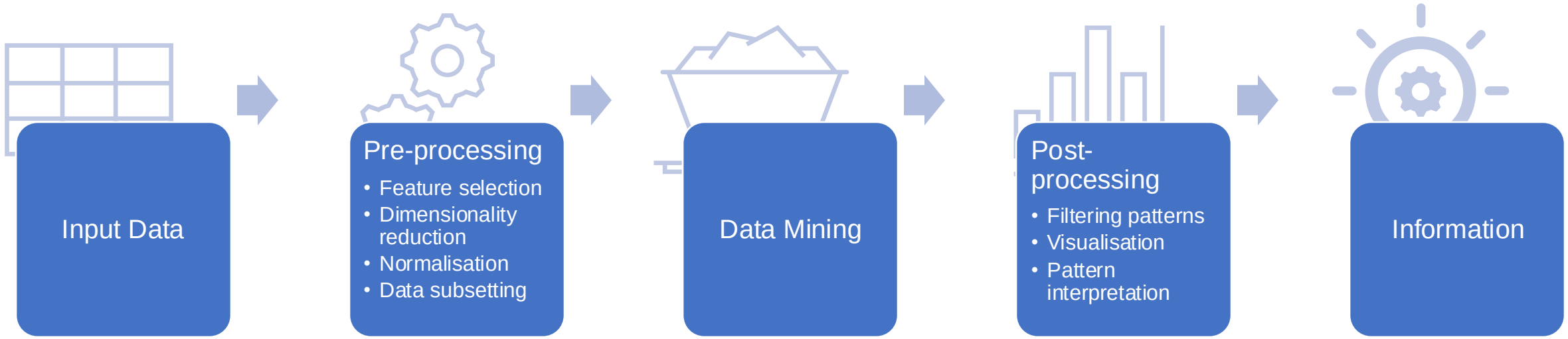
