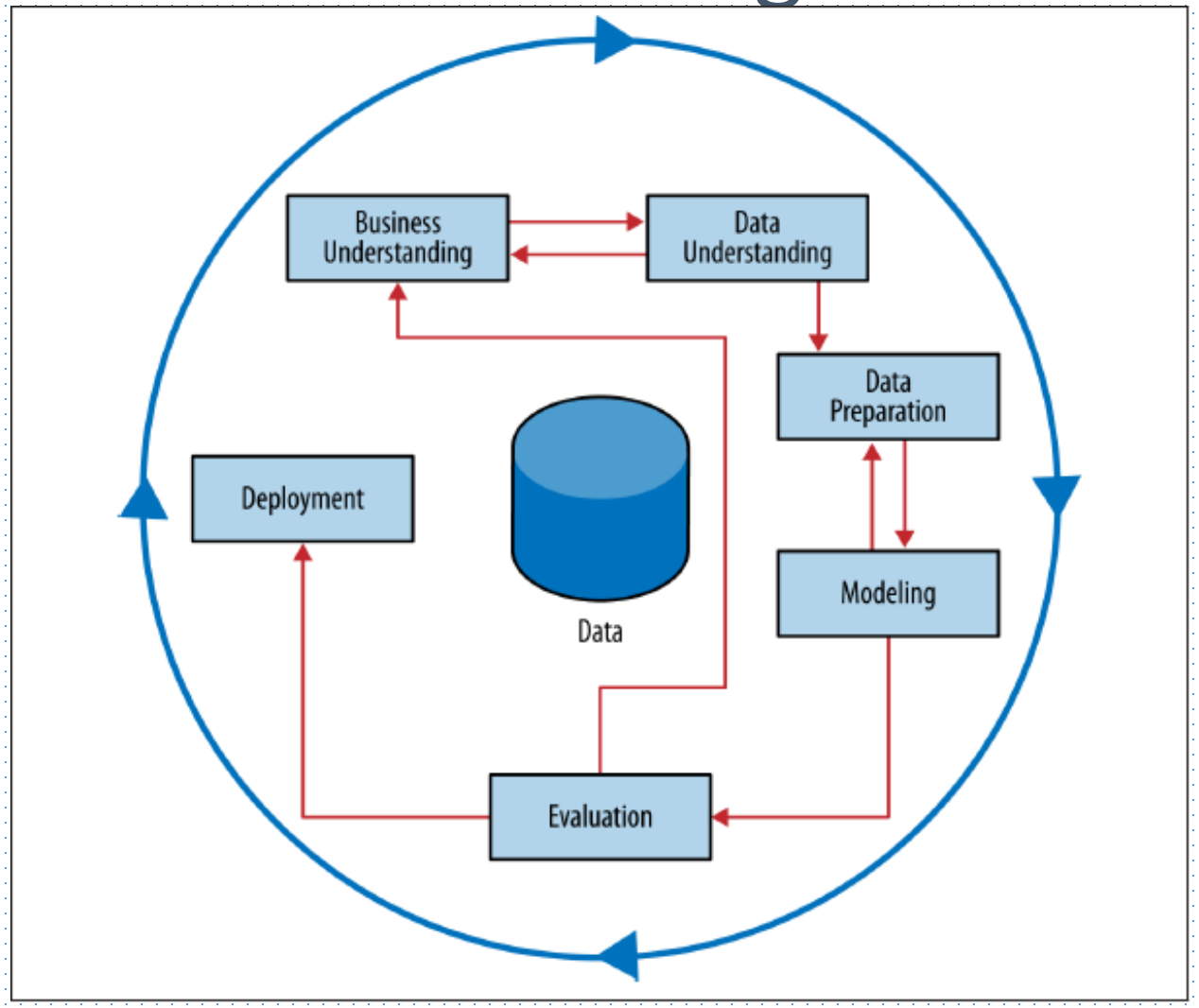
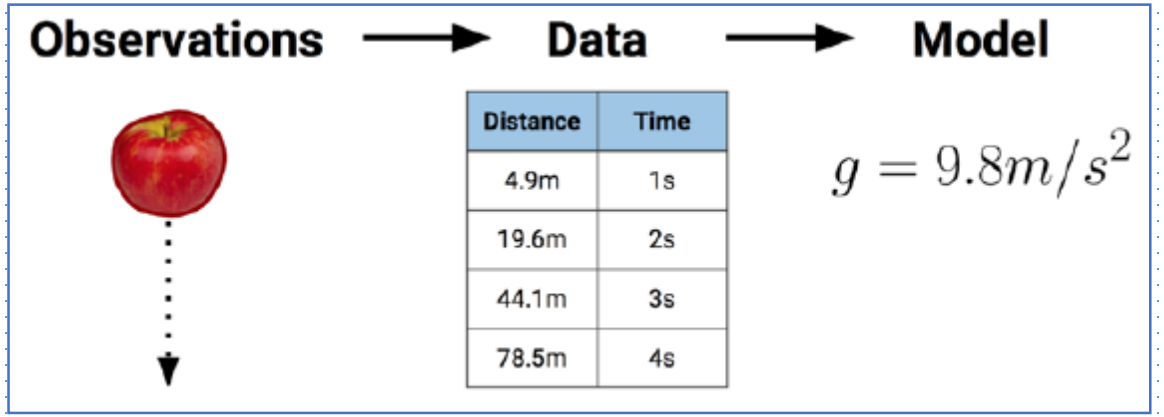
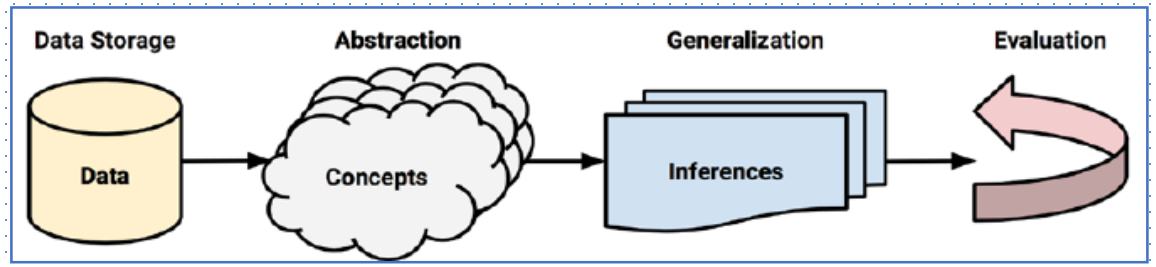
Data Science – Images

