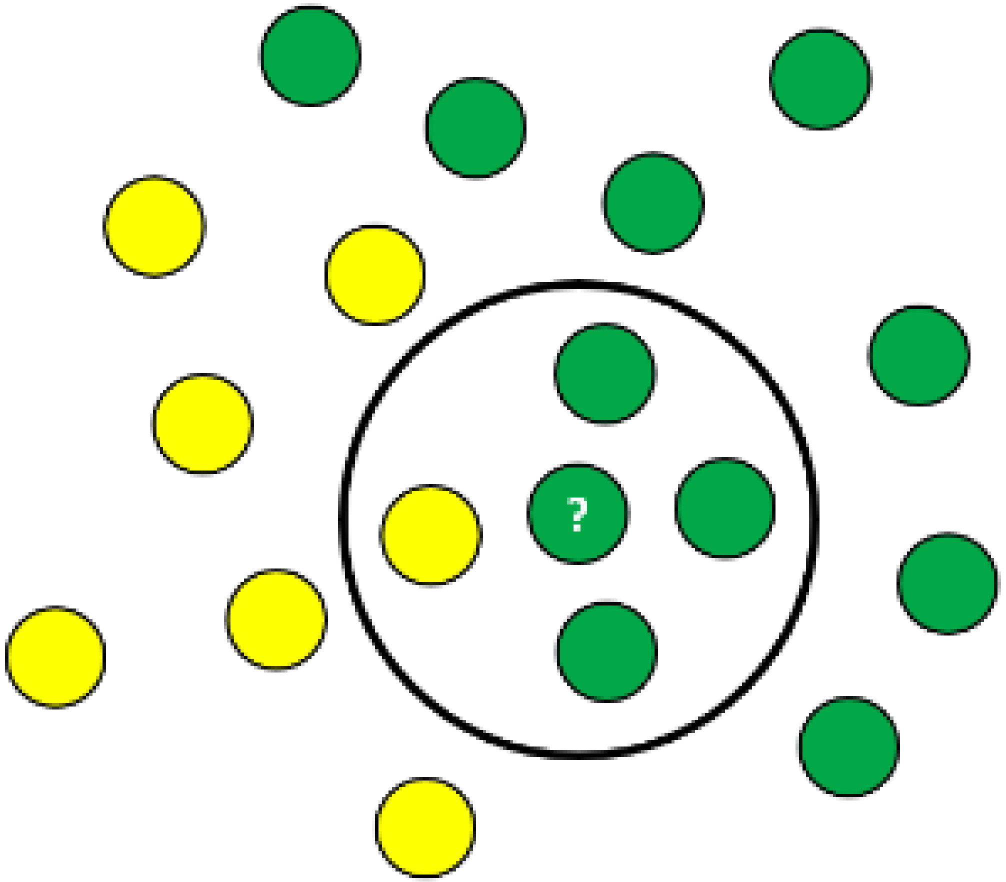
Even if these features depend on each other, or upon the existence of the other features, all of these properties independently

Naïve Bayes

$$\underbrace{P(c \mid x)}_{\text{Posterior Probability}} = \frac{\overbrace{P(x \mid c)}^{\text{Likelihood}}\,\overbrace{P(c)}^{\text{Class Prior Probability}}}{\underbrace{P(x)}_{\text{Predictor Prior Probability}}}$$

$$P(c \mid \mathrm{X}) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Naïve Bayes



NB Classification Example

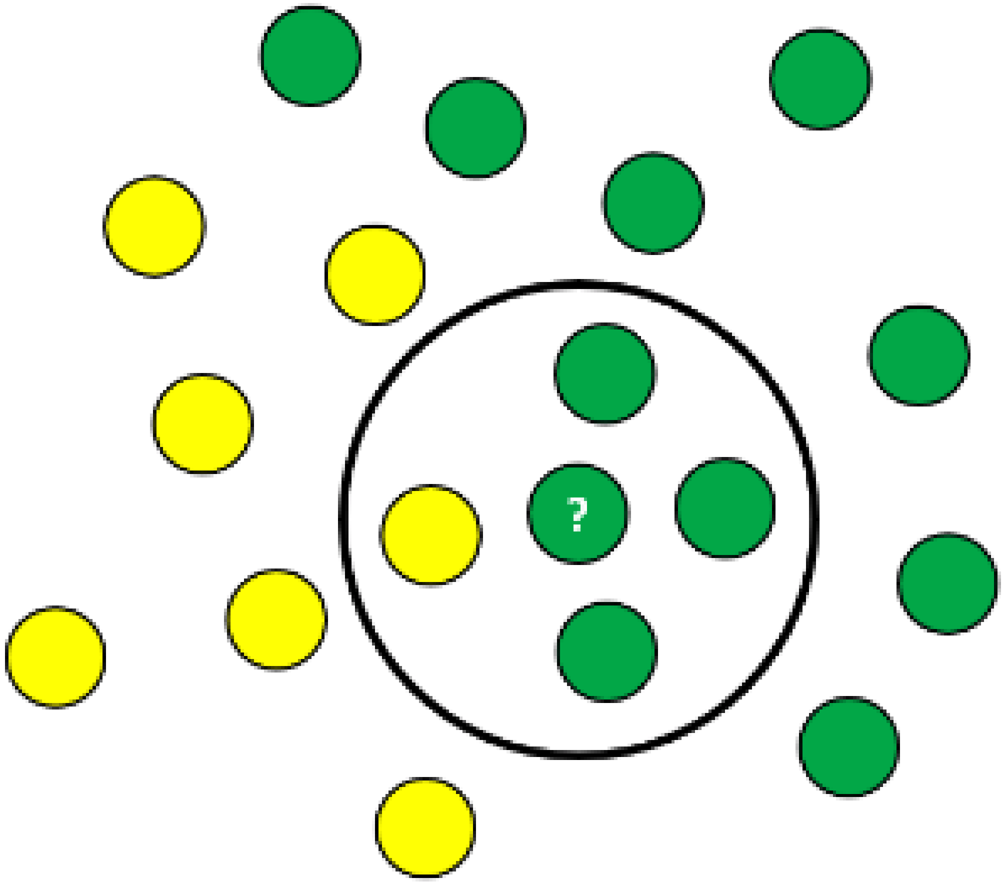
Naïve Bayes

<u>1. Calculate Prior Probability</u>

P(class) = Number of data points in the class/Total no. of observations

P(yellow) = 10/17

P(green) = 7/17



NB Classification Example

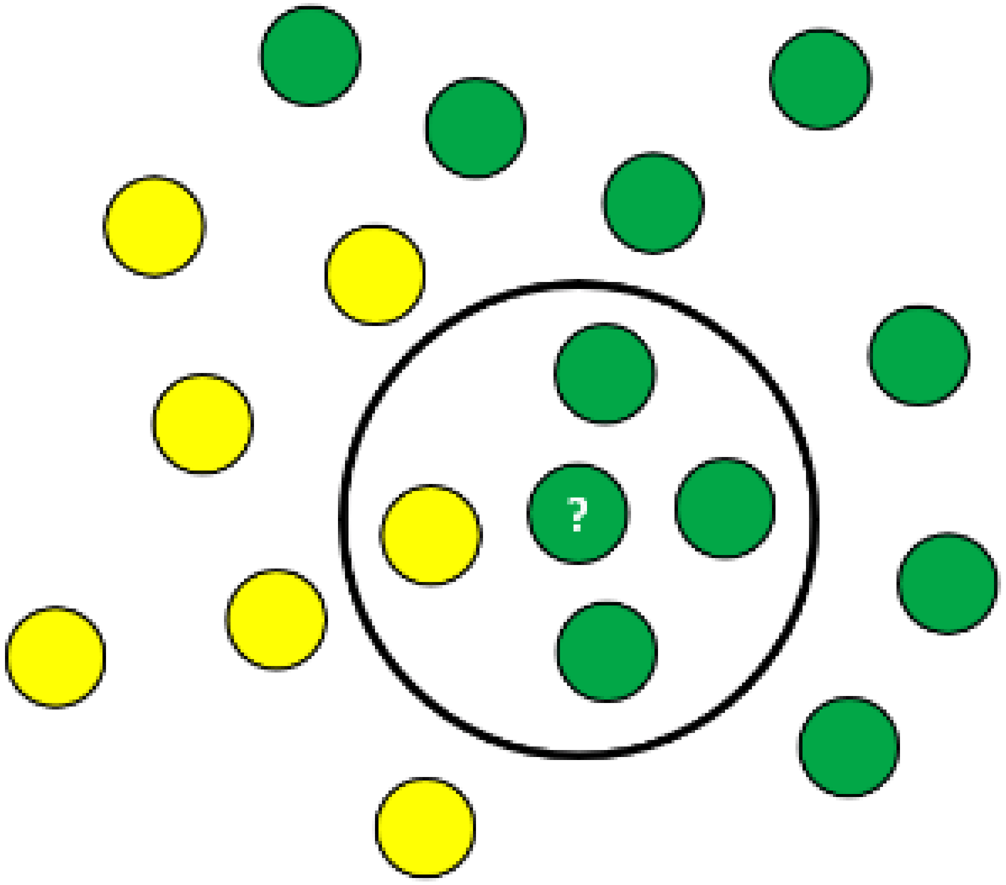
Naïve Bayes

<u>2. Calculate Marginal Likelihood</u>

P(data) = Number of data points similar to observation/Total no. of observations

P(?) = 4/17

The value is present in checking both the probabilities.



NB Classification Example

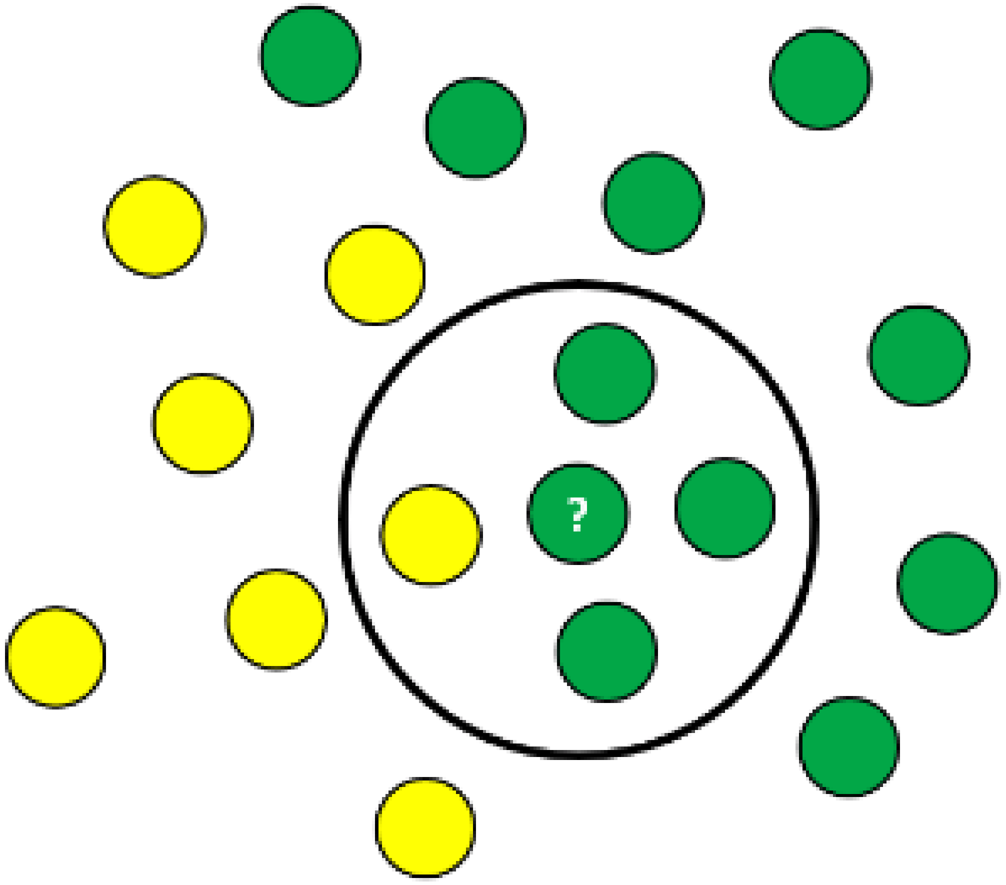
Naïve Bayes

<u>3. Calculate Likelihood</u>

P(data/class) = Number of similar observations to the class/Total no. of points in the class.

P(?/yellow) = 1/7

P(?/green) = 3/10



NB Classification Example

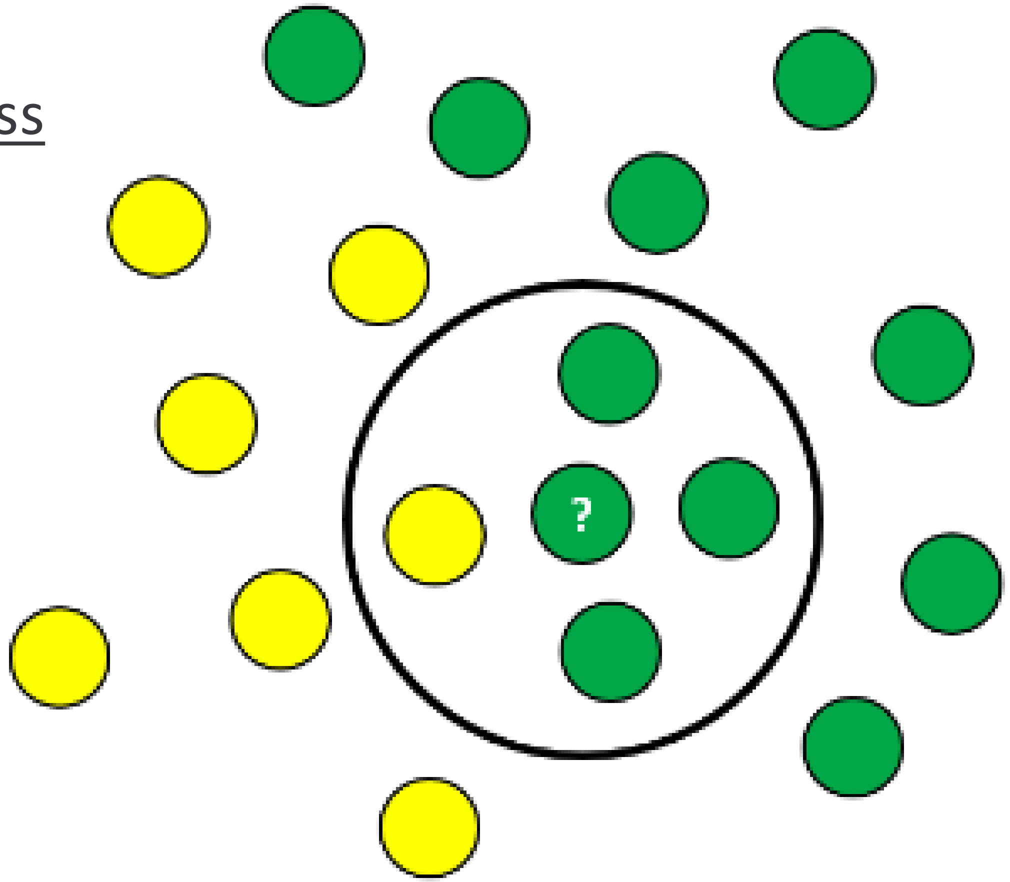
Naïve Bayes

4. Posterior Probability for Each Class

$$p(class/data) = \frac{P(data/class) * P(class)}{P(data)}$$

$$P(yellow/?) = \frac{1/7 * 7/17}{4/17} = 0.25$$

$$P(green/?) = \frac{3/10 * 10/17}{4/17} = 0.75$$



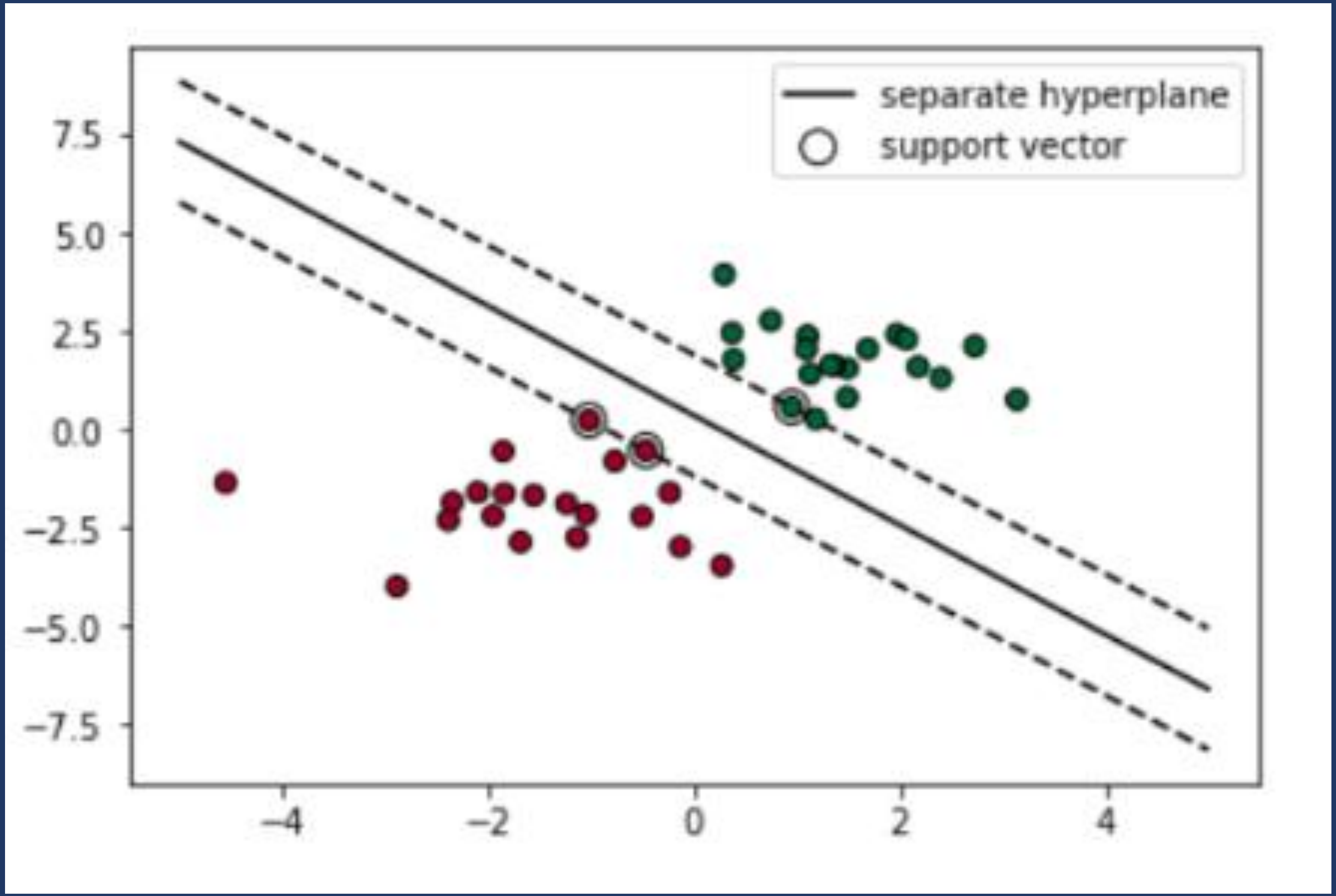
NB Classification Example

Naïve Bayes Advantages

- It is simple and easy to implement
- It doesn’t require as much training data
- It handles both continuous and discrete data
- It is highly scalable with the number of predictors and data points
- It is fast and can be used to make real-time predictions
- It is not sensitive to irrelevant features