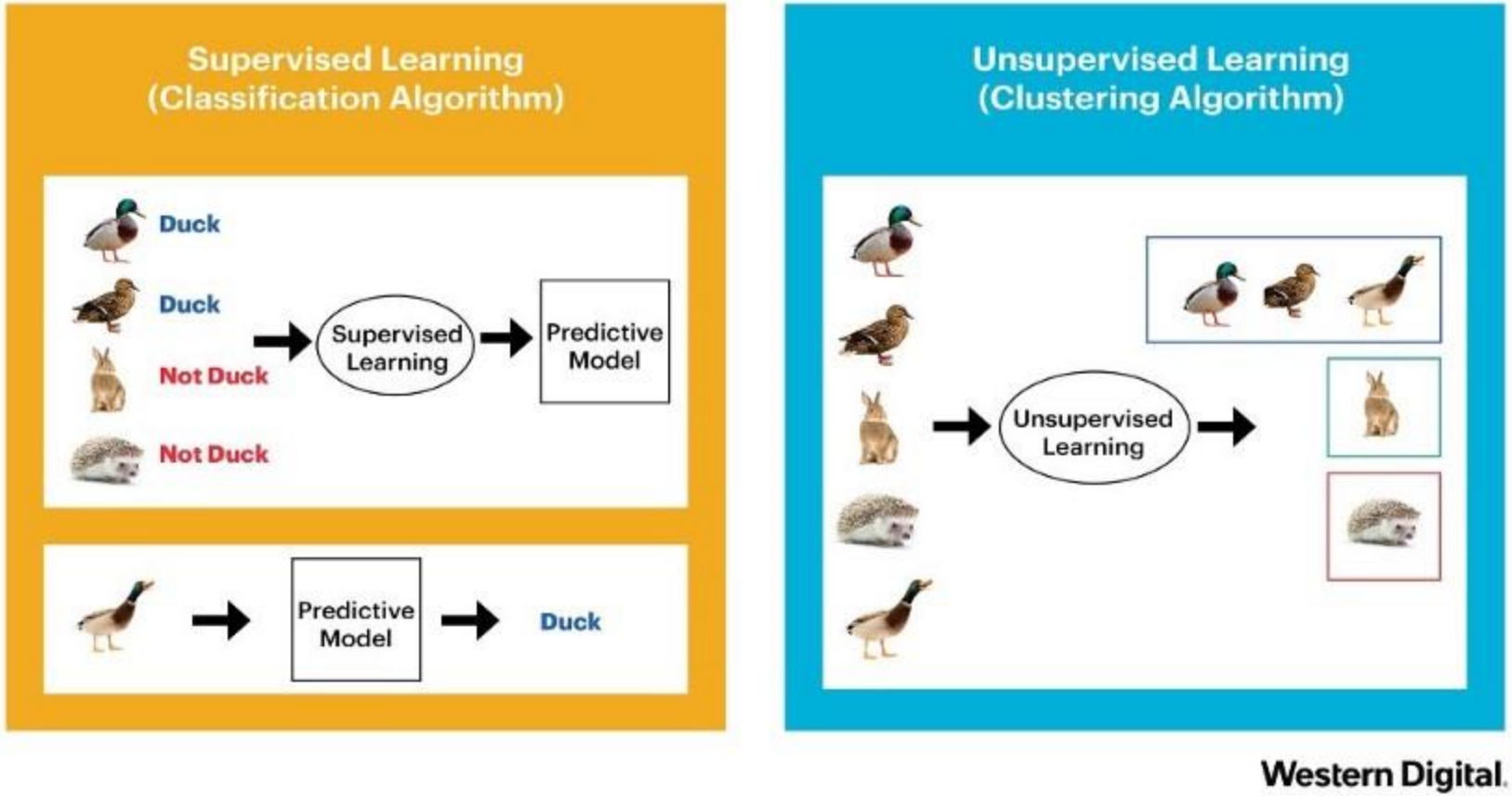
Applications of Python in Data Mining