Machine Learning Intro







- **Supervised Vs Unsupervised** – Supervisor / Label / Target/ Predicted variable / Dependent variable

- **Prediction Vs Description**
- **White box Vs Black box**

- **Discrete results Vs Continuous results**
- **Classification Vs Regression**

Machine Learning Intro

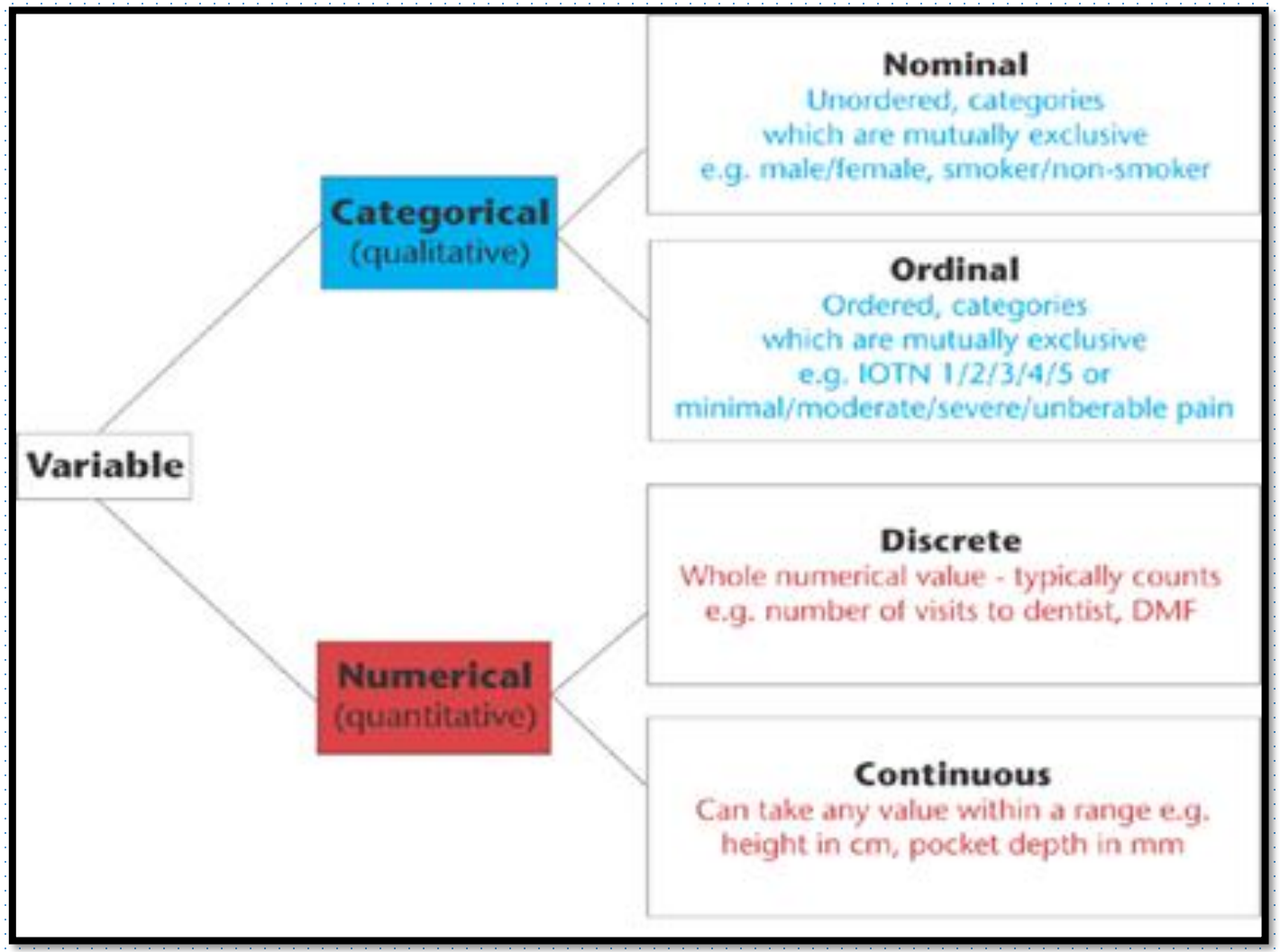
Model	Learning task
Supervised Learning Algorithms	
Nearest Neighbor	Classification
Naive Bayes	Classification
Decision Trees	Classification
Classification Rule Learners	Classification
Linear Regression	Numeric prediction
Regression Trees	Numeric prediction
Model Trees	Numeric prediction
Neural Networks	Dual use
Support Vector Machines	Dual use
Unsupervised Learning Algorithms	
Association Rules	Pattern detection
k-means clustering	Clustering
Meta-Learning Algorithms	
Bagging	Dual use
Boosting	Dual use
Random Forests	Dual use

A class of machine learning algorithms known as **meta-learners** is not tied to a specific learning task, but is rather focused on learning how to learn more effectively.

A meta-learning algorithm uses the result of some learnings to inform additional learning.

Semi-Supervised Learning - the algorithm is trained upon a combination of labeled and unlabeled data.