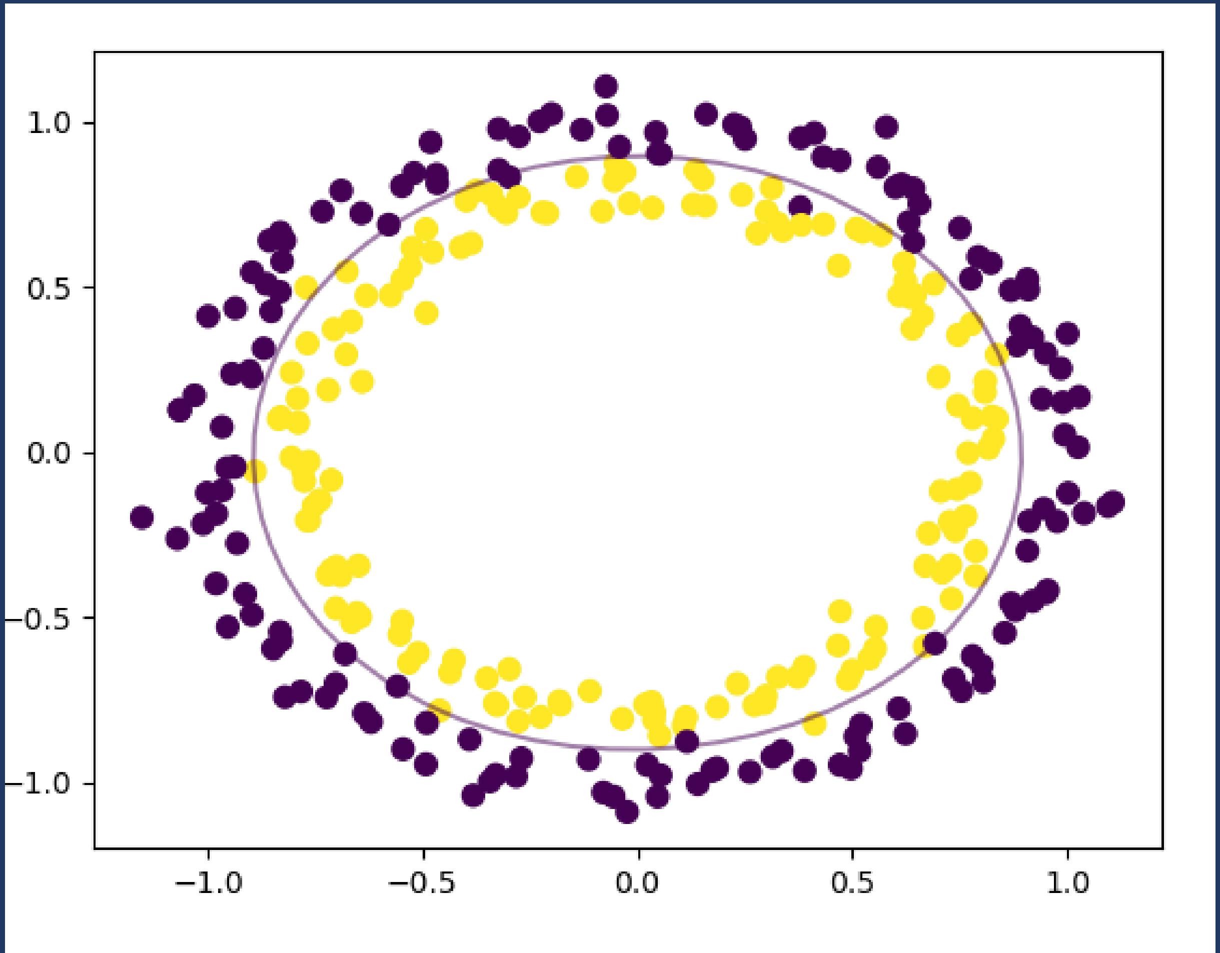
Support Vector Machines

- Support Vector Machine (the “road machine”) is responsible for finding the decision boundary to separate different classes and maximize the margin.

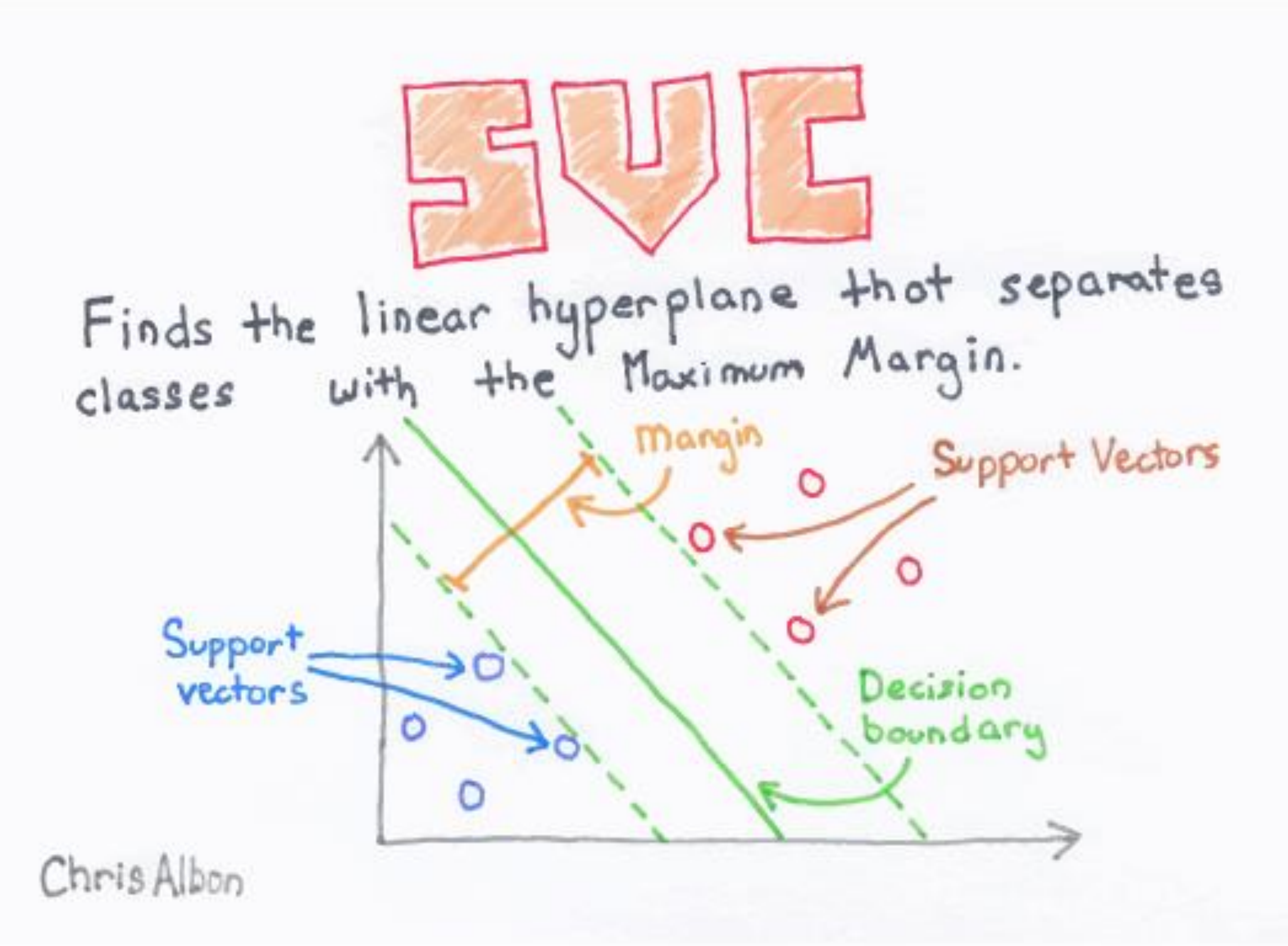


Support Vector Machines

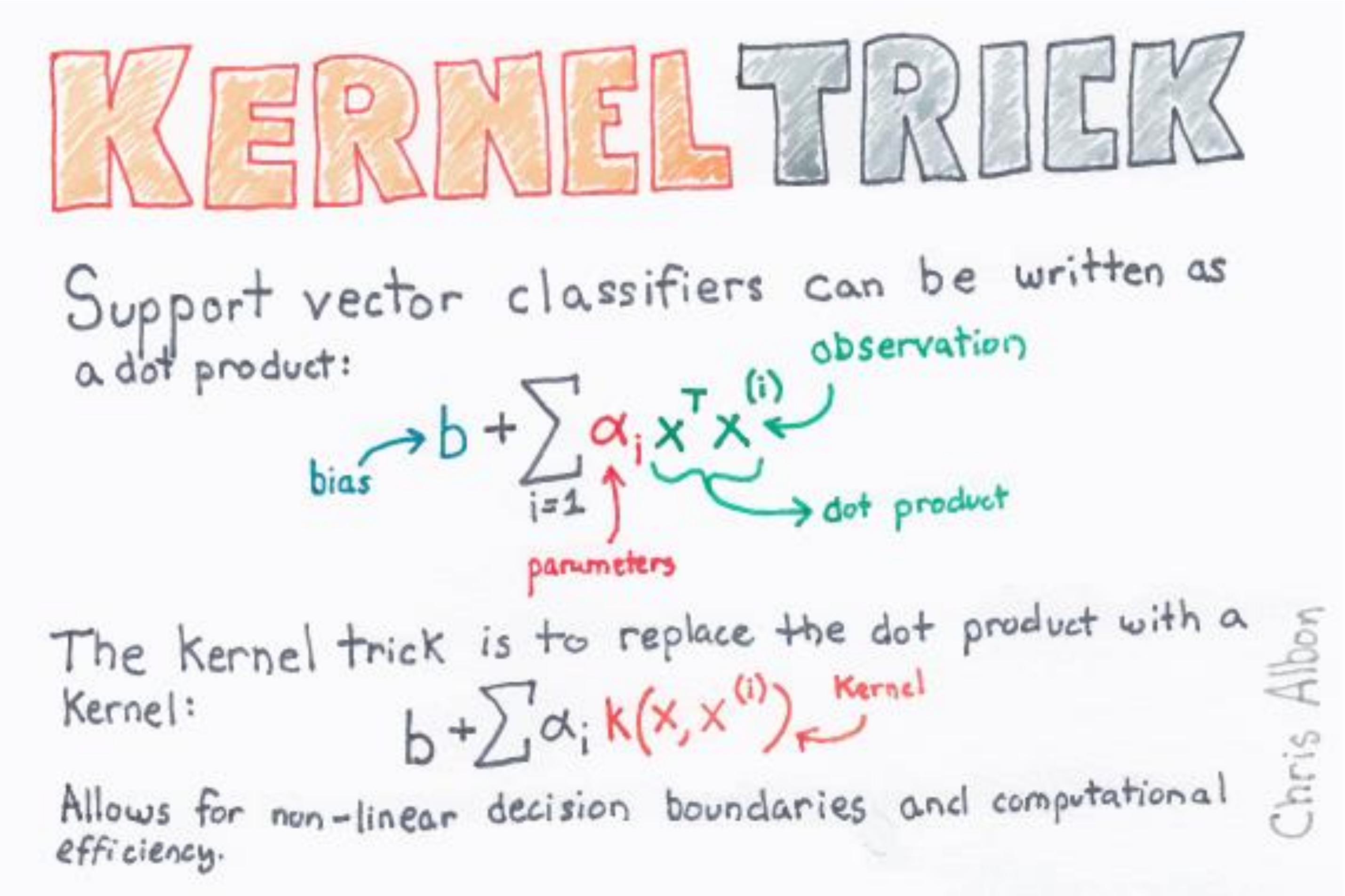
By combining the **soft margin** (tolerance of misclassifications) and **kernel trick** together, Support Vector Machine is able to structure the decision boundary for linear non-separable cases.



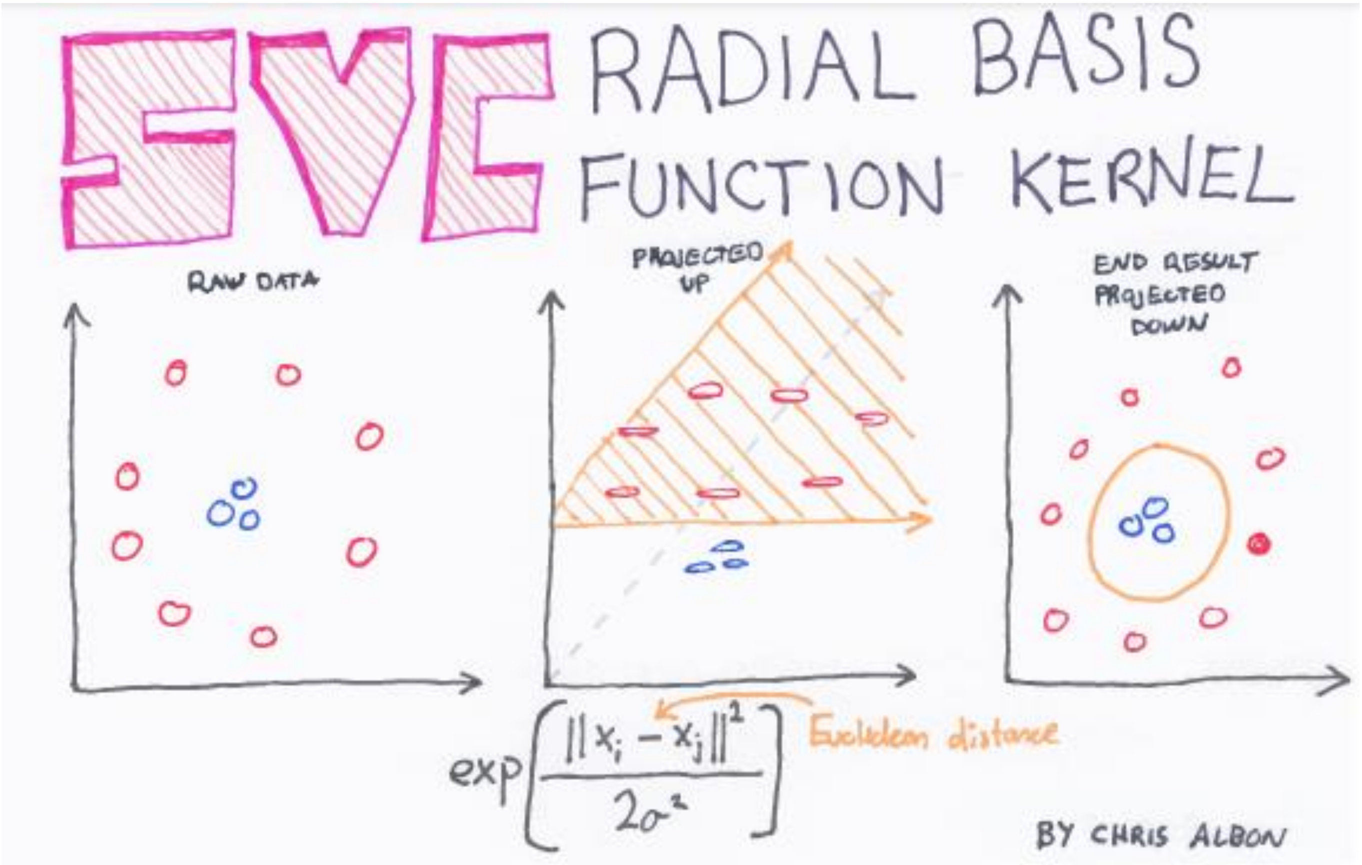
Support Vector Machines



Support Vector Machines



Support Vector Machines



Support Vector Machine Advantages

Effective on datasets with multiple features, like financial or medical data.

Effective in cases where number of features is greater than the number of data points.

Uses a subset of training points in the decision function called support vectors which makes it memory efficient.

Different kernel functions can be specified for the decision function. You can use common kernels, but it's also possible to specify custom kernels.

Support Vector Machine Disadvantages

If the number of features is a lot bigger than the number of data points, avoiding over-fitting when choosing kernel functions and regularization term is crucial.