Indrajith Ekanayake

The Knowledge Discovery process



Supervised vs unsupervised techniques



Data Mining Task Categories

Predictive

Goal: Predict (future) value of one or more attributes

Predict target (or dependent) variable

Using explanatory (independent) variables

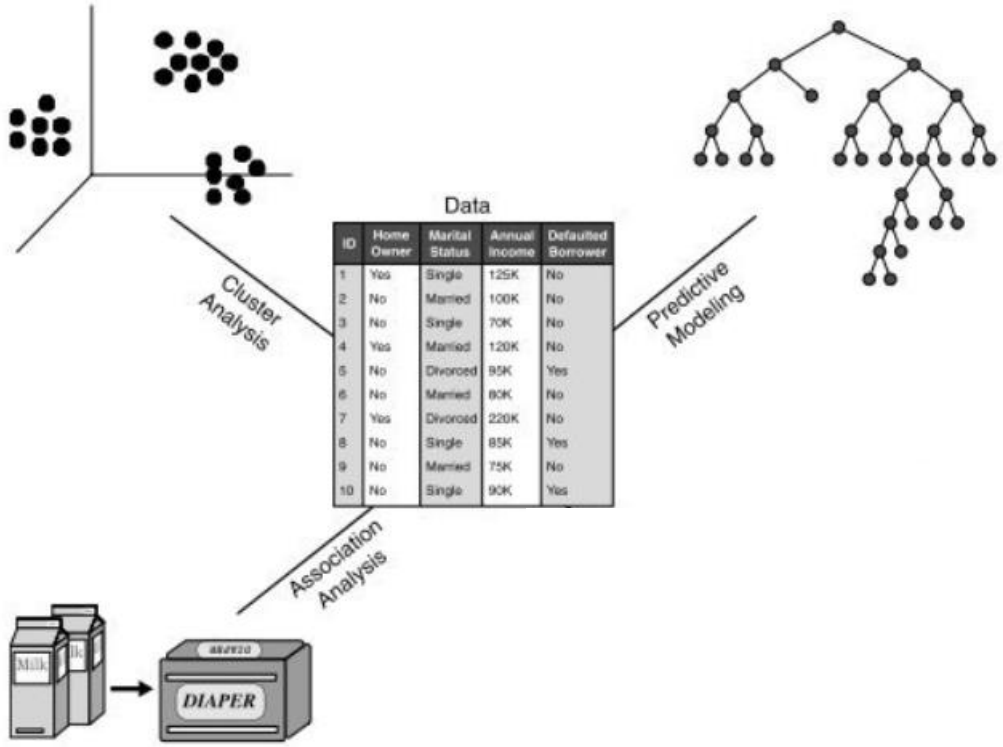
Descriptive

Derive patterns (correlations, trends, clusters, anomalies)

Often exploratory

Frequently requires post-processing to validate and explain results

Typical Data Mining tasks



Tan et al. (2014)

Classification

Predictive modelling

Model to predict target variable as function of explanatory variables

Classification predicts discrete values

Regression predicts continuous values

Applications: Identify customer likely to react to campaign, predict patient’s likelihood to develop disease based on bio markers, identify flower species based on petal and sepal dimensions

Classification

Name	Warm-blooded	Gives Birth	Aquatic Creature	Aerial Creature	Has Legs	Hibernates	Class
human	yes	yes	no	no	yes	no	mammals
python	no	no	no	no	no	yes	reptiles
salmon	no	no	yes	no	no	no	fishes
whale	yes	yes	yes	no	no	no	mammals
frog	no	no	yes	no	yes	yes	amphibians
komodo	no	no	no	no	yes	no	reptiles
bat	yes	yes	no	yes	yes	yes	mammals
pigeon	yes	no	no	yes	yes	no	birds
cat	yes	yes	no	no	yes	no	mammals
leopard shark	no	yes	yes	no	no	no	fishes
turtle	no	no	yes	no	yes	no	reptiles
penguin	yes	no	yes	no	yes	no	birds
porcupine	yes	yes	no	no	yes	yes	**?**
eel	no	no	yes	no	no	no	**?**
salamander	no	no	yes	no	yes	yes	**?**