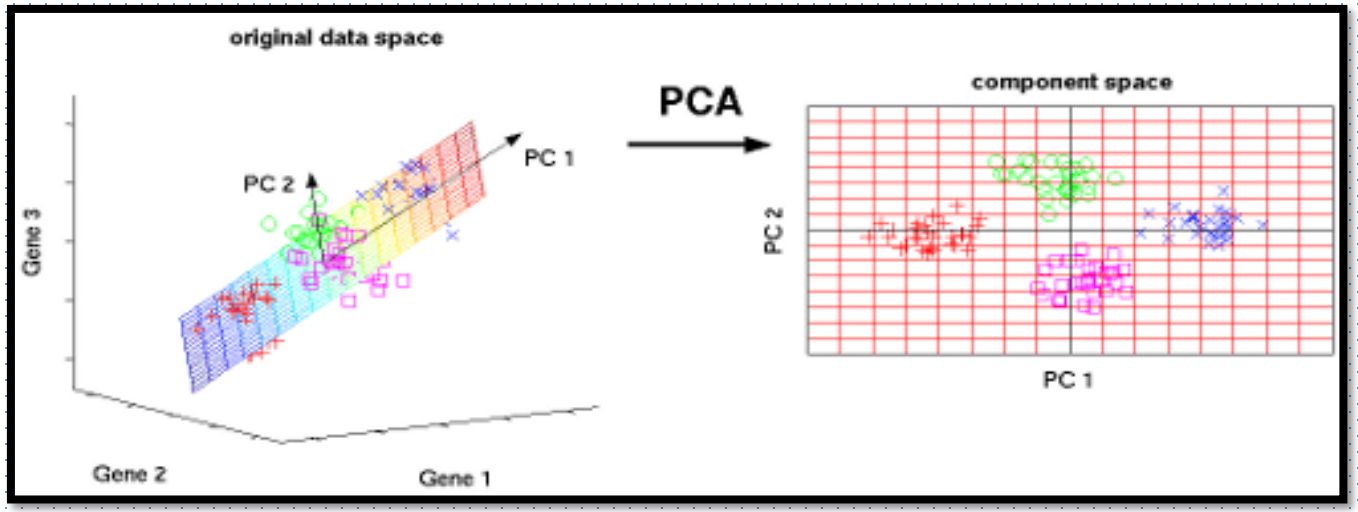
Examples : Speech Analysis, Internet Content Classification:



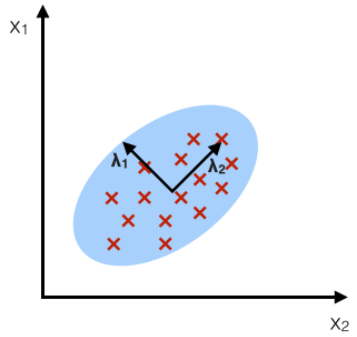
Dummy encoding

Original data:		One-hot encoding format:				
id	Color	id	White	Red	Black	Purple	Gold
1	White	1	1	0	0	0	0
2	Red	2	0	1	0	0	0
3	Black	3	0	0	1	0	0
4	Purple	4	0	0	0	1	0
5	Gold	5	0	0	0	0	1

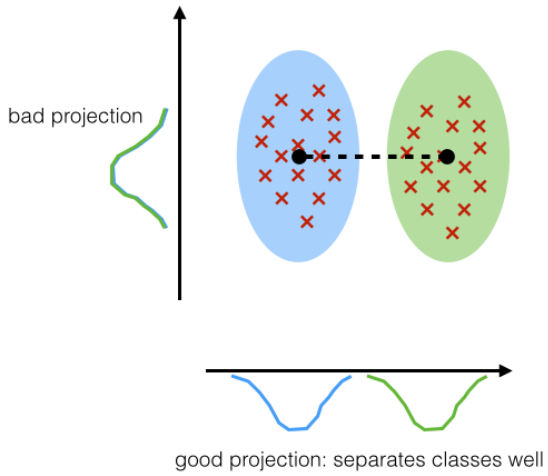
PCA and LDA

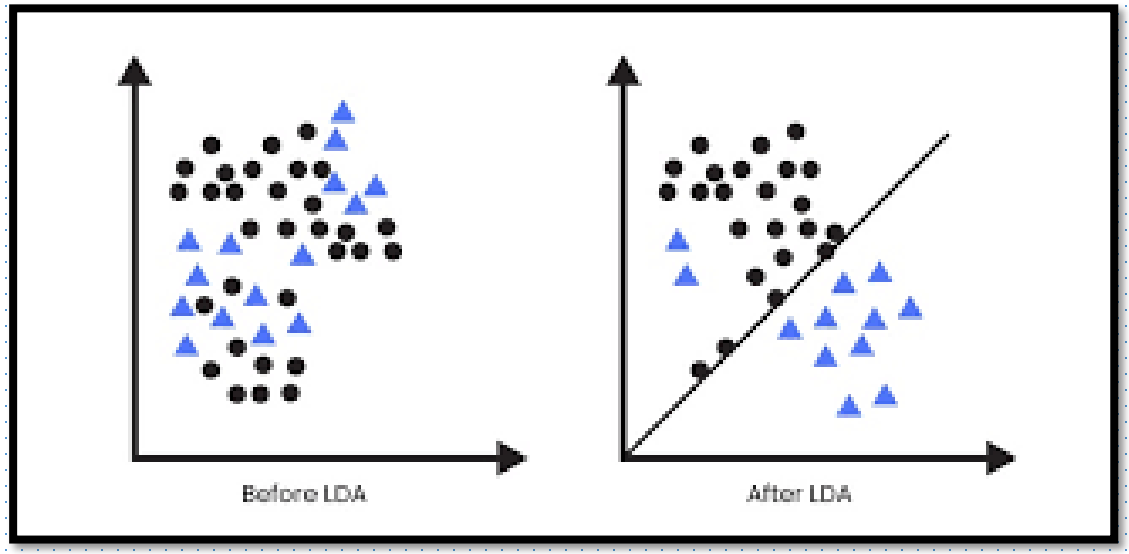


PCA:
component axes that maximize the variance



LDA:
maximizing the component axes for class-separation





K-Means Clustering

Problem to Solve :