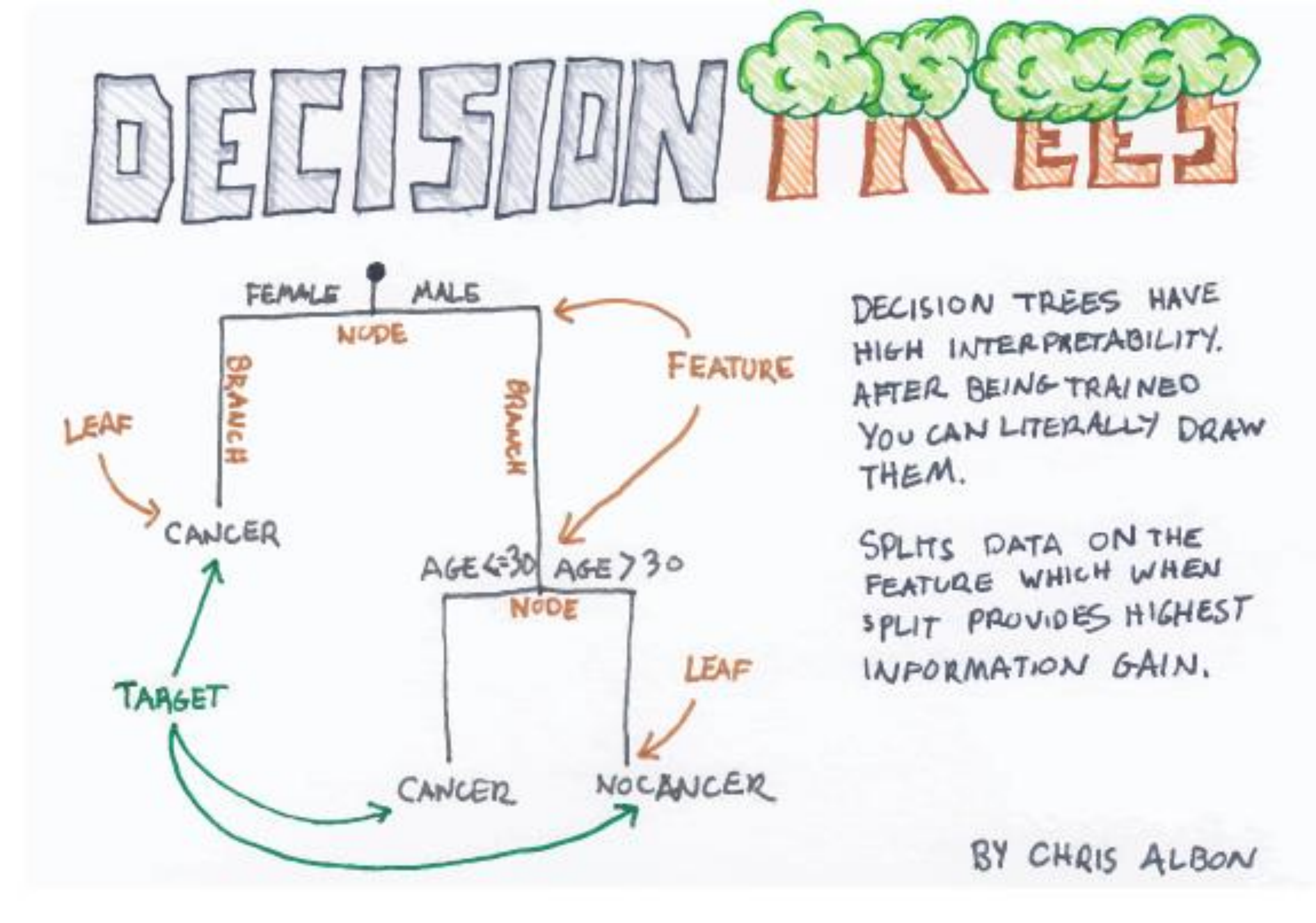
SVMs don't directly provide probability estimates. Those are calculated using an expensive five-fold cross-validation.

Works best on small sample sets because of its high training time.

Decision Trees

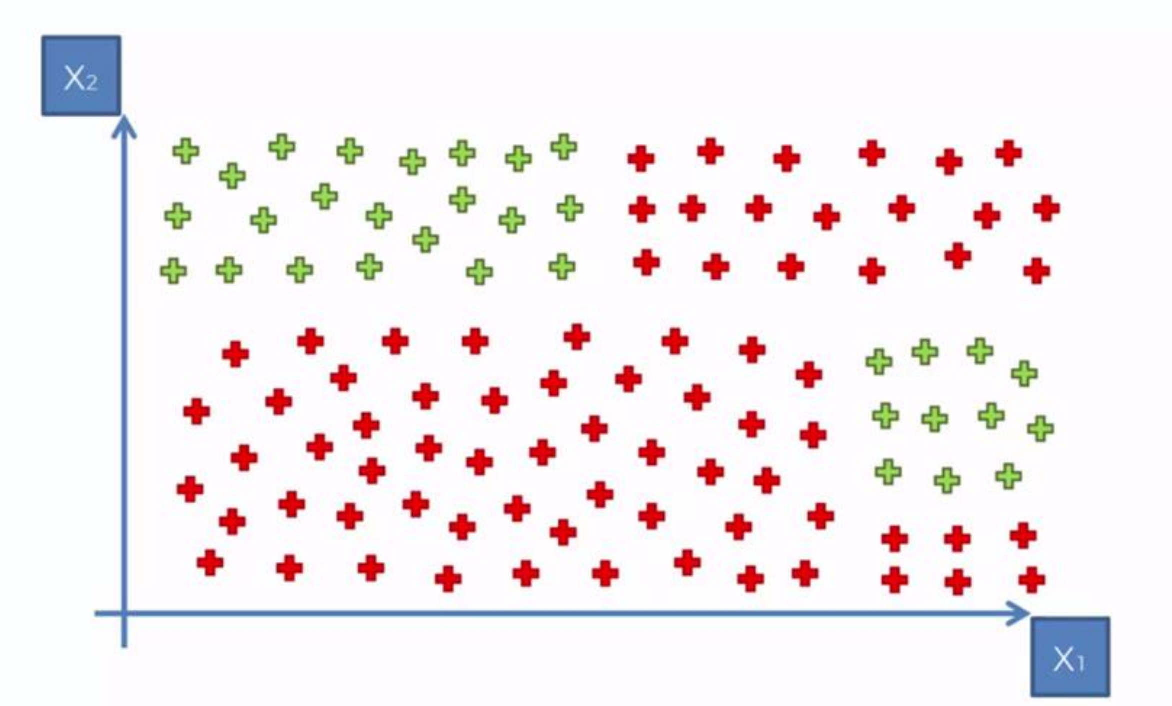
Decision tree builds classification or regression models in the form of a tree structure.

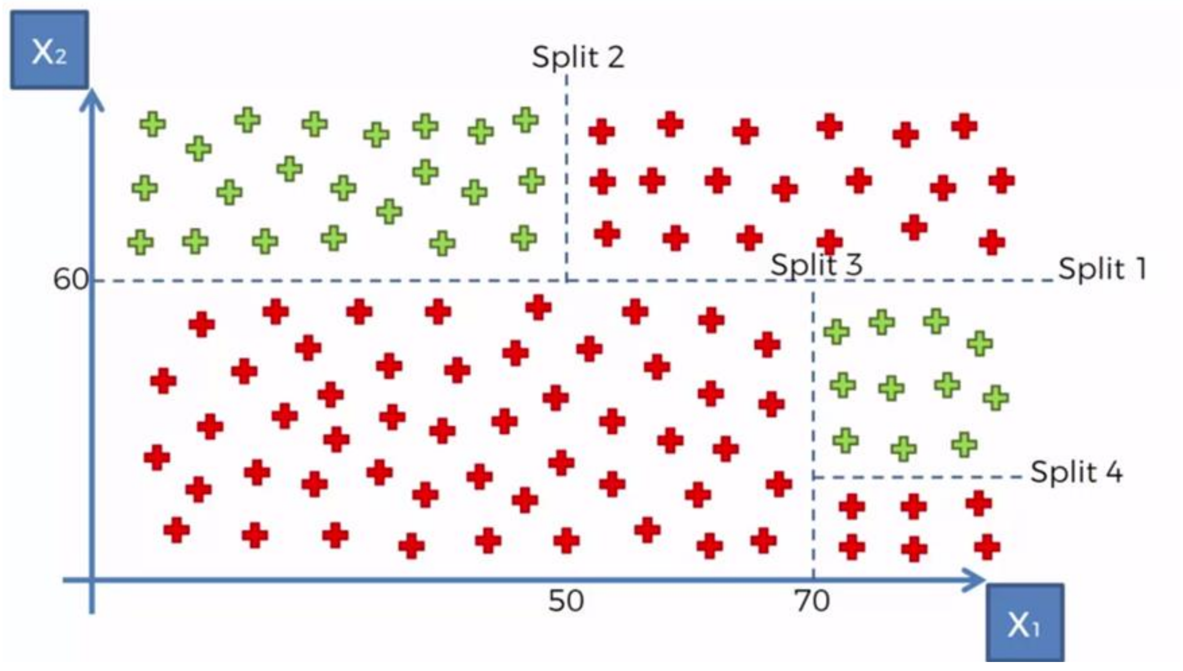
It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed



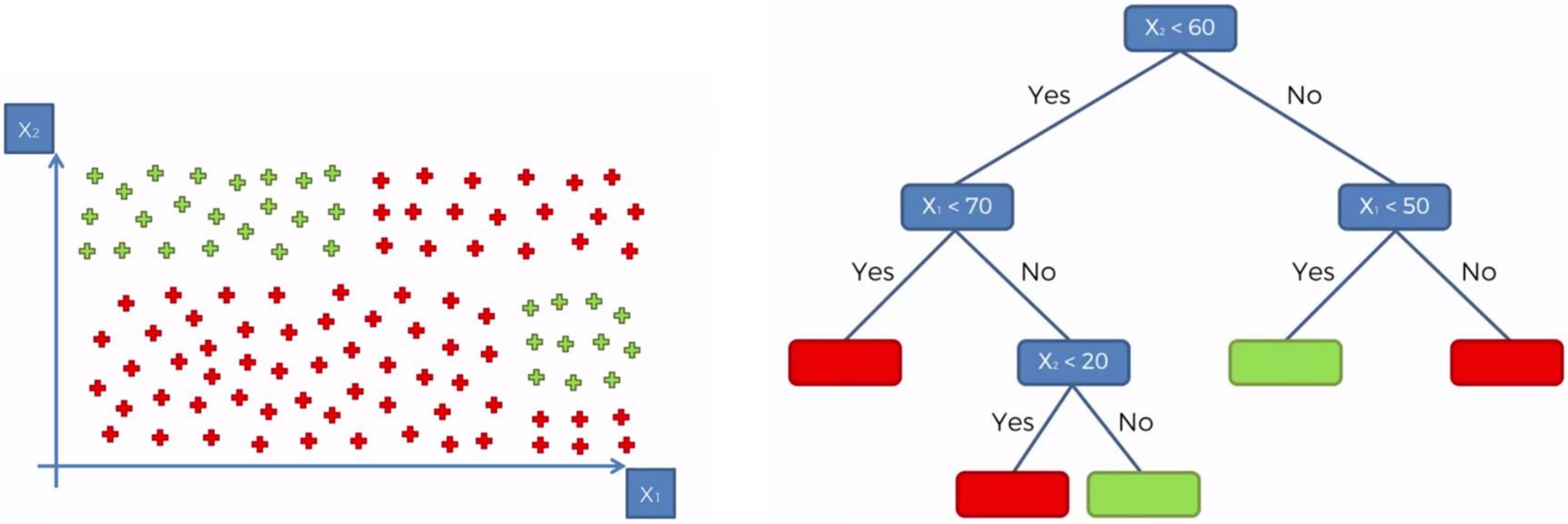
Decision Trees

Decision Trees

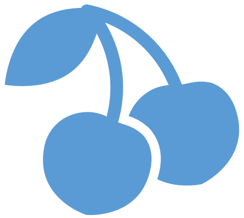




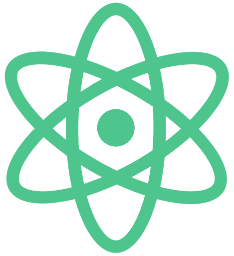
Decision Trees



Decision Trees – Split measures



Gini



Entropy

Information Gain

Decision Trees – Entropy

Entropy is the degree or amount of uncertainty in the randomness of elements. In other words, it is a measure of impurity.

Intuitively, it tells us about the predictability of a certain event.

Entropy calculates the homogeneity of a sample. If the sample is completely homogeneous the entropy is zero, and if the sample is equally divided it has an entropy of one.

Decision Trees – Gini Index

Gini Index: It is the measure of inequality of distribution. It says if we select two items from a population at random then they must be of same class and probability for this is 1 if population is pure.

It works with categorical target variable “Success” or “Failure”.

It performs only Binary splits

Lower the value of Gini, higher the homogeneity.

CART (Classification and Regression Tree) uses Gini method to create binary splits.

Decision Trees – Information Gain

Information Gain is simply a mathematical way to capture the amount of information one gains(or reduction in randomness) by picking a particular attribute

In a decision algorithm, we start at the tree root and split the data on the feature that results in the largest **information gain (IG)**. In other words, IG tells us how important a given attribute is.

Decision Trees – Advantages

It can capture non-linear relationships

Easy to understand, interpret, visualize.

It gives us and a good idea about the relative importance of attributes.

Decision tree is non-parametric

Decision Trees – Disadvantages

Decision tree often involves higher time to train the model.

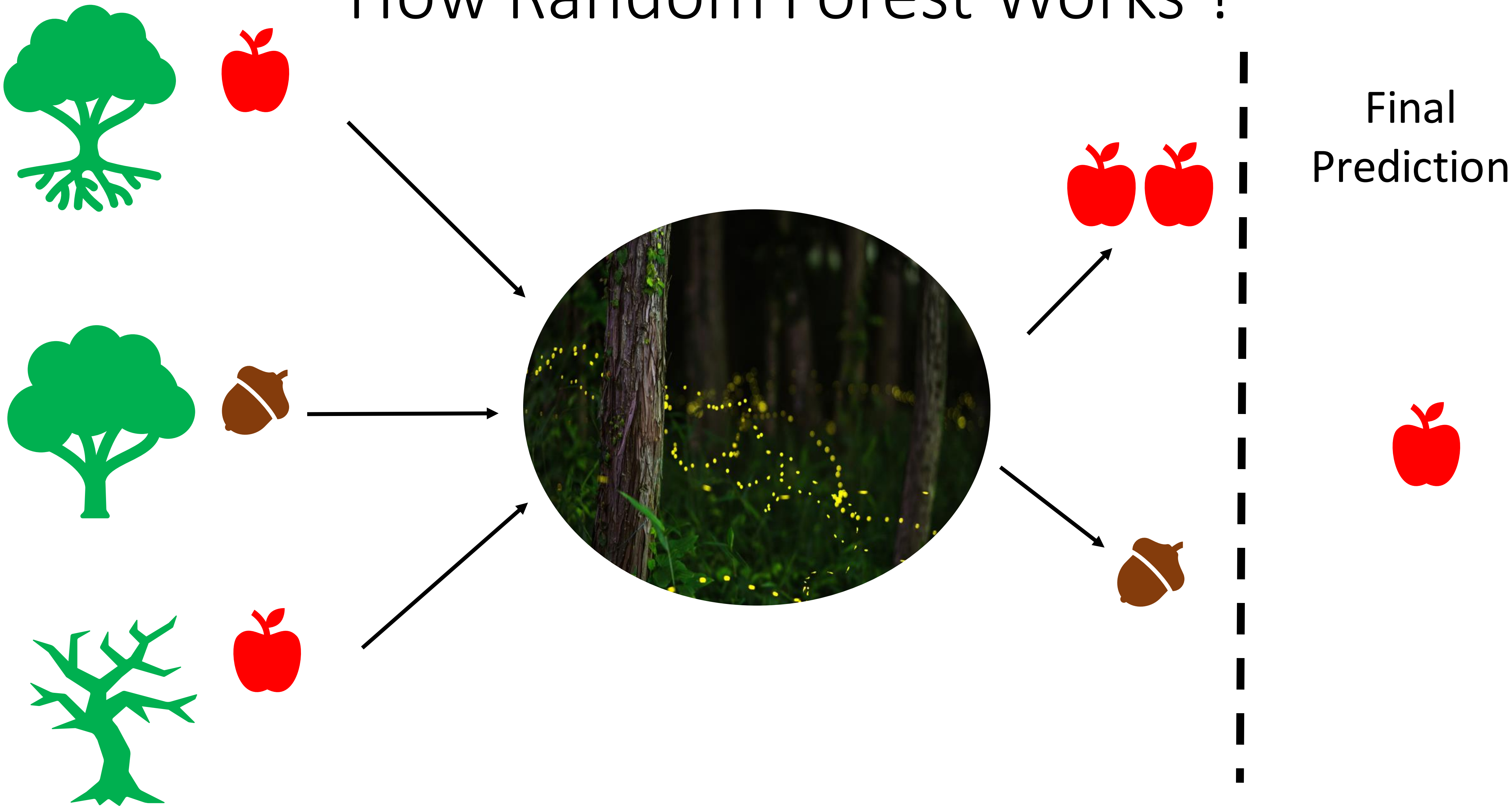
A small change in the data can cause a large change in the structure of the decision tree causing instability.

For a Decision tree sometimes calculation can go far more complex compared to other algorithms

Random Forest

- Random forest is a supervised learning algorithm.
- The "forest" it builds is an ensemble of decision trees, usually trained with the “bagging” method.
- The general idea of the bagging method is that a combination of learning models increases the overall result.

How Random Forest Works ?



Section - 6

UnSupervised Learning

Clustering

Cluster analysis or **clustering** is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar (in some sense) to each other than to those in other groups (clusters).