Predictor variables

Target

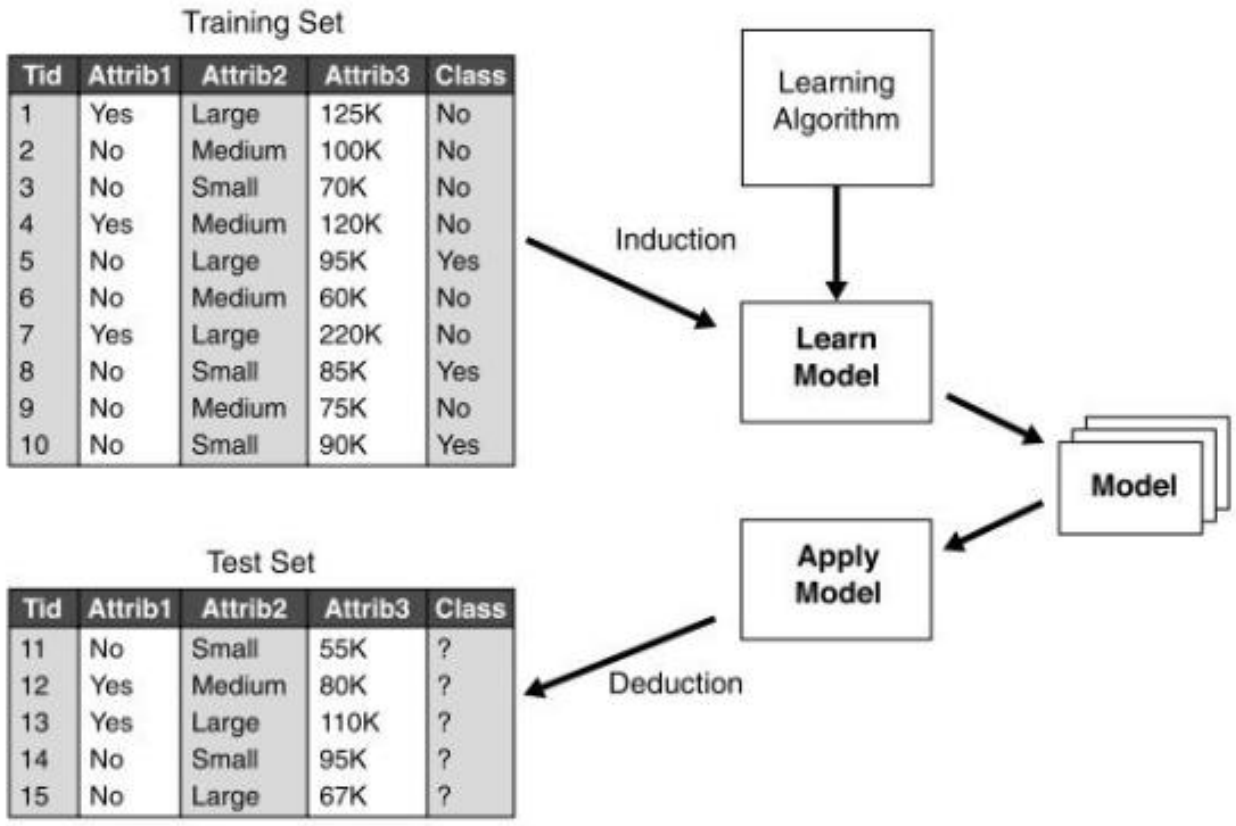
General approach

Definition: Learning a target function *f* that maps each attribute set *x* to one of the predefined class labels *y*.

Train model with a training set

Test model with a test set

Evaluate model



Confusion matrix

$$Accuracy = \frac{Correct\ predictions}{All\ predictions} = \frac{TP + TN}{TP + FP + FN + TN}$$

$$Error\ rate = \frac{Wrong\ predictions}{All\ predictions} = \frac{FP + FN}{TP + FP + FN + TN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

		Predicted	
		yes	**no**
Actual	**yes**	True positive (TP)	False negative (FN)
	no	False positive (FP)	True negative (TN)

An example

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} = \frac{45 + 35}{45 + 12 + 8 + 35} = 80\%$$

$$Error\ rate = \frac{FP + FN}{TP + FP + FN + TN} = \frac{12 + 8}{45 + 12 + 8 + 35} = 20\%$$