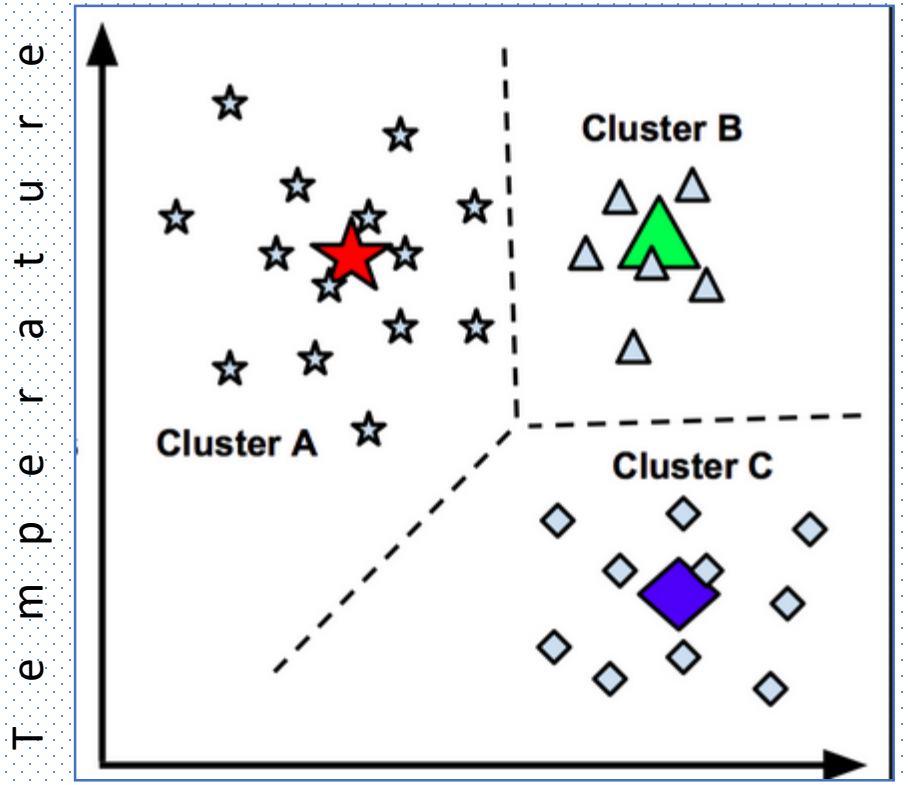
Target different types of clothing to different groups of towns :

1. Temperature
2. Humidity

Recursive process

1. Centroids are chosen at random
2. Distance is computed between each node and centroids
3. Node is assigned to nearest centroid
4. Calculate new centroids
5. REPEAT

$$\text{dist}(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$



Linear Regression

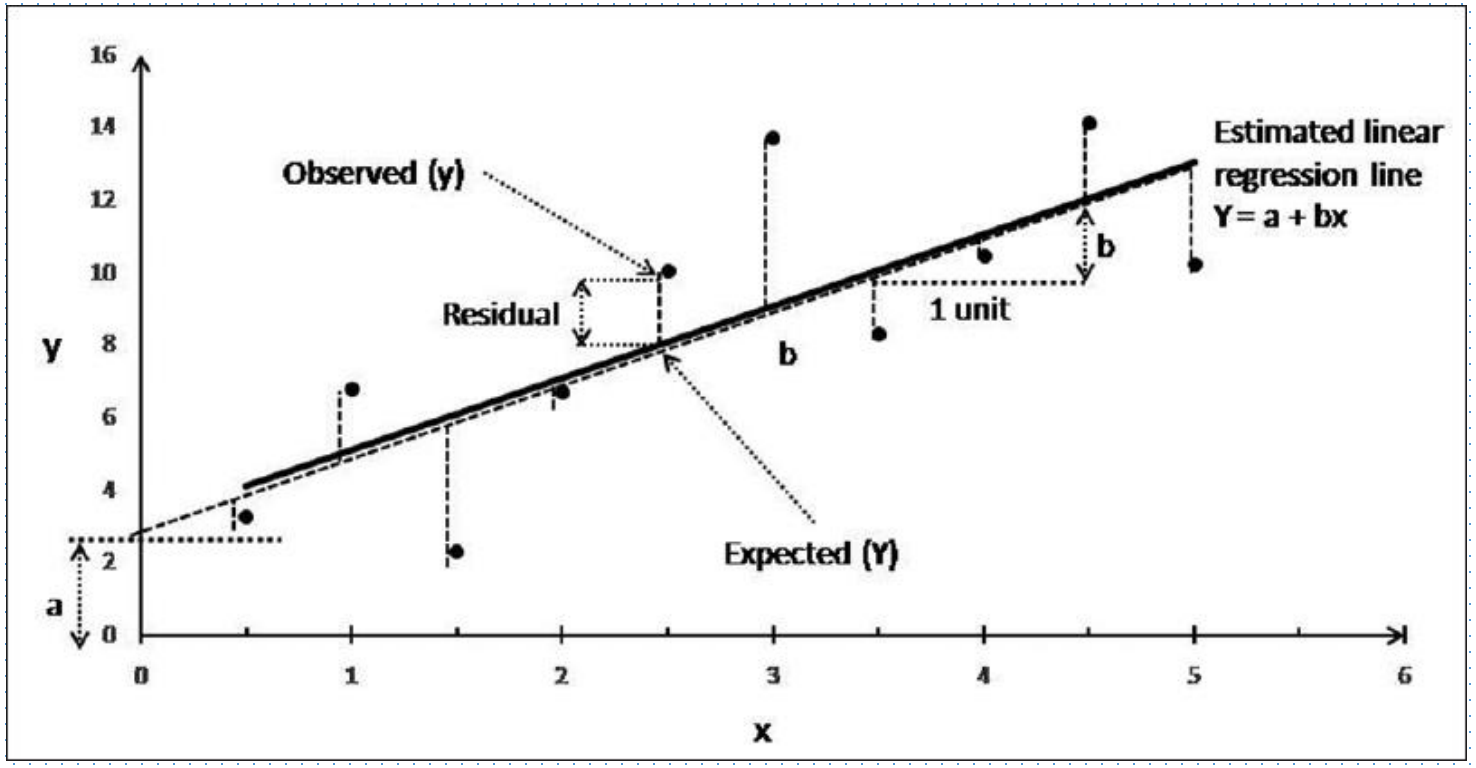
Problem to Solve :

Target different brands of clothing (with different price bands) to different towns based on cost of living index :

Important points

1. Y – dependent variable
2. X – independent variable
3. a – value of y when x is 0 - "**Intercept"**
4. b – slope, increase in value of y per 1 unit increase in x - "**Coefficient"**
5. Regression line to fit Least sum of squares
6. e – residual error