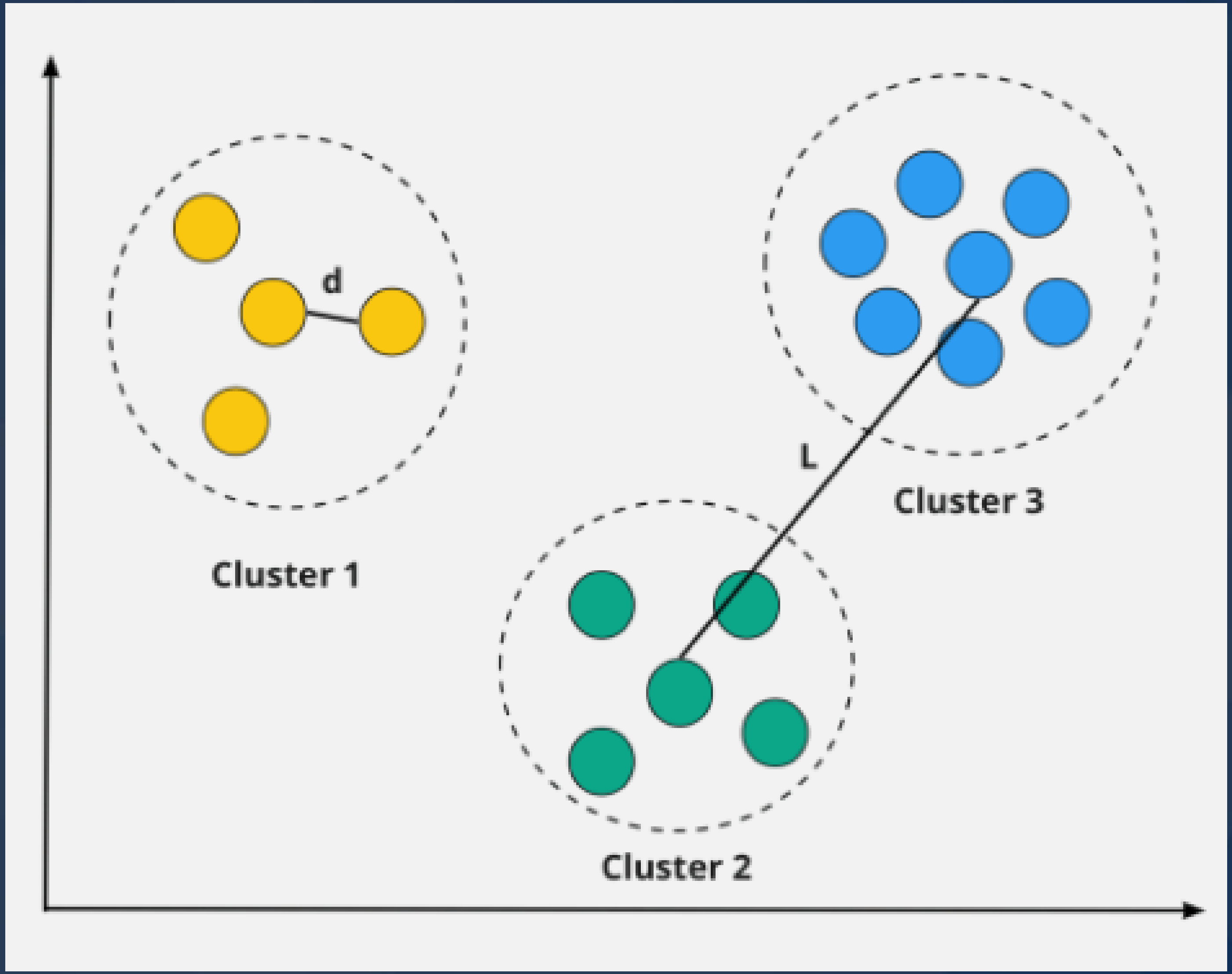
Types of Clustering

Centroid based clustering

Density based clustering

Hierarchical clustering

K-Means Clustering



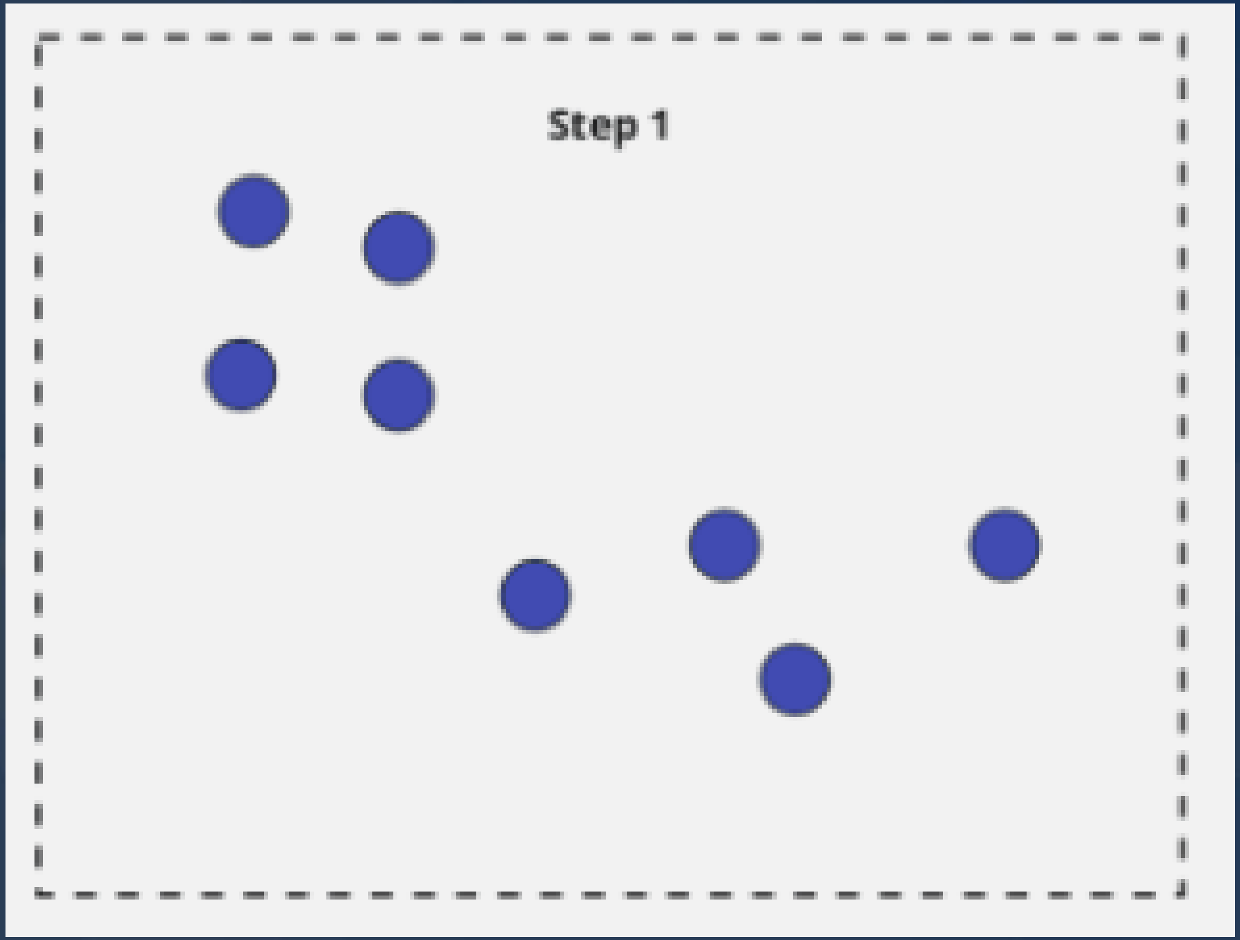
K-Means Algorithm

- First it selects *k* number of objects at random from the set of n objects.
- These *k* objects are treated as the centroids or center of gravities of *k* clusters.
- For each of the remaining objects, it is assigned to one of the closest centroid.
- Thus, it forms a collection of objects assigned to each centroid and is called a cluster.

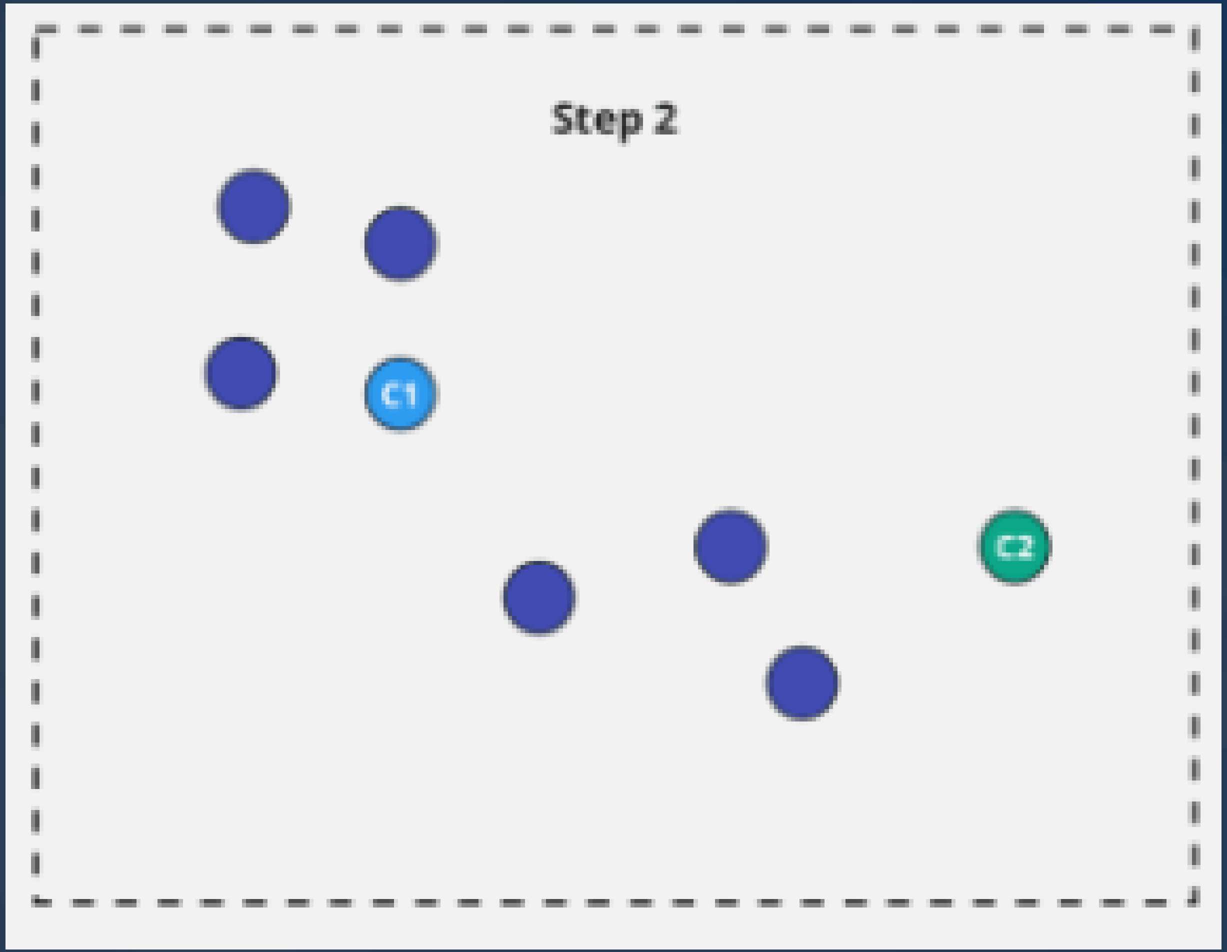
K-Means Algorithm

- Next, the centroid of each cluster is then updated (by calculating the mean values of attributes of each object).
- The assignment and update procedure is until it reaches some stopping criteria (such as, number of iteration, centroids remain unchanged or no assignment, etc.)