$$Precision = \frac{TP}{TP + FP} = \frac{45}{45 + 8} \approx 85\%$$

$$Recall = \frac{TP}{TP + FN} = \frac{45}{45 + 12} \approx 79\%$$

		Predicted	
		yes	**no**
Actual	**yes**	45	12
	no	8	35

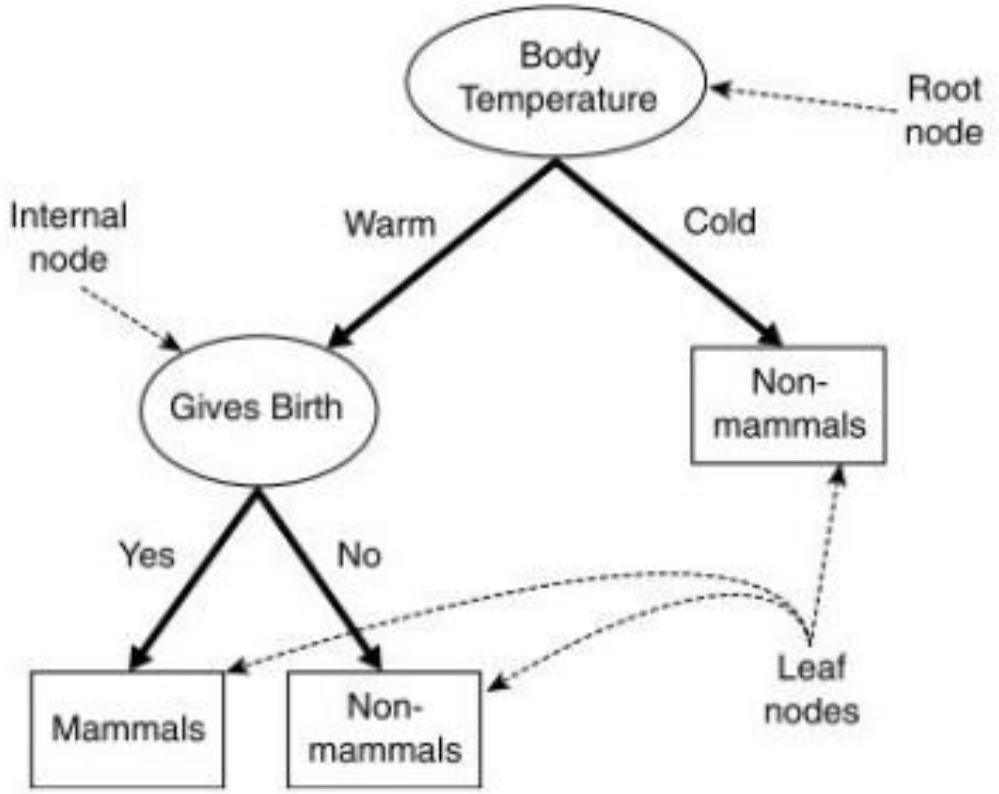
Decision tree

Decision tree is a widely used classification technique

Produces a *white-box model*, i.e. we know why it works the way it works

Arranges attributes as nodes and values as edges

Following the tree leads to the predicted class



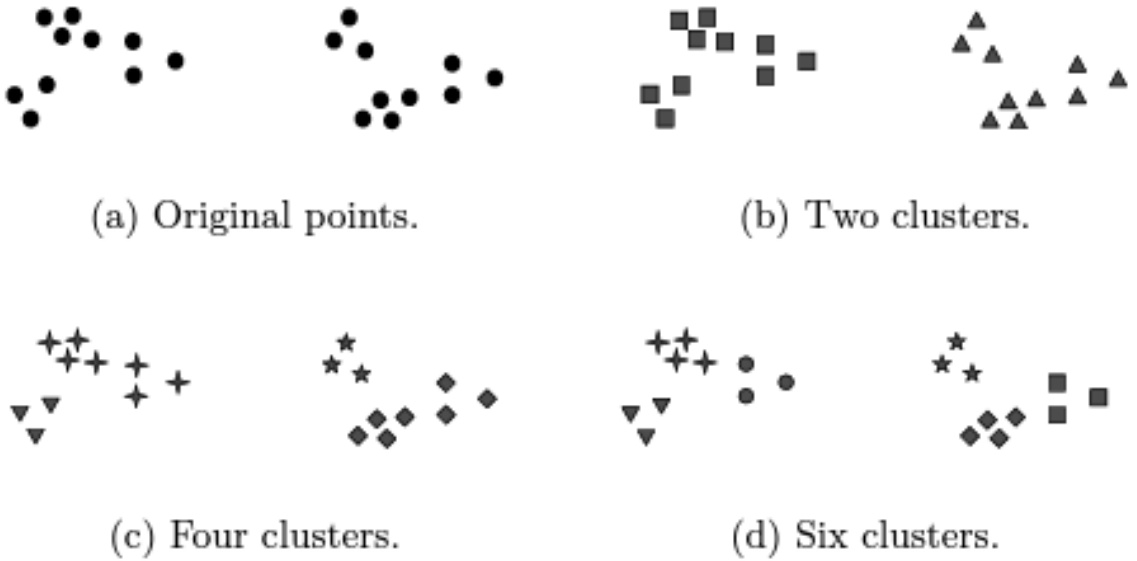
Cluster analysis

What is clustering?