Price band

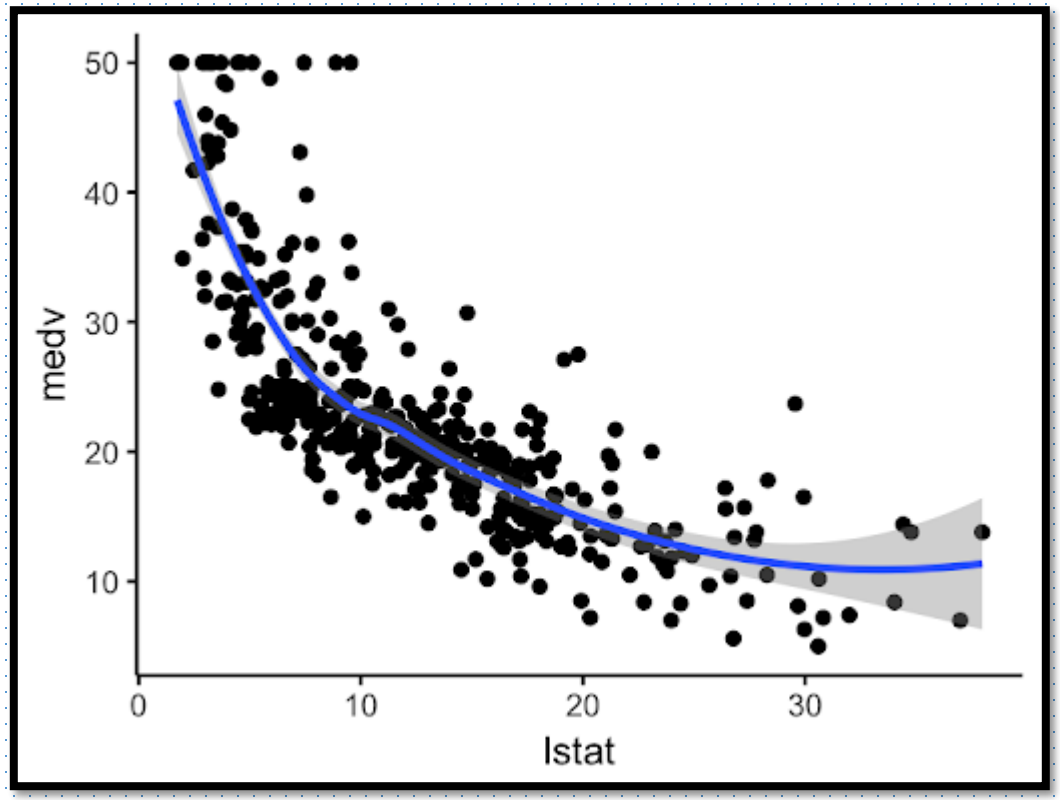


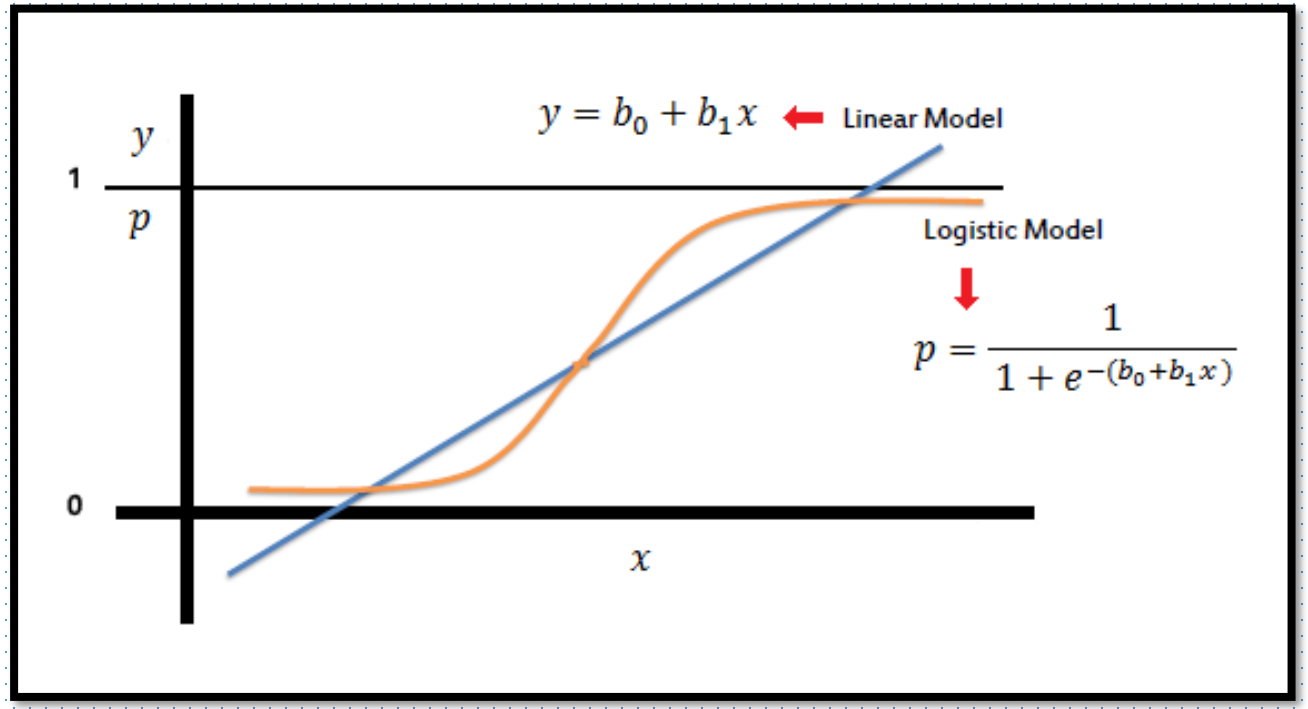
Cost of living index

Multiple Linear Regression with intercept, coefficient and residual error

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_i x_i + \varepsilon$$

Non-Linear regression



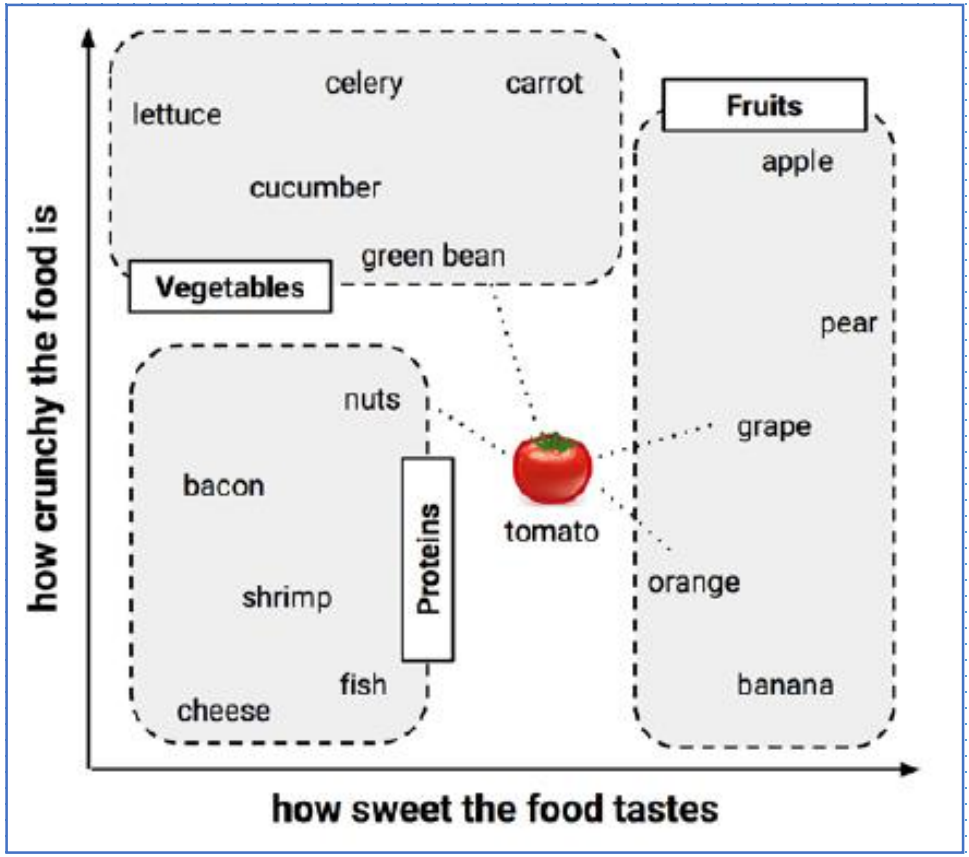


Logistic regression

Classification using nearest Neighbors

Basic concepts

- Suppose that prior to eating the mystery meal we had created a dataset in which we recorded our impressions of a number of ingredients we tasted previously. To keep things simple, we rated only two features of each ingredient.
- The k-NN algorithm treats the features as coordinates in a multidimensional feature space.



Euclidean distance is specified by the following formula, where p and q are the examples to be compared, each having n features. The term p_1 refers to the value of the first feature of example p, while q_1 refers to the value of the first feature of example q:

$$\text{dist}(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ldots + (p_n - q_n)^2}$$