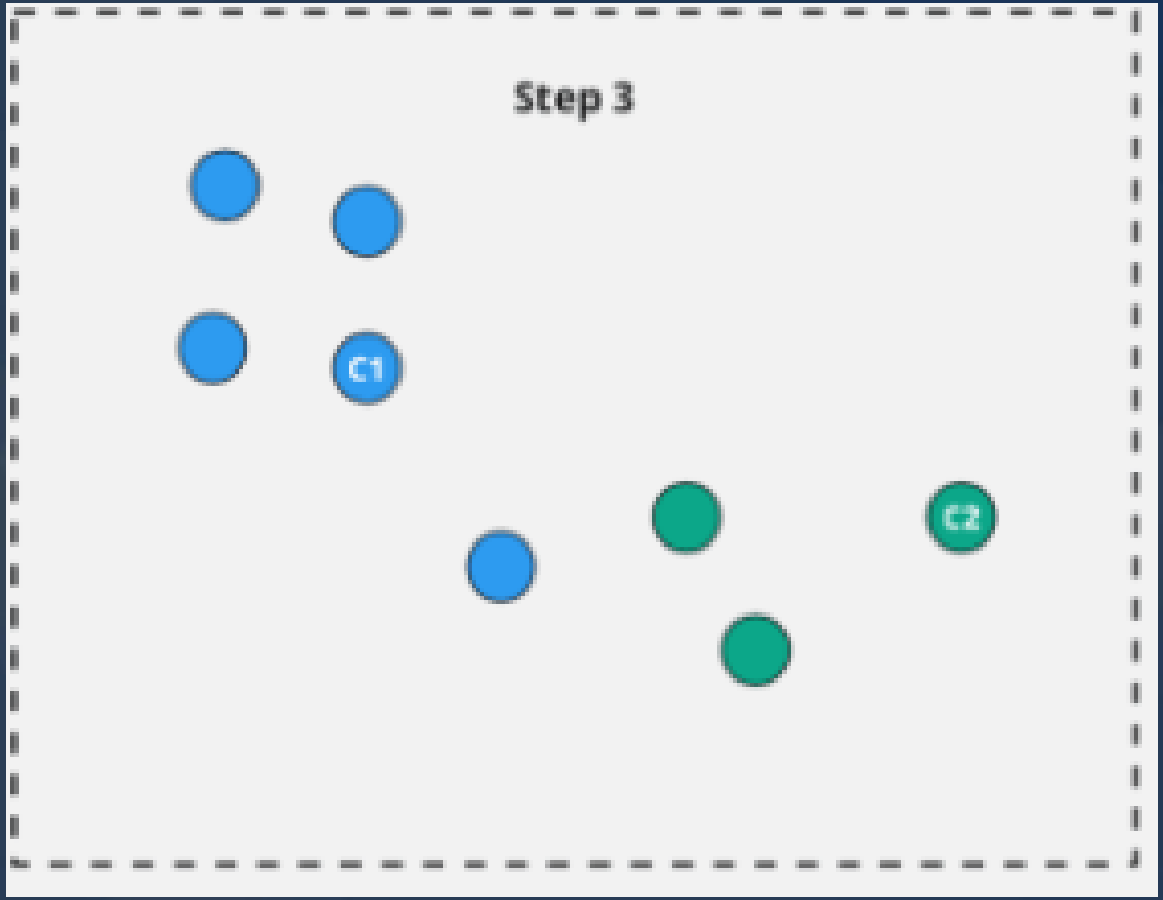
K-Means Algorithm



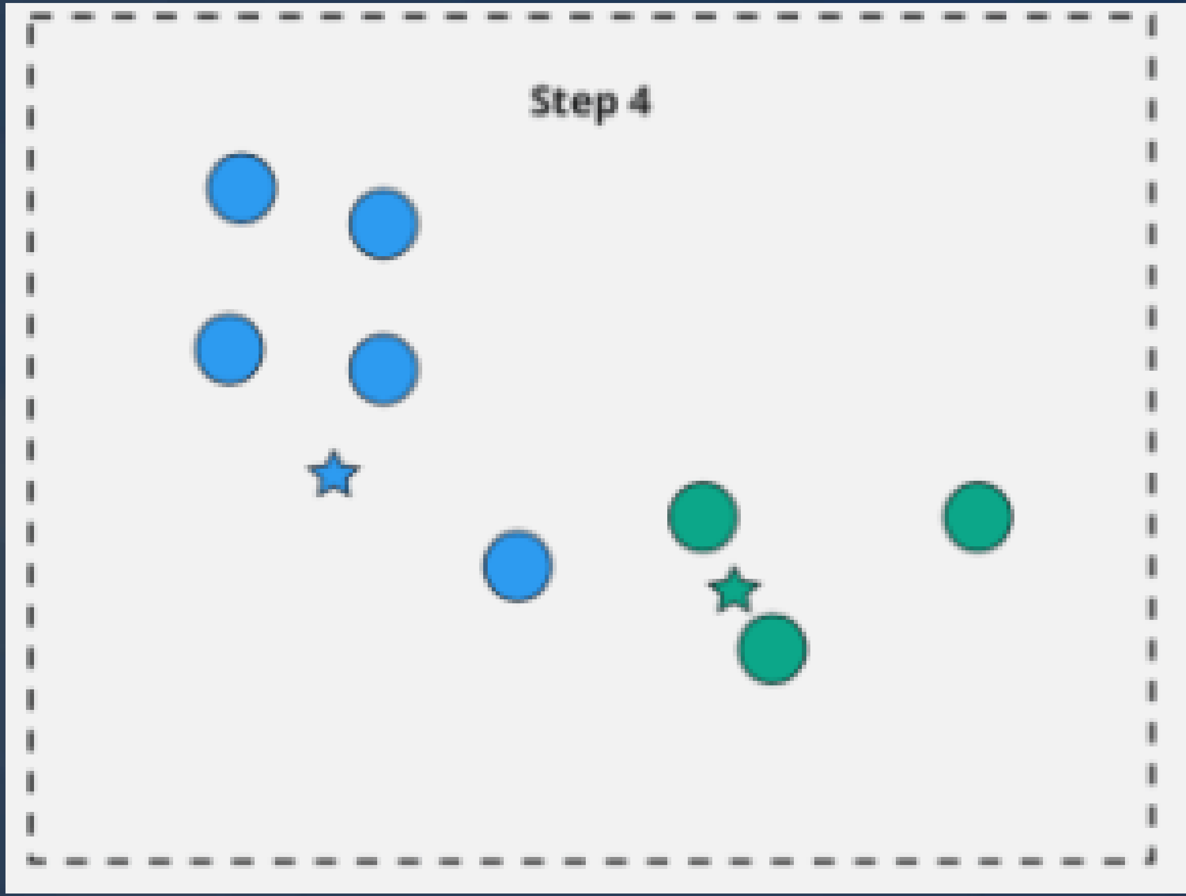
K-Means Algorithm



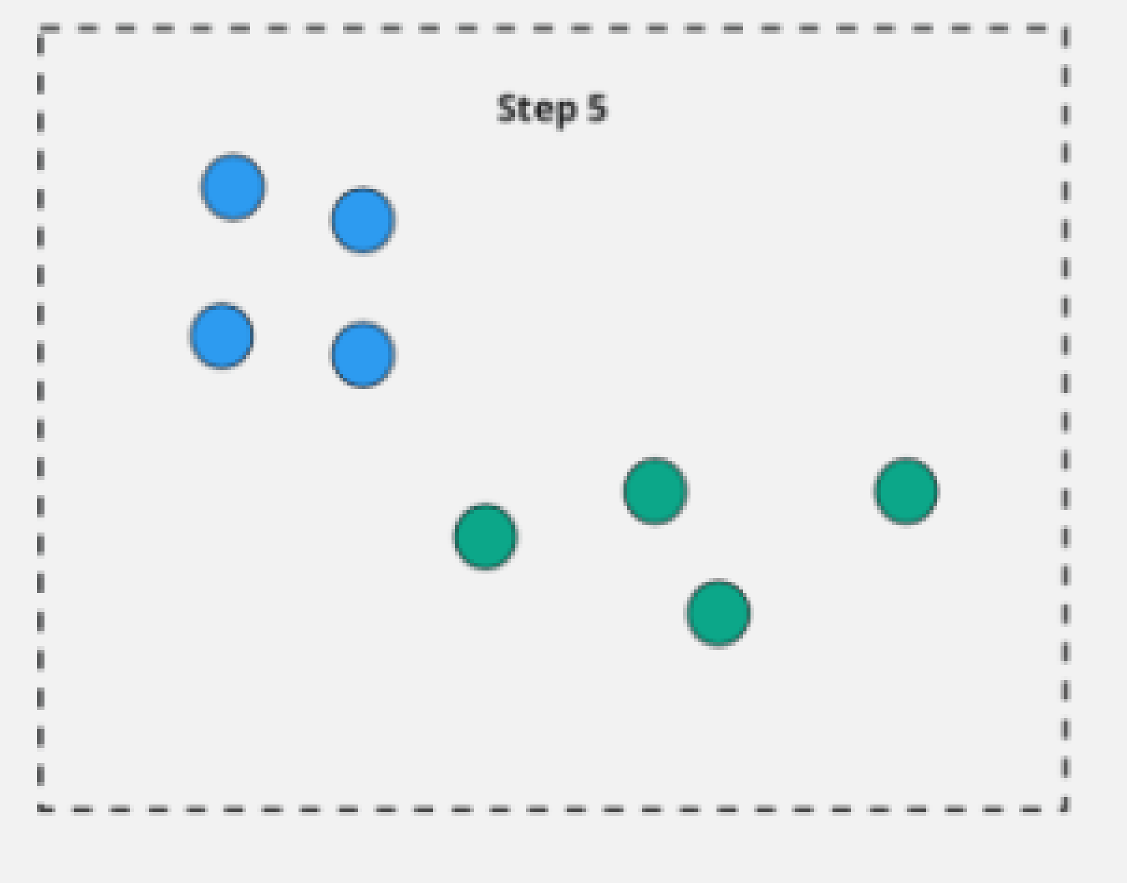
K-Means Algorithm



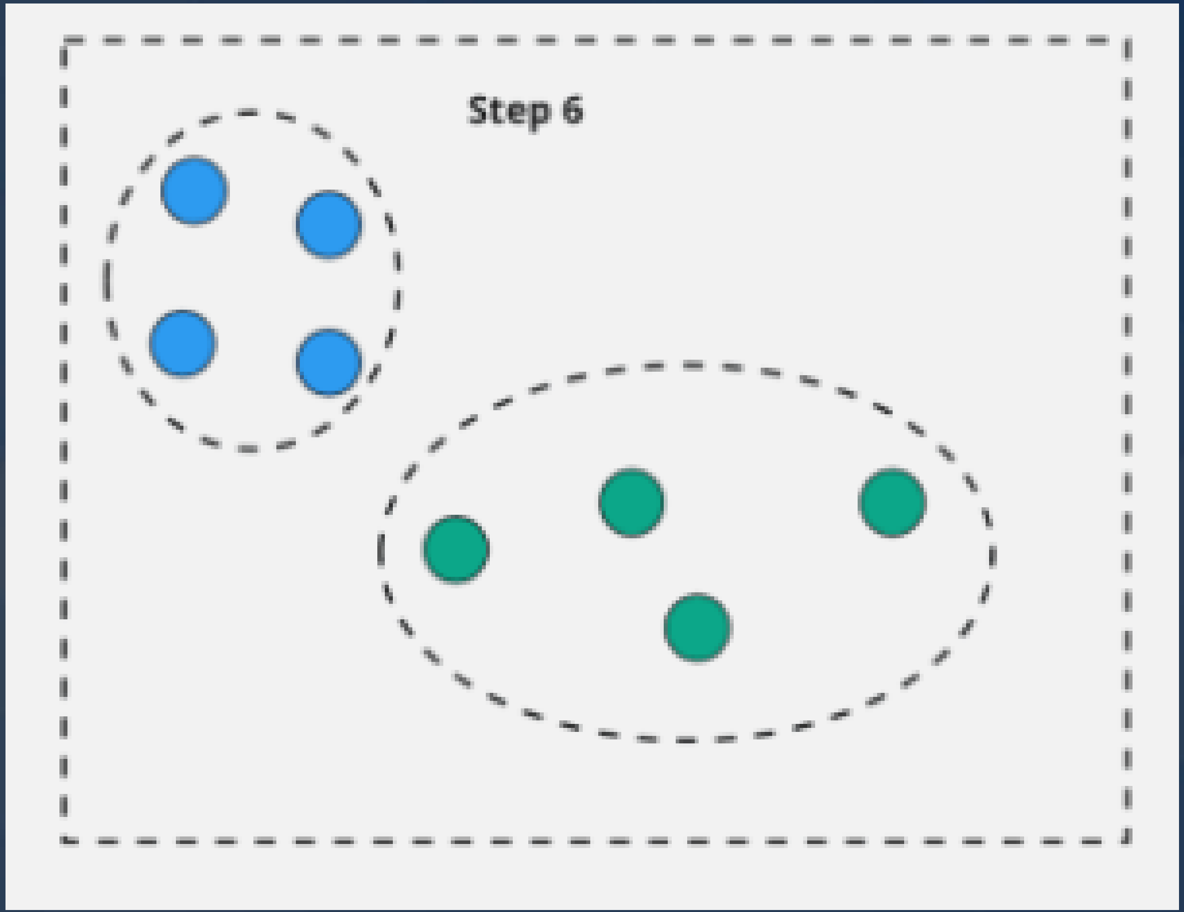
K-Means Algorithm



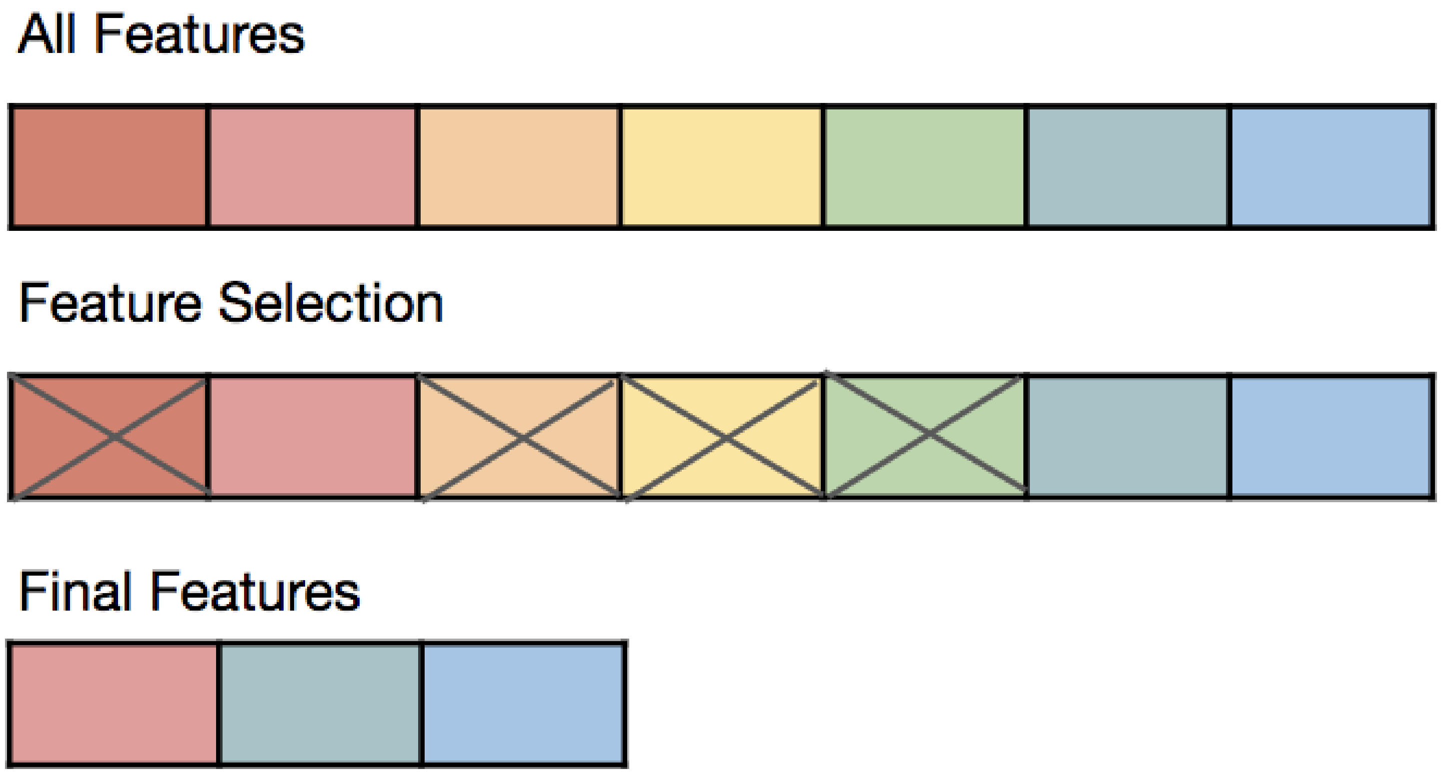
K-Means Algorithm



K-Means Algorithm



Feature Selection



Feature Selection

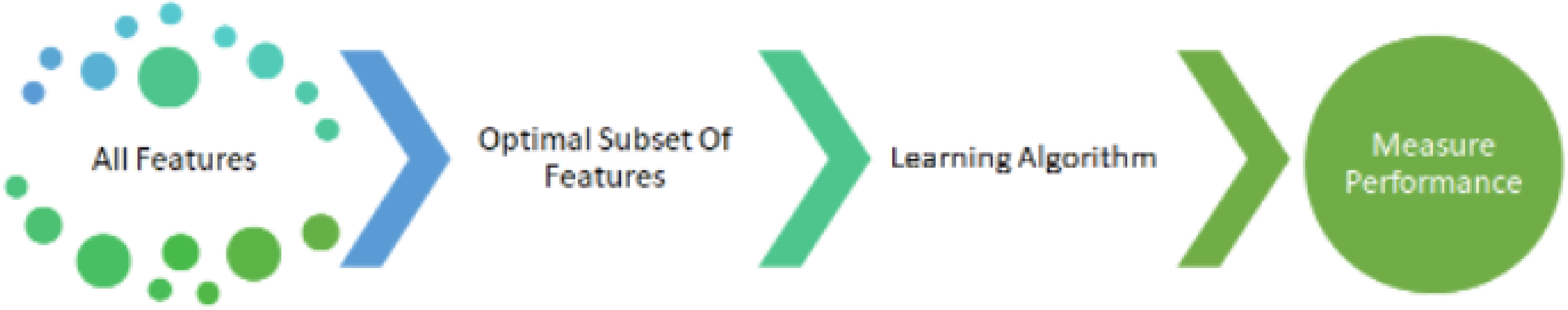
Filter Methods

Wrapper Methods

Embedded Methods

Feature Selection

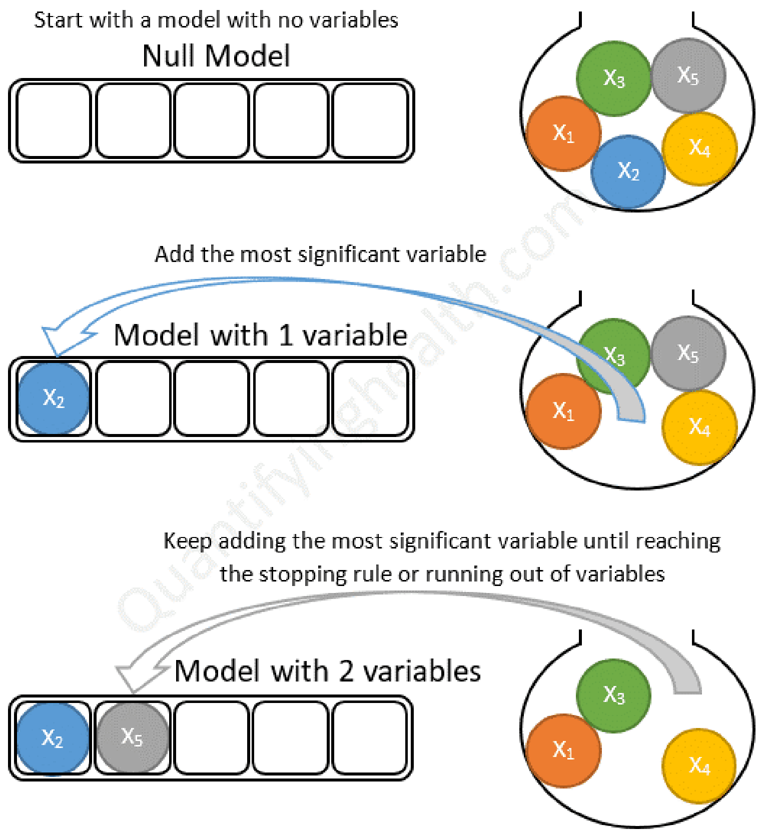
Filter Method



Feature Selection

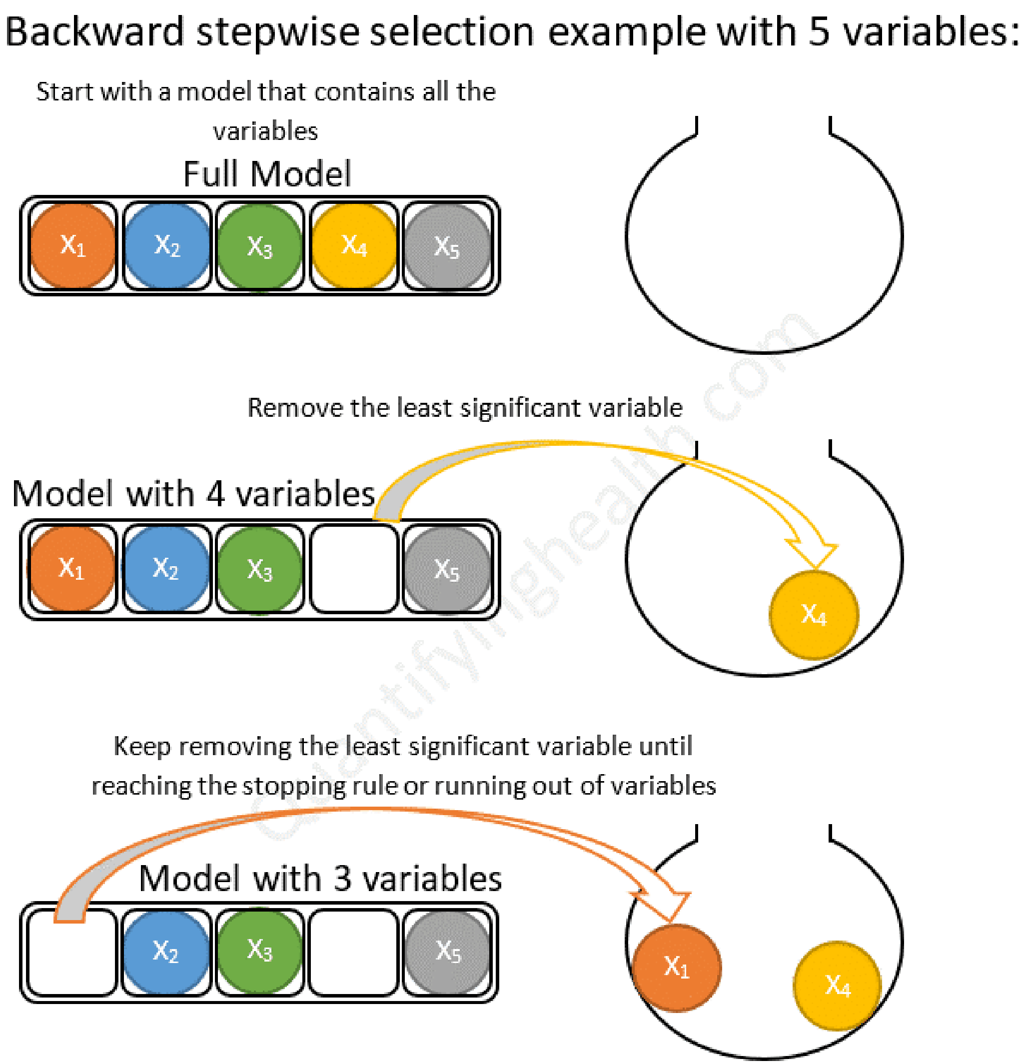
Wrapper Method – Forward Stepwise Selection

Forward stepwise selection example with 5 variables:



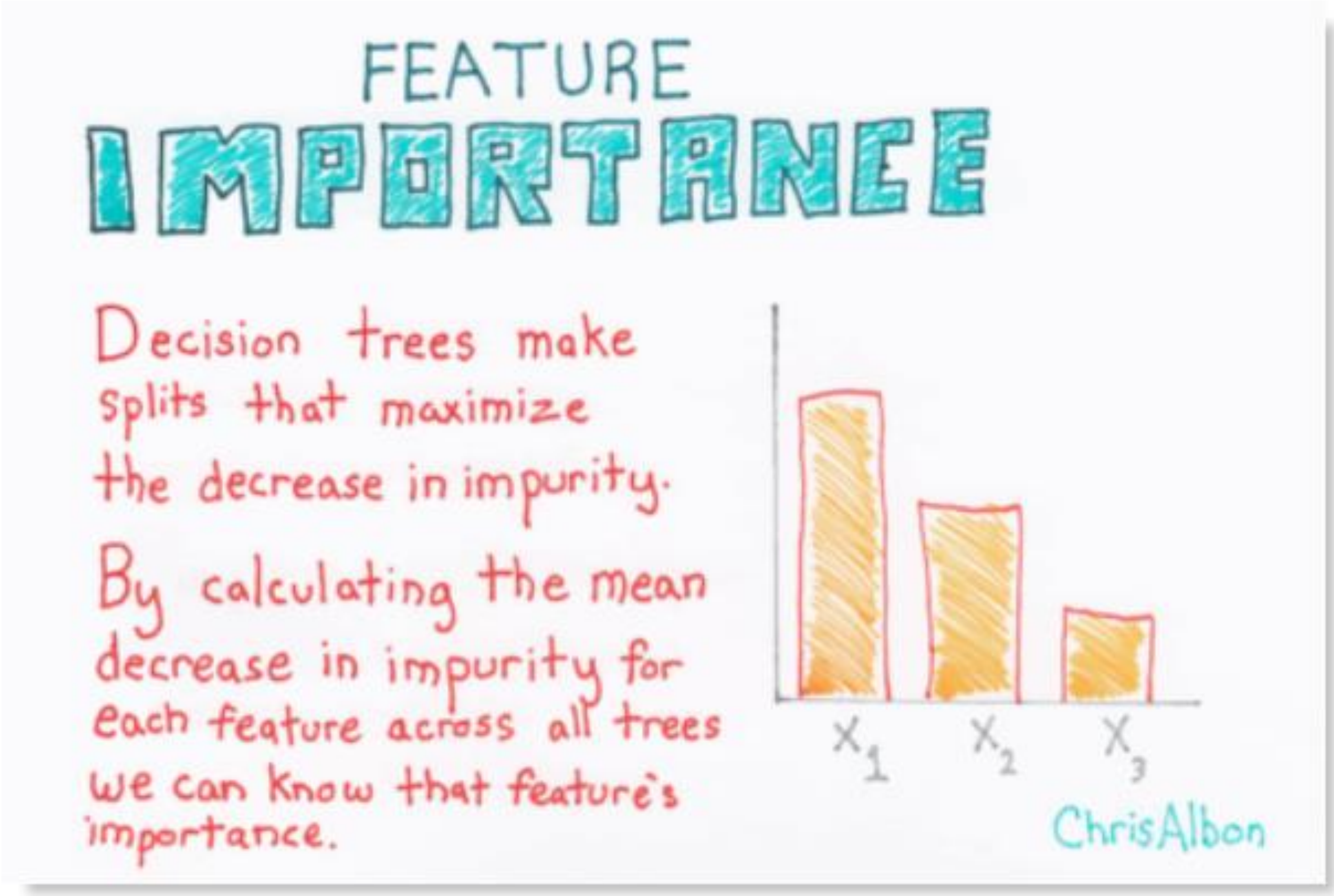
Wrapper Method – Backward Stepwise Selection