Automatically group observations into similar clusters

Assign previously unknown labels to data

Helps understanding data



(a) Original points. (b) Two clusters.

(c) Four clusters. (d) Six clusters.

Euclidean distance

How do we measure similarity (or distance)?

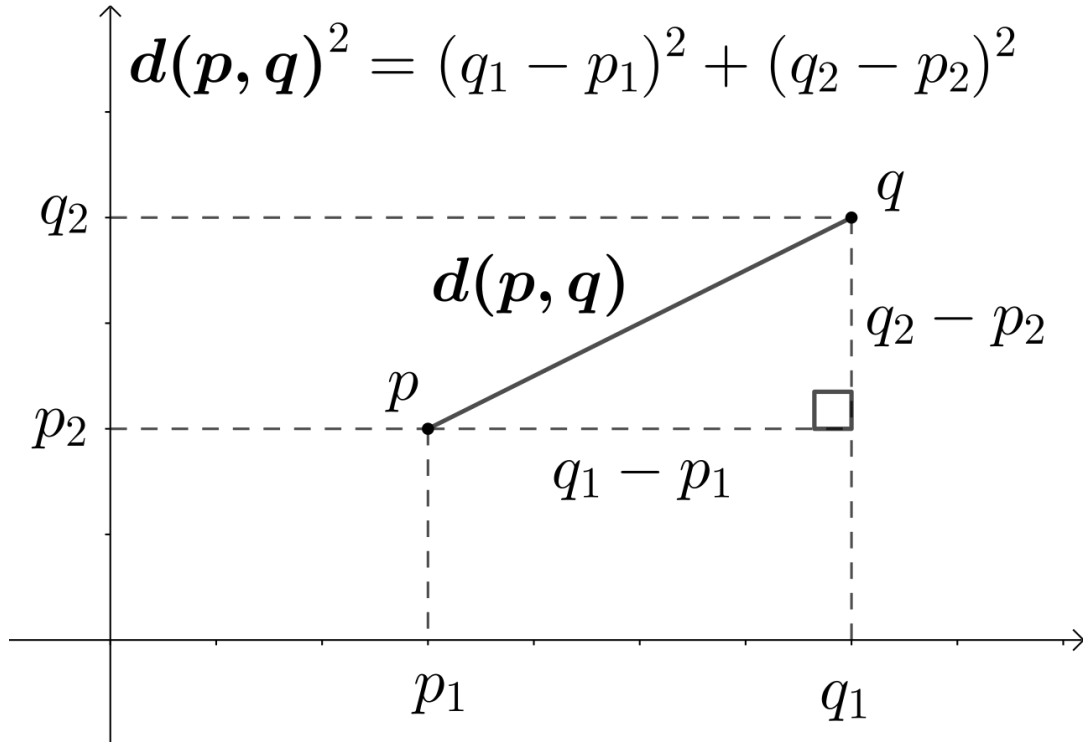