The distance formula involves comparing the values of each feature. For example, to calculate the distance between the tomato (*sweetness* = 6, *crunchiness* = 4), and the green bean (*sweetness* = 3, *crunchiness* = 7), we can use the formula as follows:

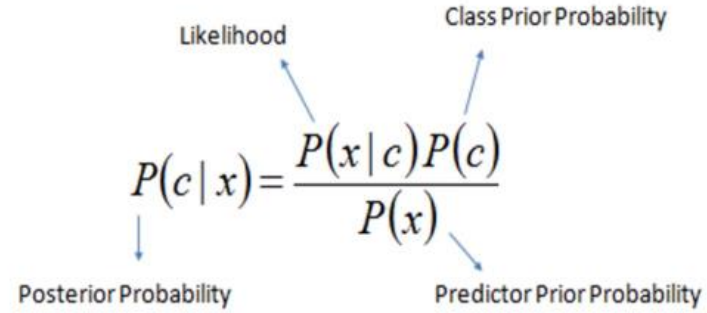
$$\text{dist}(\text{tomato, green bean}) = \sqrt{(6 - 3)^2 + (4 - 7)^2} = 4.2$$

Classification using Naïve Bayes

Text	Tag
“A great game”	Sports
“The election was over”	Not sports
“Very clean match”	Sports
“A clean but forgettable game”	Sports
“It was a close election”	Not sports

Now, which tag does the sentence “A very close game” belong to?

Naive Bayes Classifier



$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \cdots \times P(x_n|c) \times P(c)$$

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

$$P(game|Sports) = \frac{2}{11}$$

$$P(a\,very\,close\,game|Sports) \times P(Sports)$$

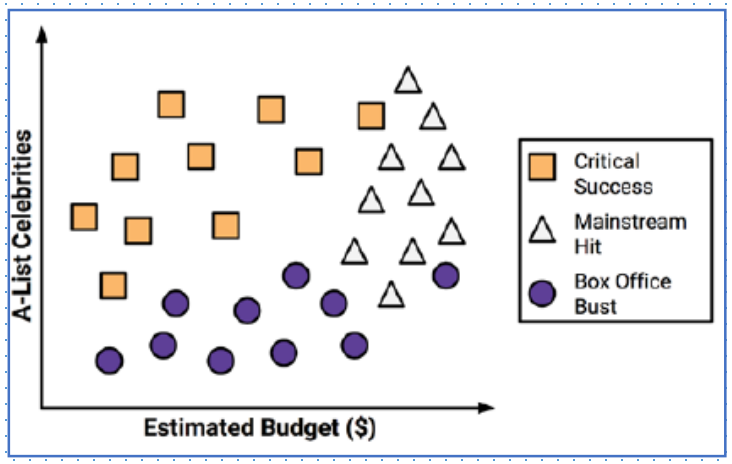
with

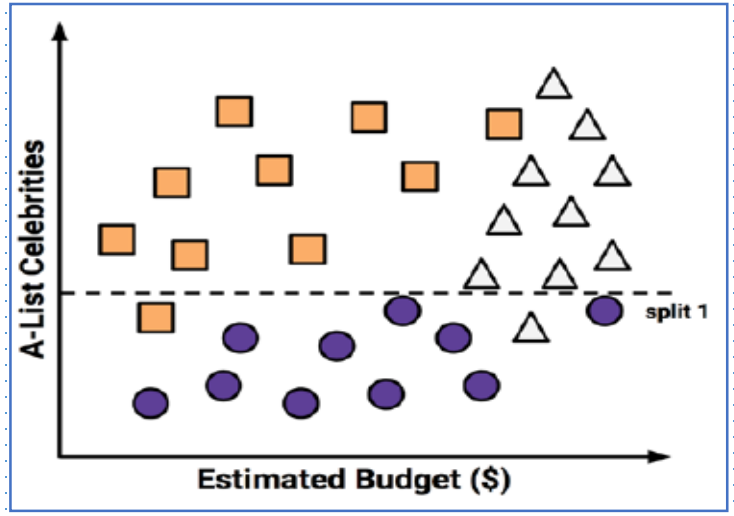
$$P(a\,very\,close\,game|Not\,Sports) \times P(Not\,Sports)$$