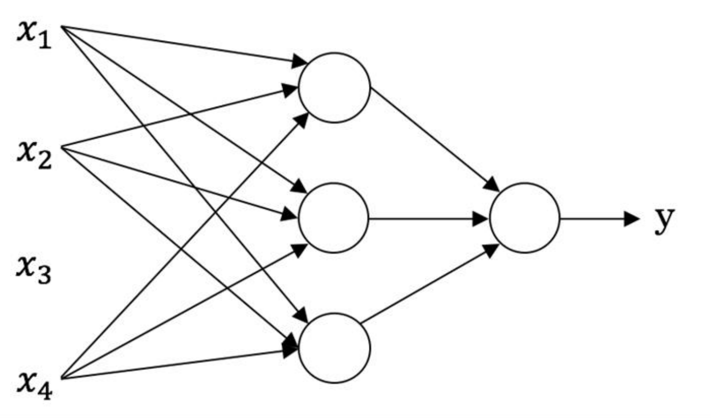
Feature Selection



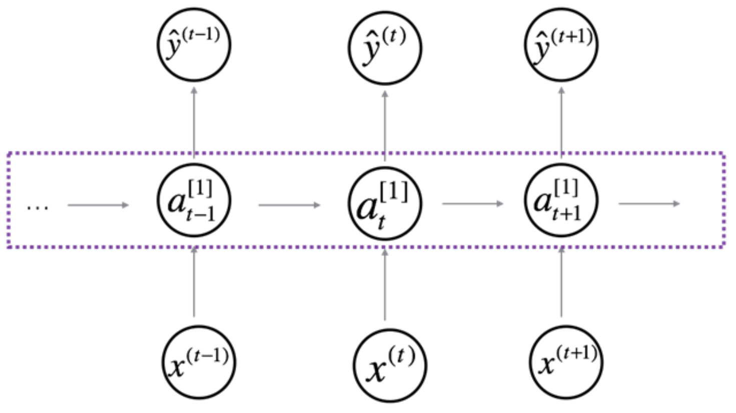
Embedded Methods– Feature Importance

Feature Selection

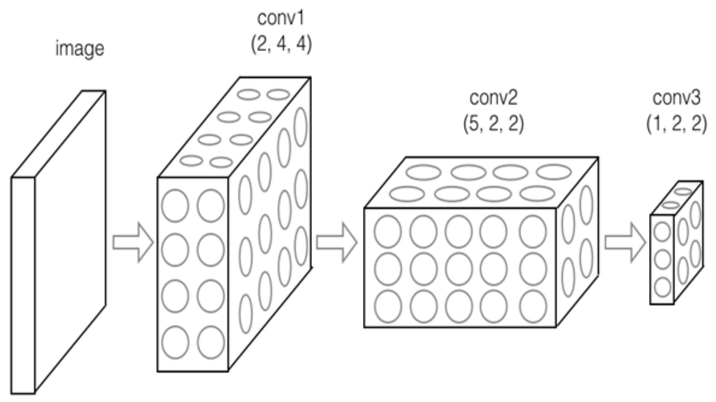




Standard NN



Recurrent NN

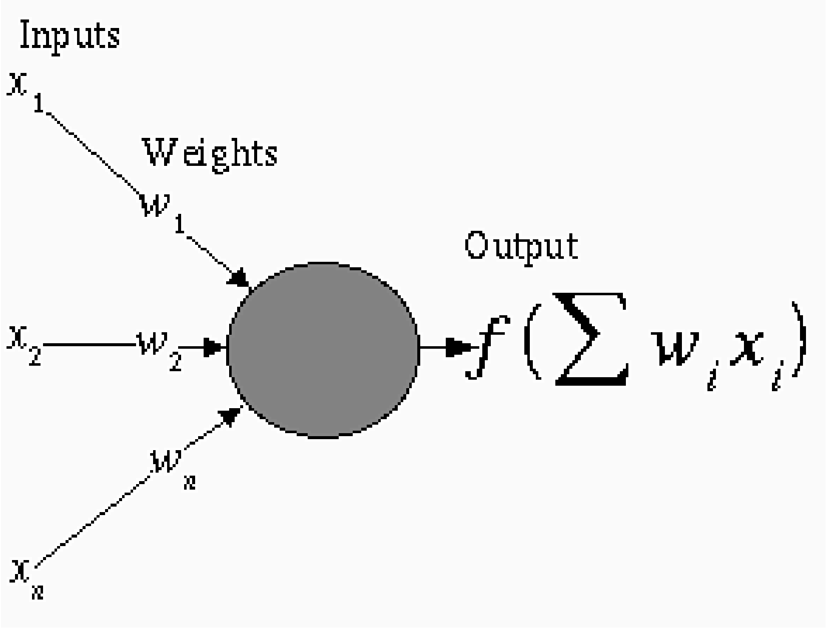


Convolutional NN

Neural Network Examples

Basic Neuron Model In A Feedforward Network

- **Inputs x_i** arrive through pre-synaptic connections
- Synaptic efficacy is modeled using real **weights w_i**
- The response of the neuron is a **nonlinear function *f*** of its weighted inputs



Inputs To Neurons

- Arise from other neurons or from outside the network
- Nodes whose inputs arise outside the network are called *input nodes* and simply copy values
- An input may *excite* or *inhibit* the response of the neuron to which it is applied, depending upon the weight of the connection

Weights

- Represent synaptic efficacy and may be *excitatory* or *inhibitory*
- Normally, positive weights are considered as excitatory while negative weights are thought of as inhibitory
- ***Learning*** is the process of modifying the weights in order to produce a network that performs some function

Output

- The response function is normally nonlinear
- Samples include
  - Sigmoid

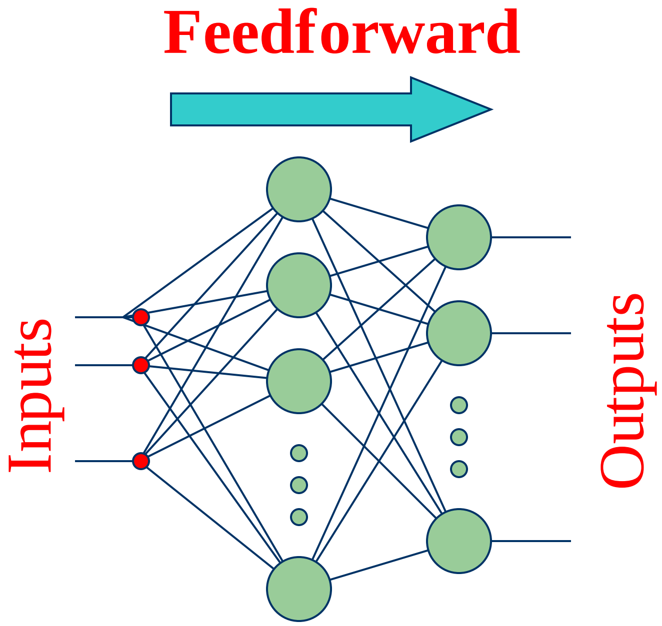
$$f(x) = \frac{1}{1+e^{-\lambda x}}$$

  - Piecewise linear

$$f(x) = \begin{cases} x, if \ \ x \geq \theta \\ 0, if \ \ x < \theta \end{cases}$$

Apply Inputs From A Pattern

- Apply the value of each input parameter to each input node
- Input nodes compute only the identity function



Calculate Outputs For Each Neuron Based On The Pattern

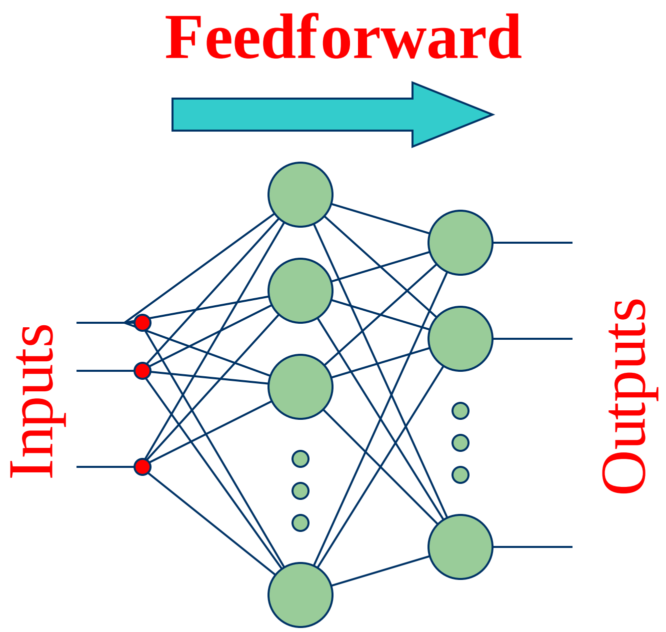
- The output from neuron j for pattern p is O_{pj} where

$$O_{pj}(net_j) = \frac{1}{1+e^{-\lambda net_j}}$$

and

$$net_j = bias * W_{bias} + \sum_k O_{pk} W_{kj}$$

k ranges over the input indices and W_{jk} is the weight on the connection from input k to neuron j



Calculate The Error Signal For Each Output Neuron

- The output neuron error signal δ_{pj} is given by $\delta_{pj}=(T_{pj}-O_{pj})\, O_{pj}\, (1-O_{pj})$
- T_{pj} is the target value of output neuron j for pattern p
- O_{pj} is the actual output value of output neuron j for pattern p

Calculate The Error Signal For Each Hidden Neuron

- The hidden neuron error signal δ_{pj} is given by

$$\delta_{pj} = O_{pj}(1 - O_{pj}) \sum_k \delta_{pk} W_{kj}$$

where δ_{pk} is the error signal of a post-synaptic neuron k and W_{kj} is the weight of the connection from hidden neuron j to the post-synaptic neuron k

Calculate And Apply Weight Adjustments

- Compute weight adjustments ΔW_{ji} at time t by

$\Delta W_{ji}(t) = \eta\ \delta_{pj}\ O_{pi}$

- Apply weight adjustments according to

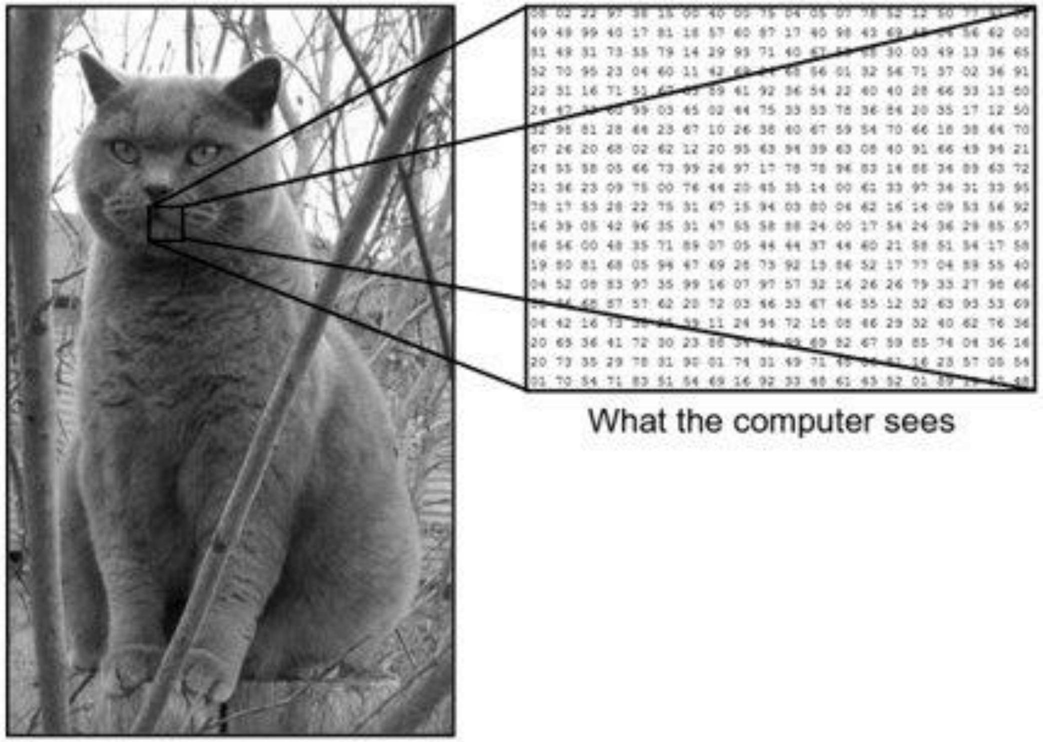
$W_{ji}(t+1) = W_{ji}(t) + \Delta W_{ji}(t)$

- *Some add a momentum term* $\alpha * \Delta W_{ji}(t-1)$

Image classification

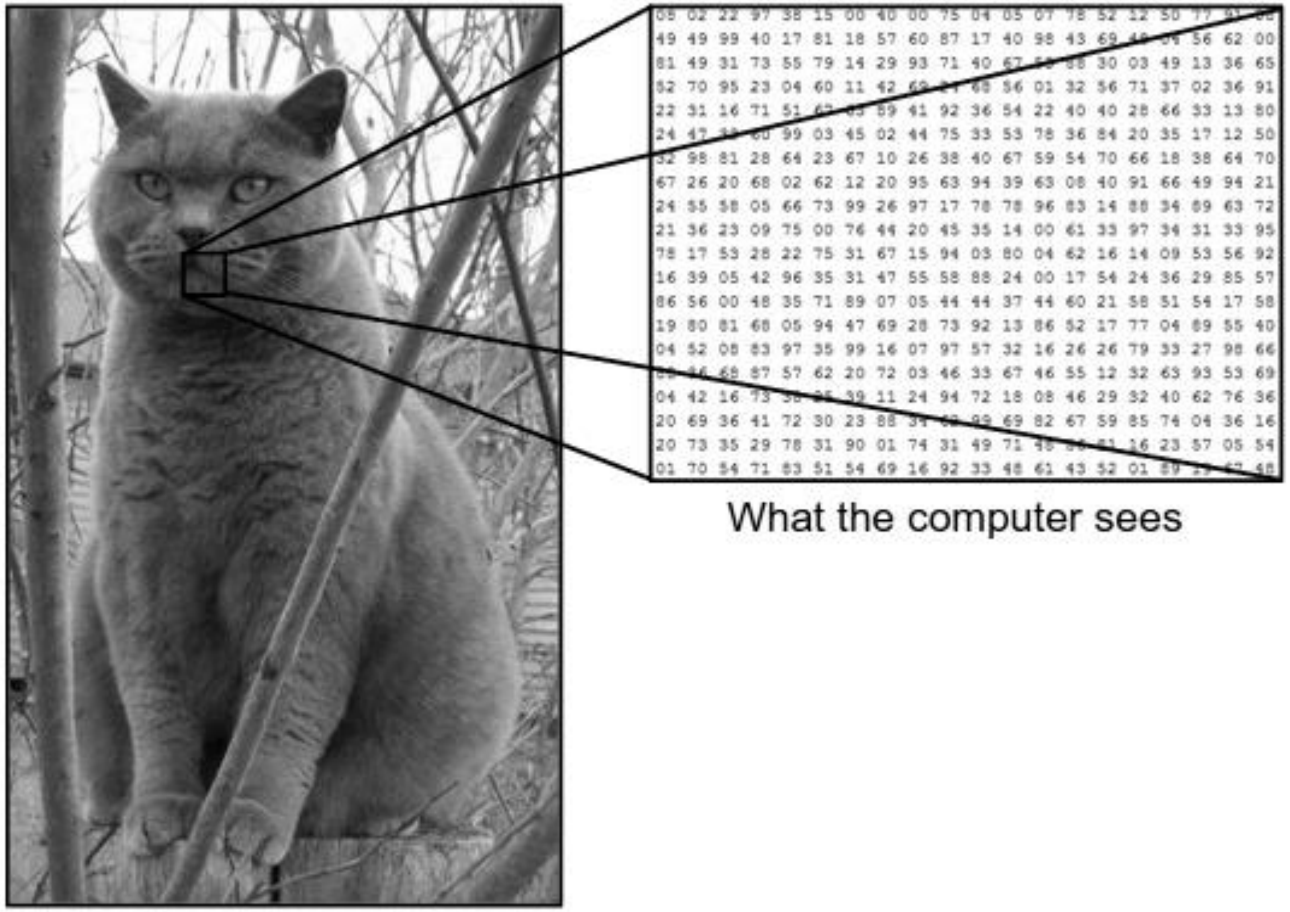
- Image classification is the task of taking an input image and outputting a
- class or a probability of classes that best describes the image
- • For humans, this task is one of the first skills we learn and it comes naturally and effortlessly as adults
- • Being able to quickly recognize patterns, generalize from prior knowledge, and adapt to different image environments are difficult tasks for machines