Euclidean distance

Use Pythagorean theorem to calculate distance

Most-used distance measure

All values should be on similar scales/ranges

Susceptible to outliers in a single variable



k-means

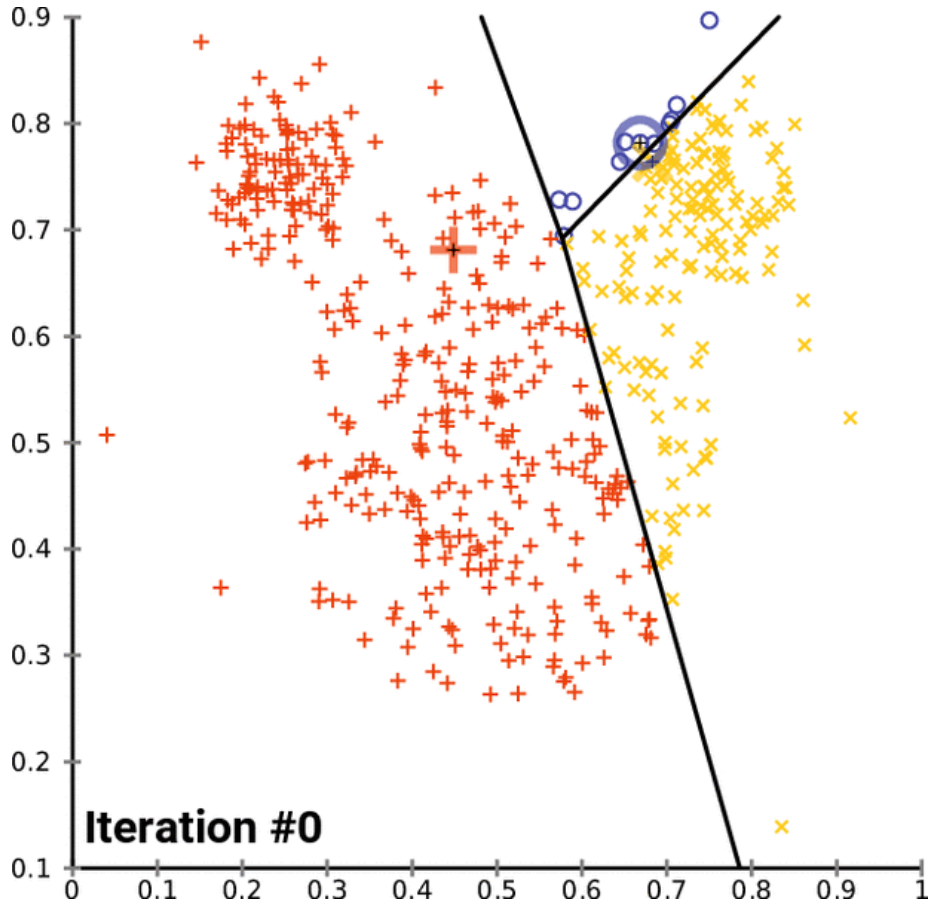
Need to specify number of clusters *n* first