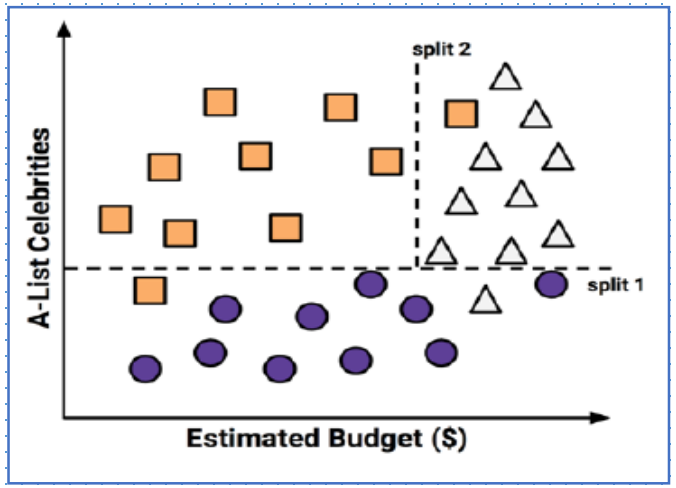
$$P(a\,very\,close\,game) = P(a) \times P(very) \times P(close) \times P(game)$$

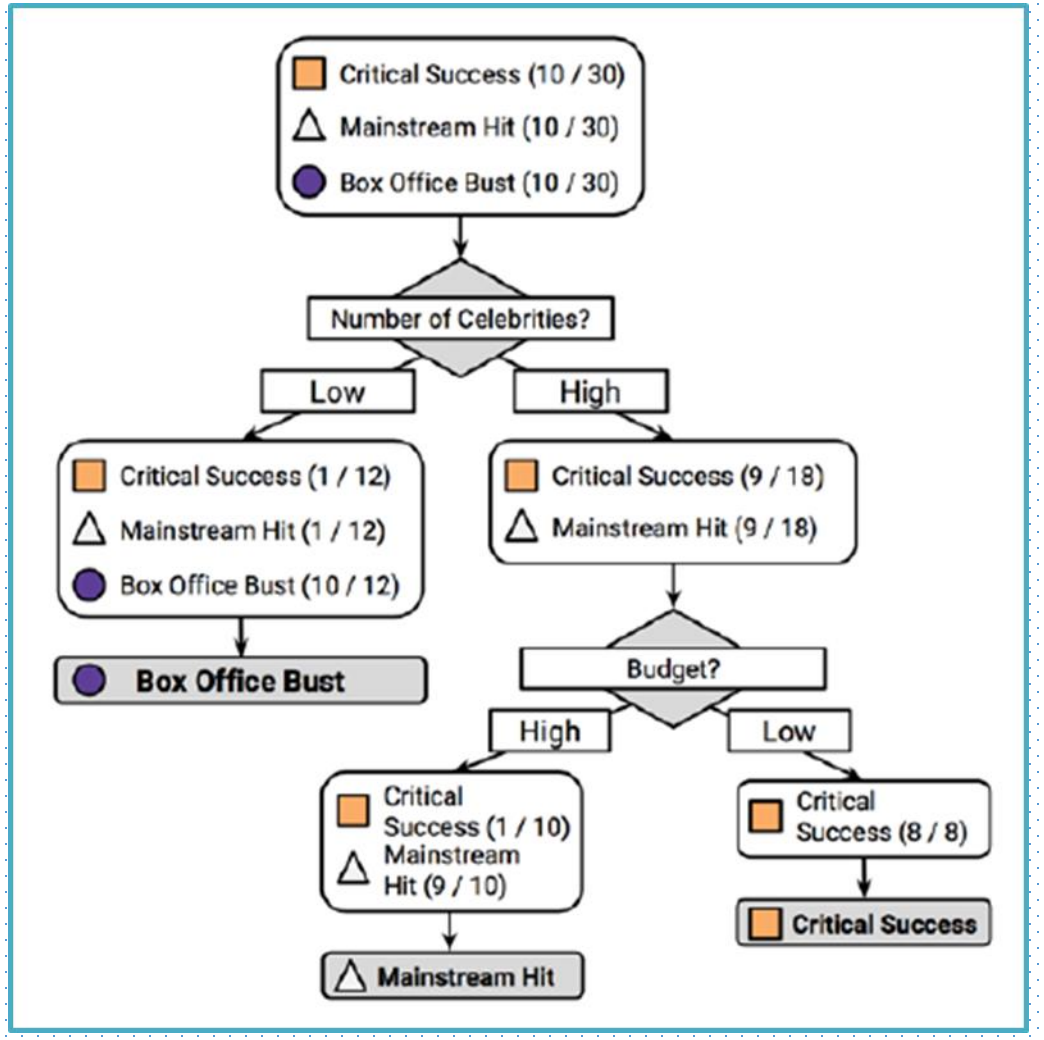
$$P(a\,very\,close\,game|Sports) = P(a|Sports) \times P(very|Sports) \times P(close|Sports) \times P(game|Sports)$$