Image classification

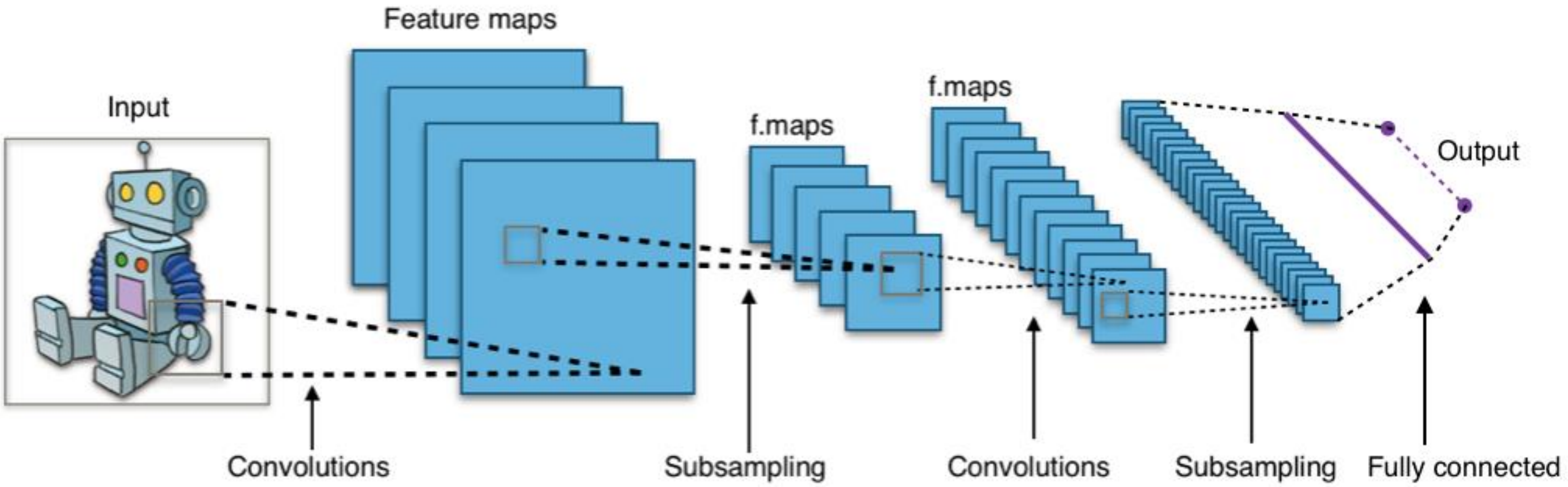


Convolution Neural Networks

A **Convolutional Neural Network (ConvNet/CNN)** is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other

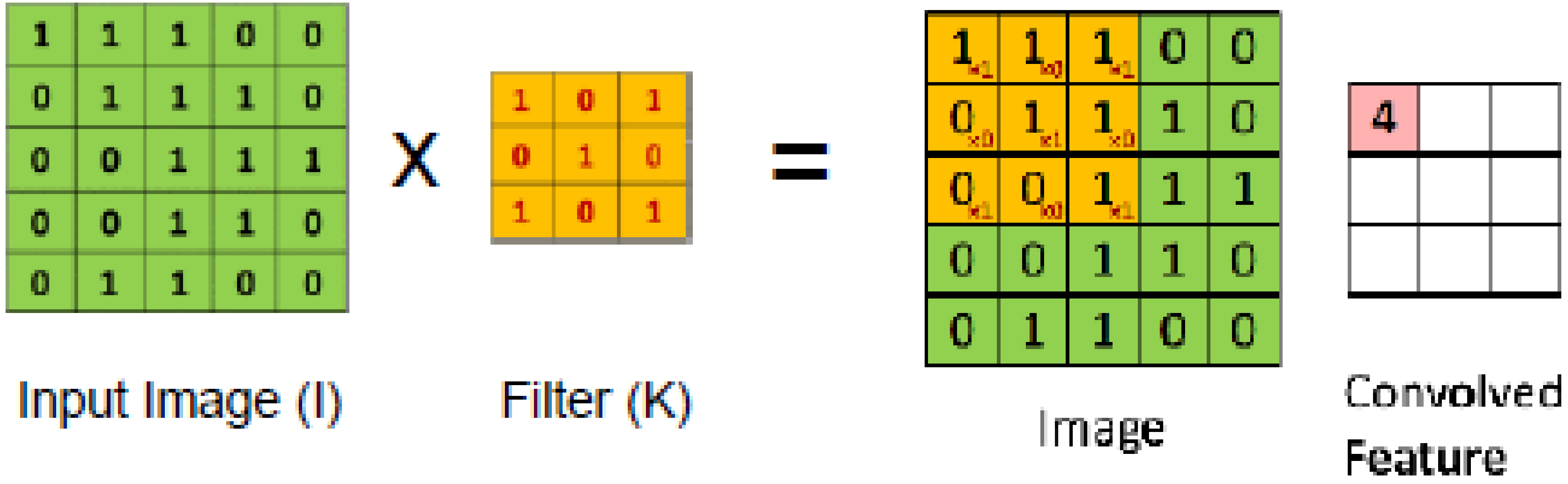


CNN - Architecture



CNN – Convolution Operator

- The 3×3 matrix (K) is called a **‘filter‘** or **‘kernel’** or ‘f**eature detector**’
- and the matrix formed by sliding the filter over the image and
- computing the dot product is called the ‘Convolved Feature’ or
- ‘Activation Map’ or the ‘Feature Map‘.



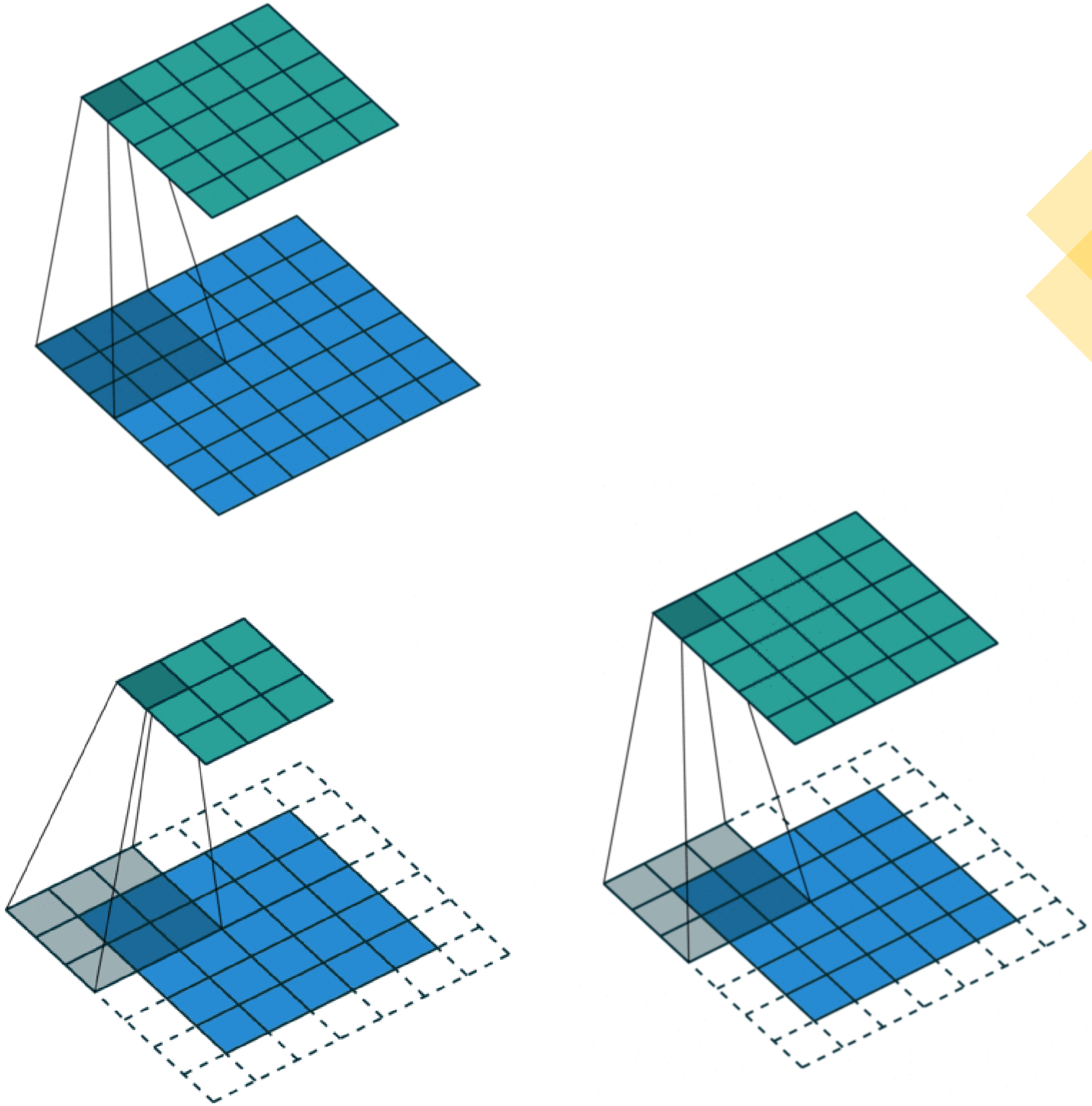
CNN – Convolution Operator

Different filters will produce different Feature Maps for the same input image. For example:

Operation	Filter	Convolved Image
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur (approximation)	$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	

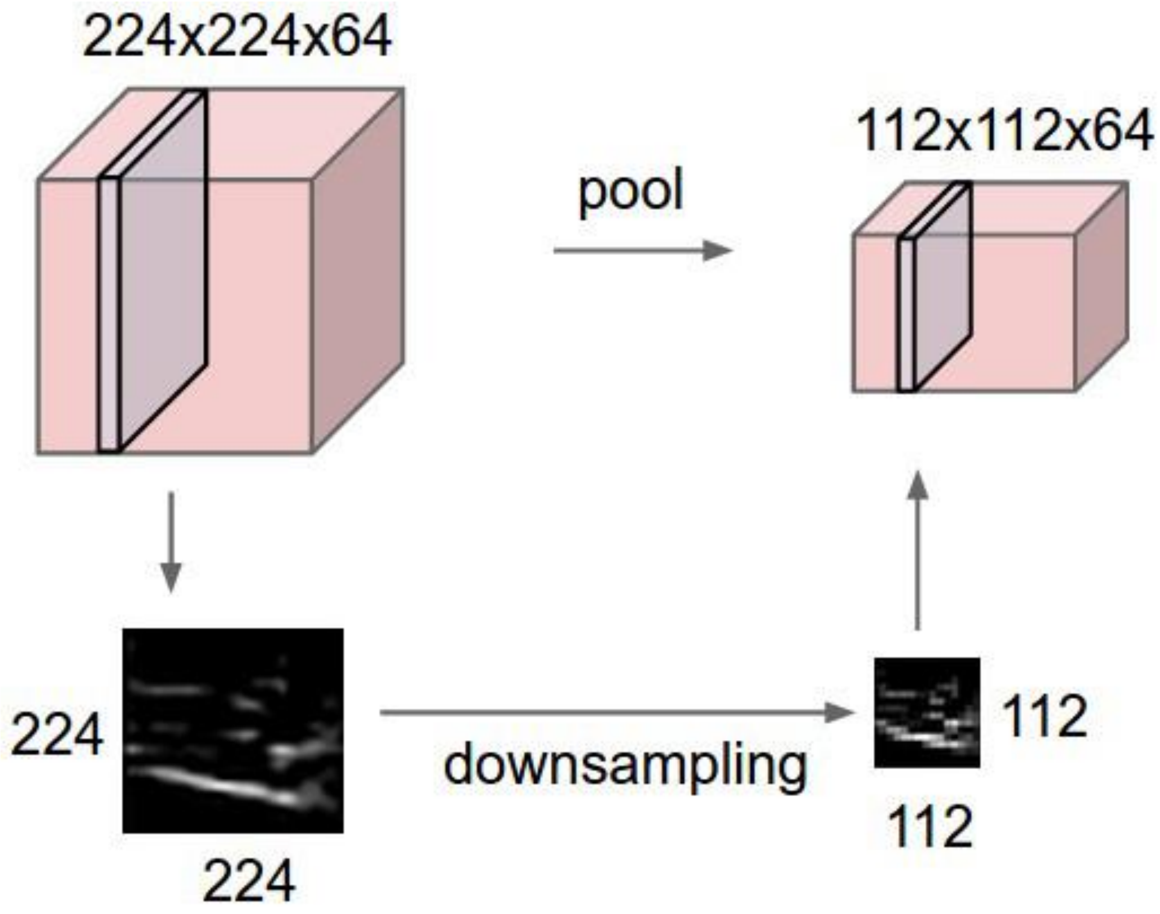
CNN – Convolution Layer

- In practice, a CNN learns the values of these filters on its own during
- the training process
- • Although we still need to specify parameters such as number of
- filters, filter size, padding, and stride before the training process



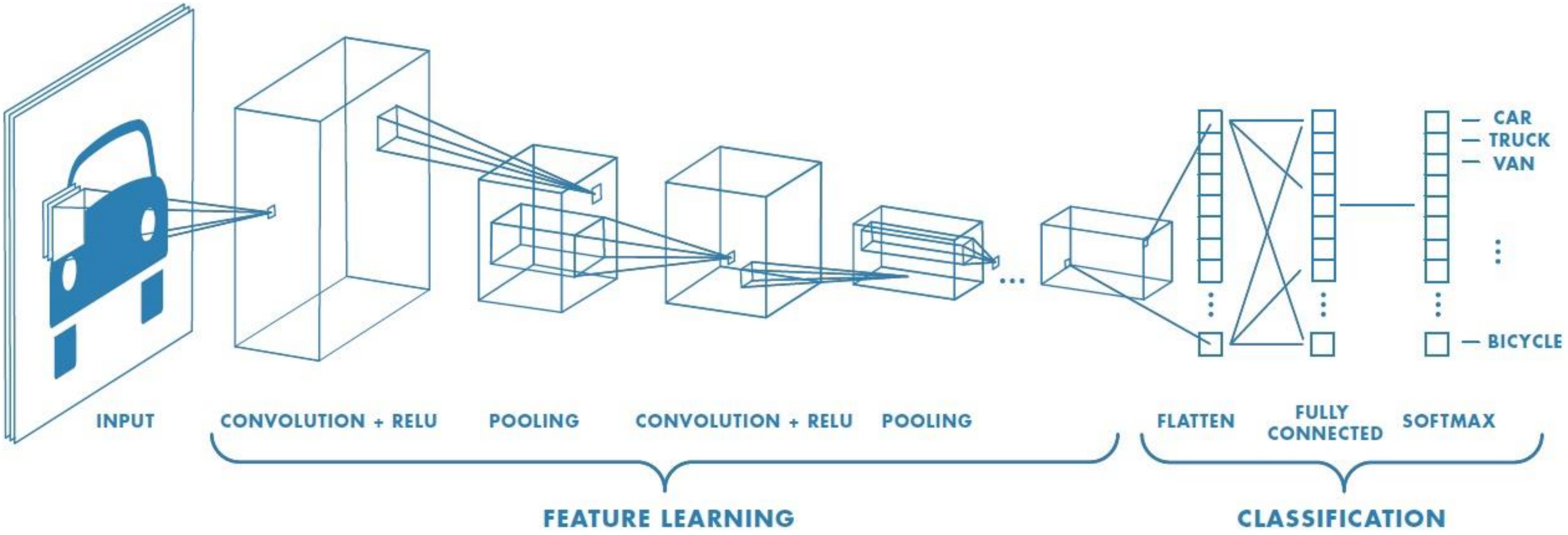
CNN – Pooling Layer

- In practice, a CNN learns the values of these filters on its own during
- the training process
- • Although we still need to specify parameters such as number of
- filters, filter size, padding, and stride before the training process



CNN – Fully Connected Layer

- Neurons in a fully connected layer have full connections to all activations in the previous layer, as seen in regular neural networks