k-means places *n* centres randomly, then alternates between two steps

1. Assign each observation to the cluster with the nearest centre
2. Move centre to middle of cluster

The algorithm completes when the clusters don't change anymore

Several refinements of the standard algorithm exist



Cluster quality and ideal number of cluster *n*

Cluster quality can be measures as Squared Sum of Errors (SSE) or inertia

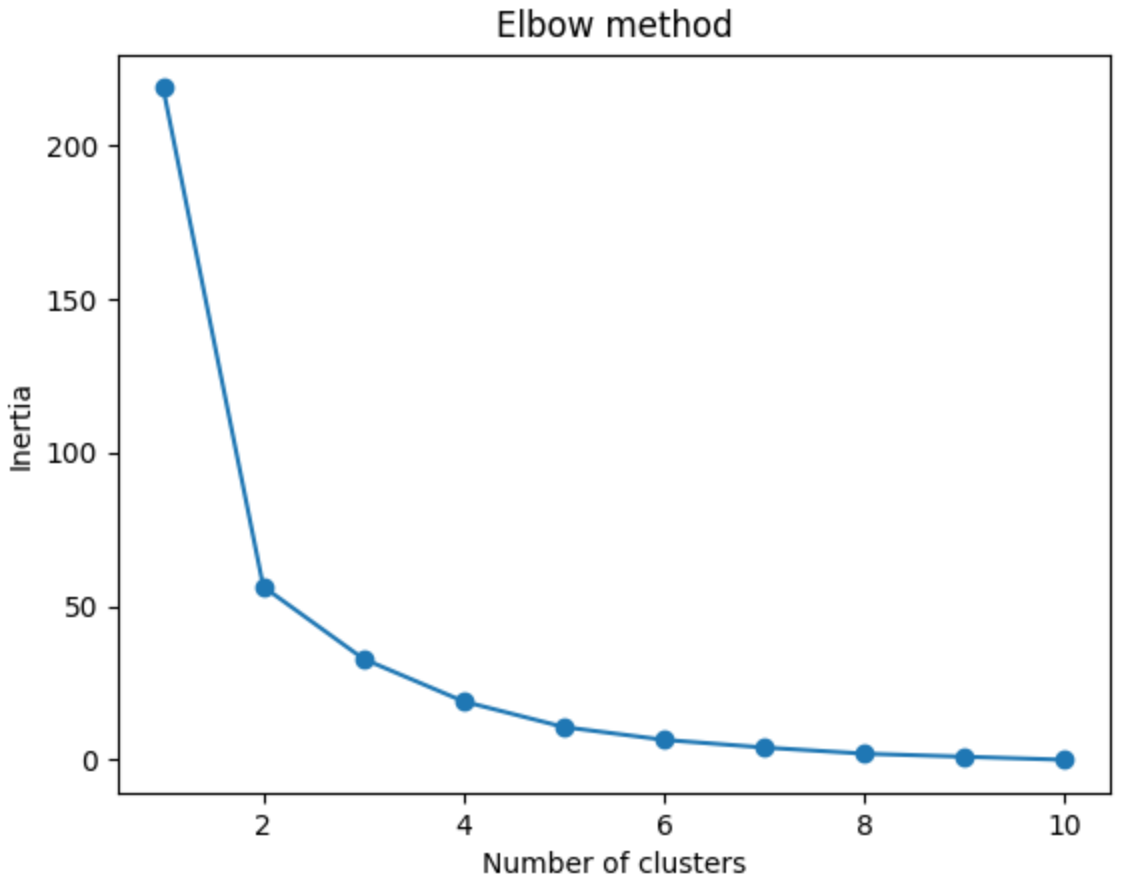
Sum of the distance between each observation and its centroid

Smaller SSE is better

Higher number of cluster *n* always results in smaller SSE

When plotting the SSE per clusters *n*, there is usually a point where increasing *n* doesn't significantly decrease the SSE

Indicates ideal number of clusters



Association rules

Itemset and Support Count

Each transaction contains a subset of items of zero or more items (an itemset)

E.g. $\{Beer, Nappies, Milk\}$

Support count is the number of transaction containing an itemset

Expressed as $\sigma(X)$

Support count of $\{Beer, Nappies, Milk\}$ is 2

ID	Items
1	{Bread, Milk}
2	{Bread, Nappies, Beer, Eggs}
3	{Milk, Nappies, Beer, Cola}
4	{Bread, Milk, Nappies, Beer}
5	{Bread, Milk, Nappies, Cola}

Association rule

An association rules is an implication expression $X \rightarrow Y$, where *X* and *Y* are disjoint itemsets

The strength of a rule can be measured in terms of **support** and **confidence**

Support: How often is a rule applicable?

Consider $\{Milk, Nappies\} \rightarrow \{Beer\}$.

Support count for itemset $\{Beer, Nappies, Milk\}$ is 2, total numbers of record is 5

Ergo support is $\frac{2}{5} = 0.4$

Confidence: Support count of $\{Beer, Nappies, Milk\}$ divided by support count of $\{Beer, Nappies\}$

Ergo confidence is $\frac{2}{3} \approx 0.67$

ID	Items
1	{Bread, Milk}
2	{Bread, Nappies, Beer, Eggs}
3	{Milk, Nappies, Beer, Cola}
4	{Bread, Milk, Nappies, Beer}
5	{Bread, Milk, Nappies, Cola}

Why use Support and Confidence?

Low support means rule may occur only by chance

Low support rules are often uninteresting for decision makers

Support $s(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{N}$

Confidence $c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}$

Confidence measures reliability

High confidence means a transaction it is likely to contain *Y*

Like correlation, confidence does **NOT** prove causation!

Code

Support my ongoing research

by filling this questionnaire

Indrajith@ieee.org