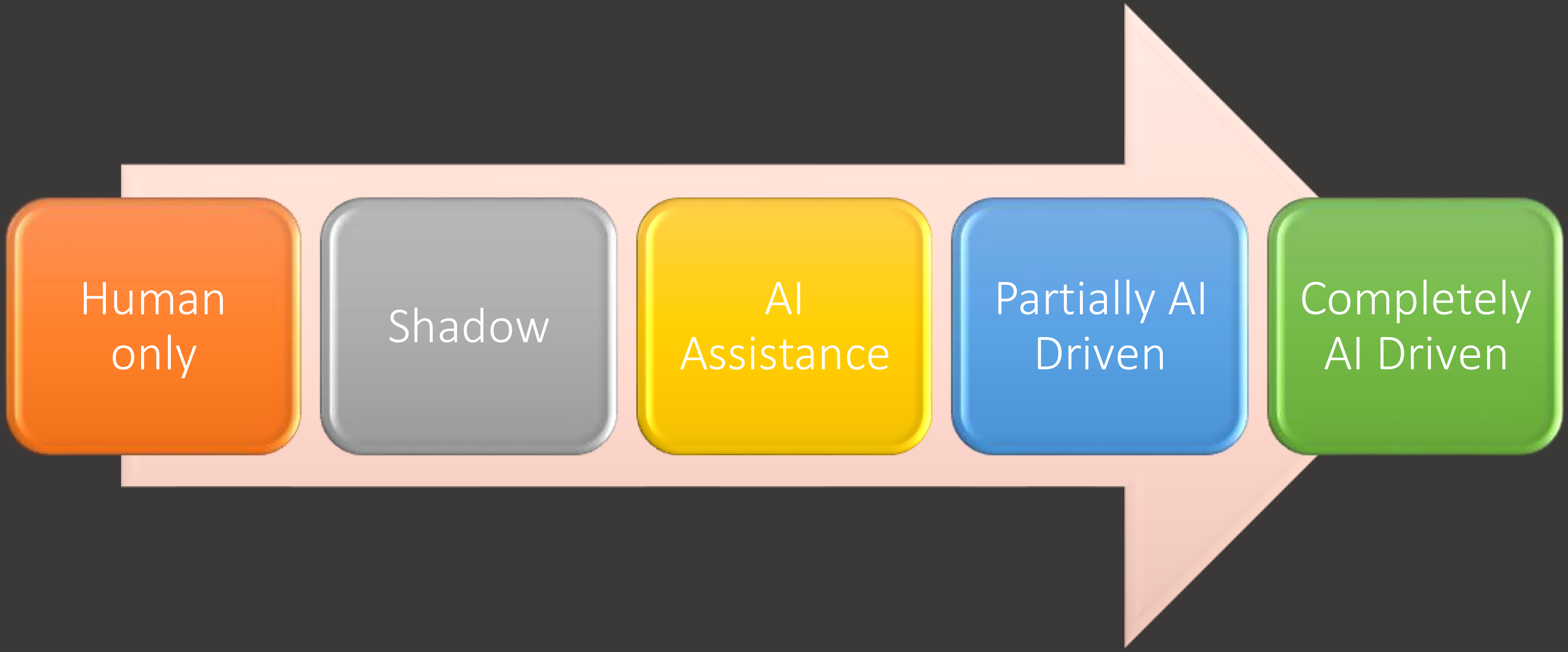
Dog vs Cats

- https://www.kaggle.com/competitions/dogs-vs-cats/data?select=test1.zip

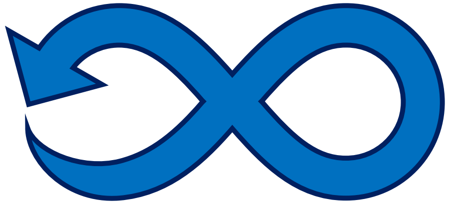
Section - 8

MLOps

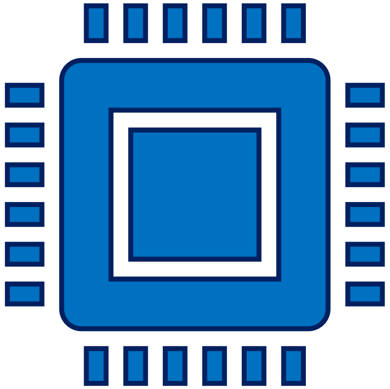
Stages of AI Automation



Modeling process

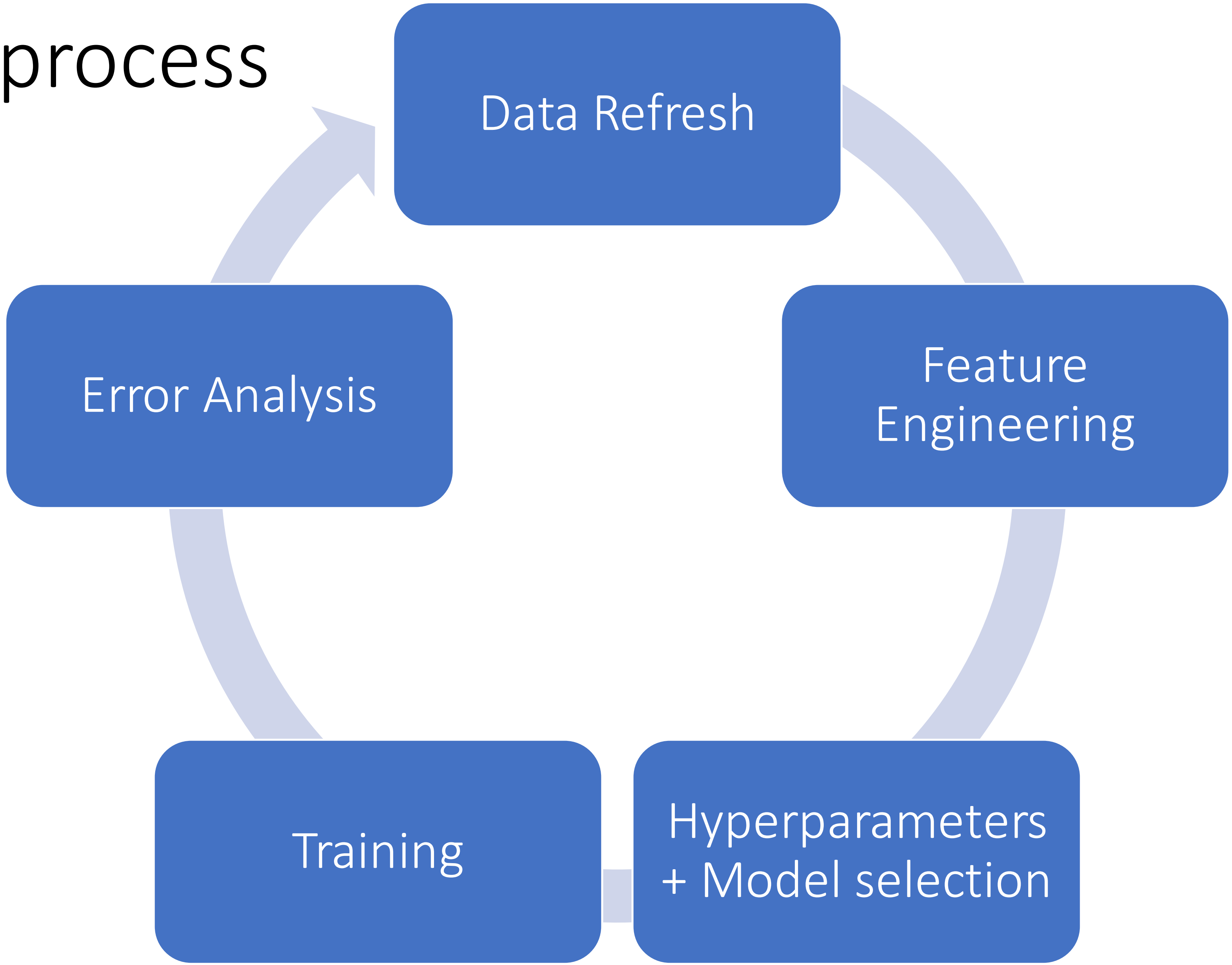


Training Pipeline

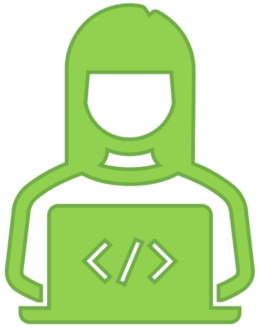


Inference Pipeline

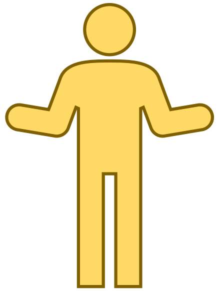
Modeling process



Business Value



Data Scientist: Model is performing very well on test set



Product Owner: However, this isn't flexible/feasible for the application

Value Driven Machine Learning

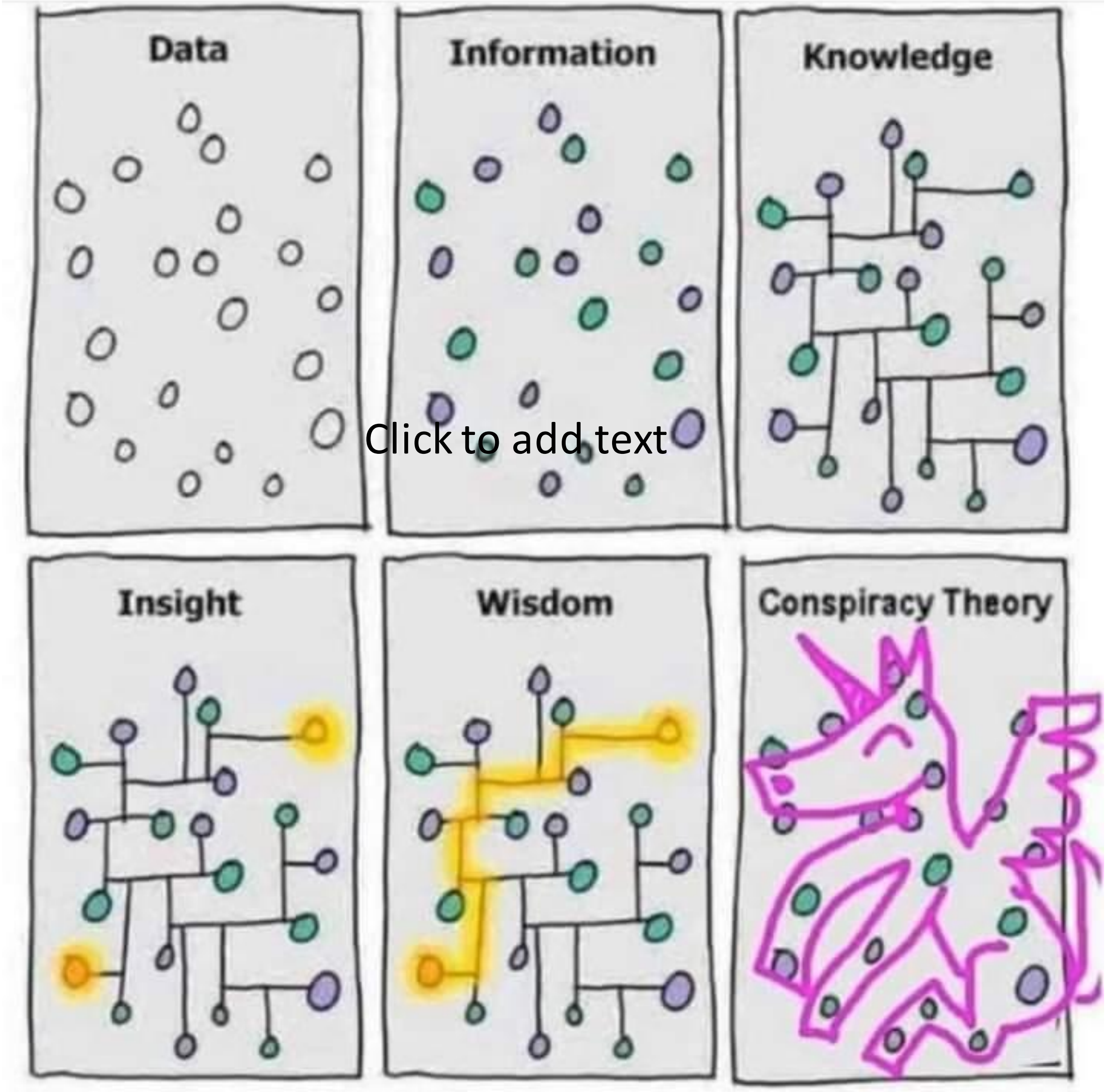
Data Centric
- Open source data
- Quality data over more data

Baseline
- Identify state of art models
- Human level metrics

Model
- Simple deployable model
- Intuitive features

Refresh
- Pipelines
- Inference

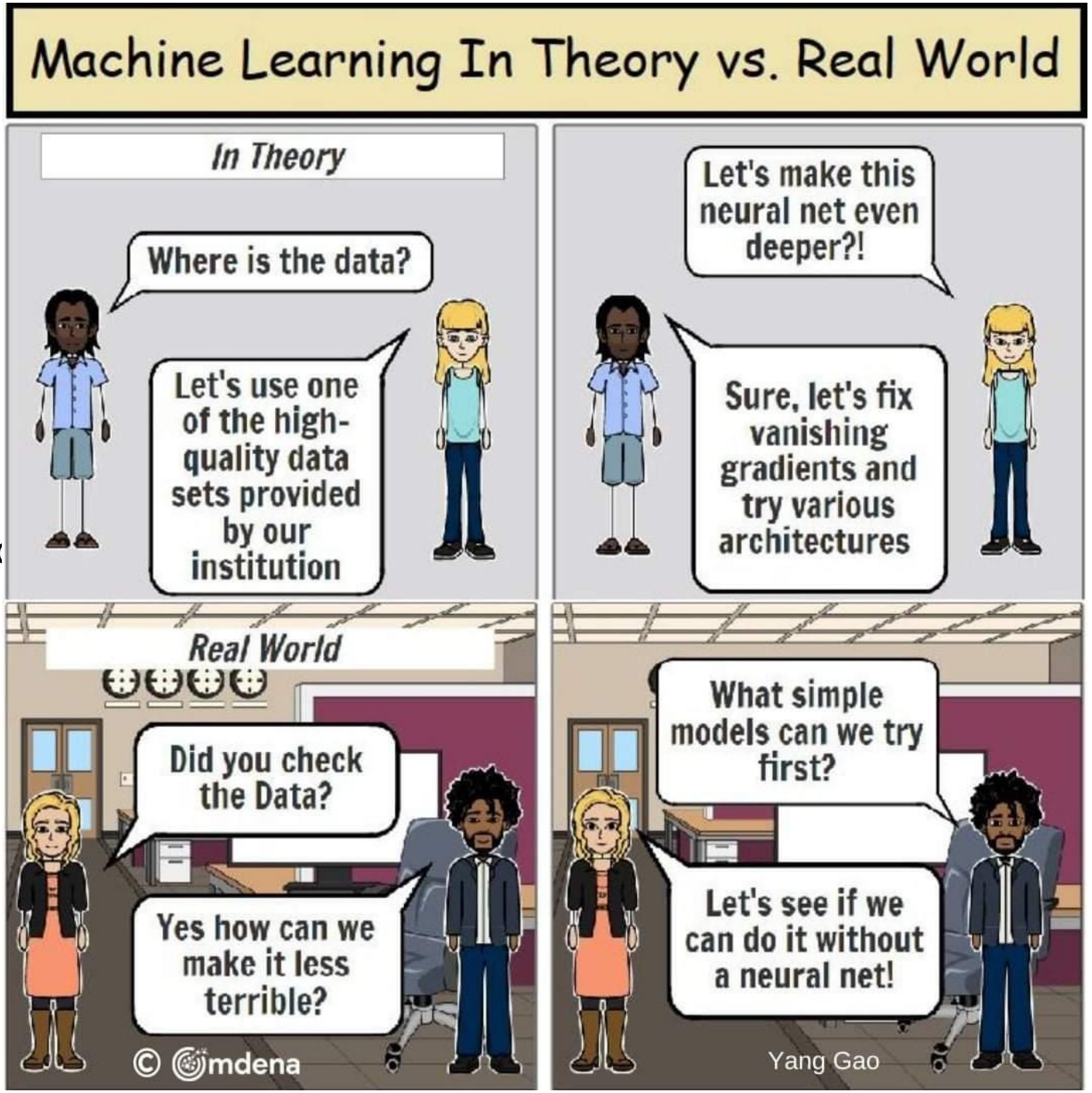
Responsible AI



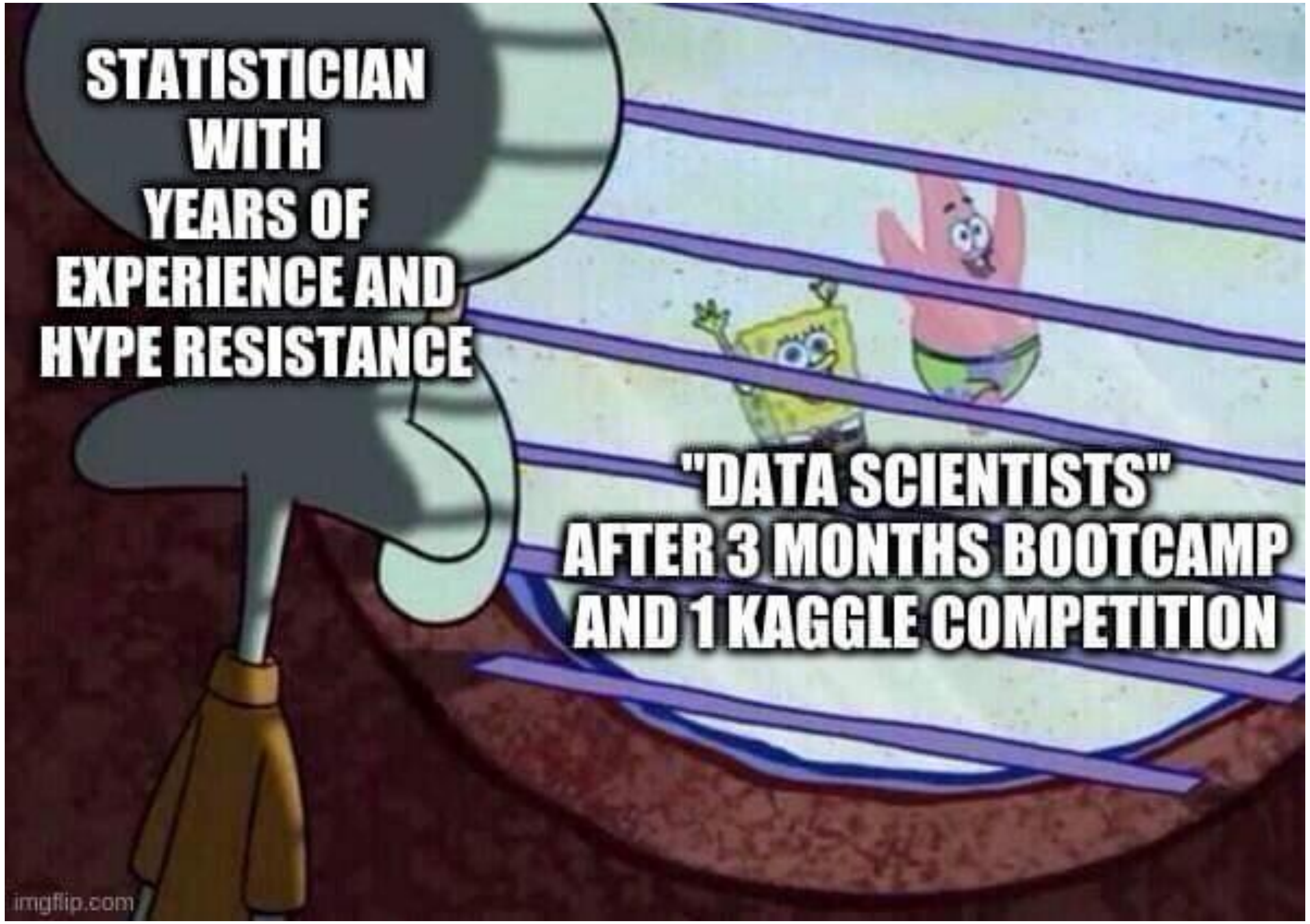
Click to add text

Practical AI

Cli



Practical AI



Section - 8

Career Opportunities

Job Roles

Data Analyst

Data Engineer

Machine Learning Engineer

Data Scientist

Data Architect

Statistician

Data and Analytics Manager

Data Architect

Statistician

Data and Analytics Manager