Classification using Decision trees

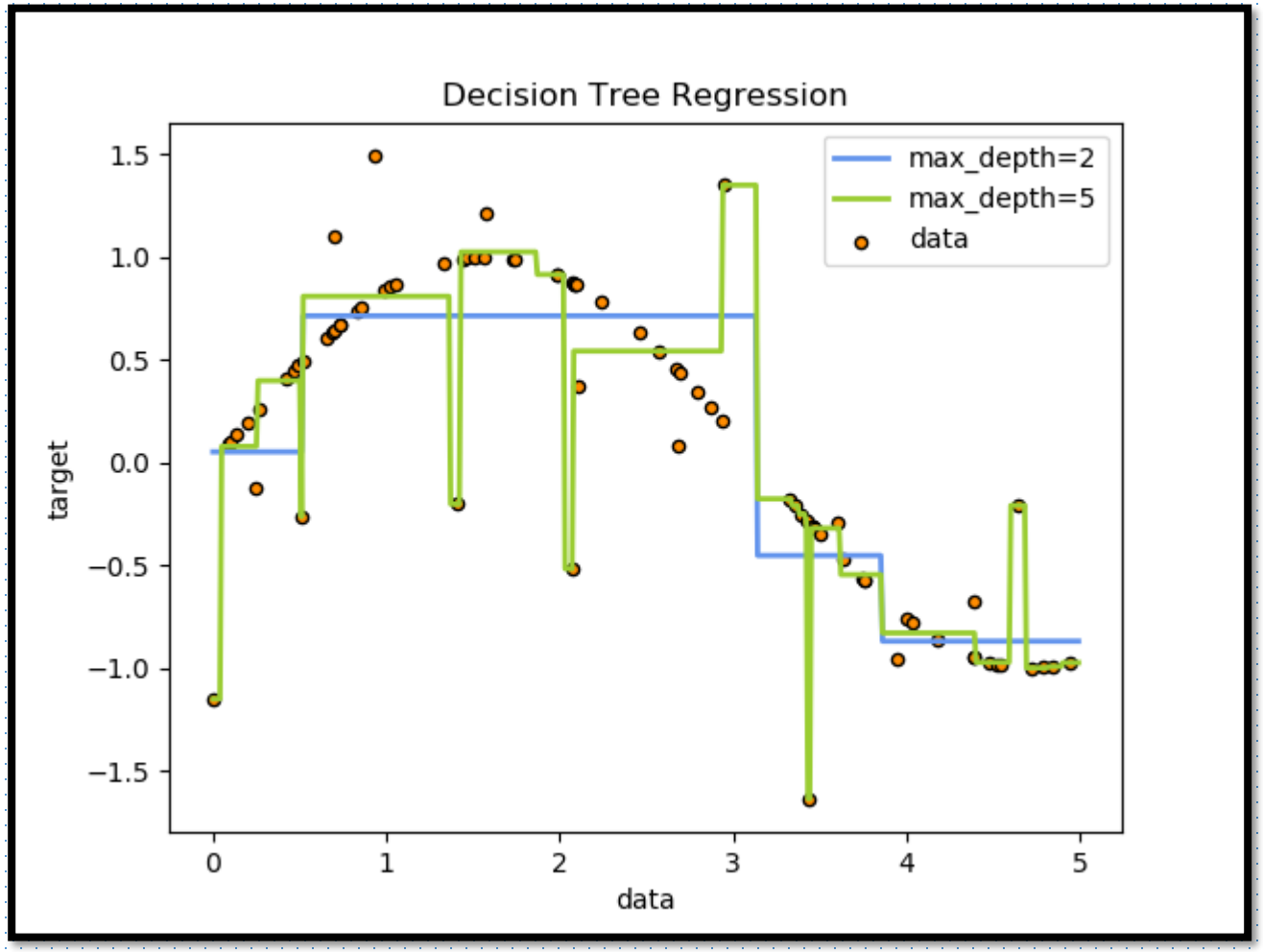


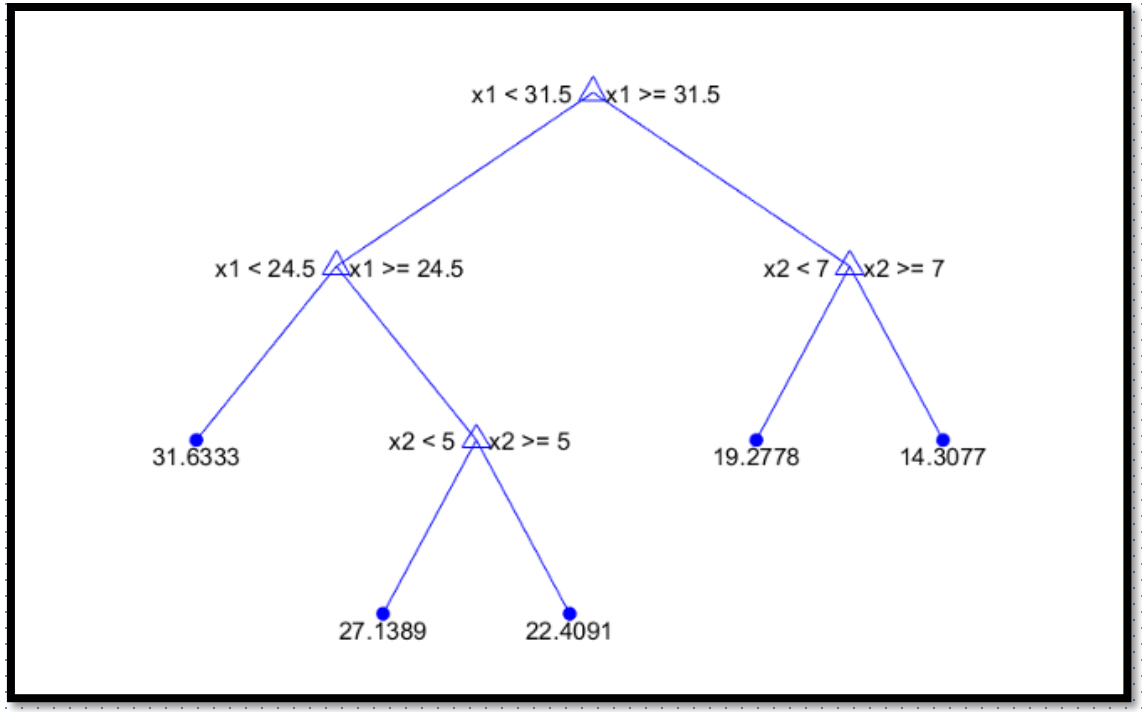




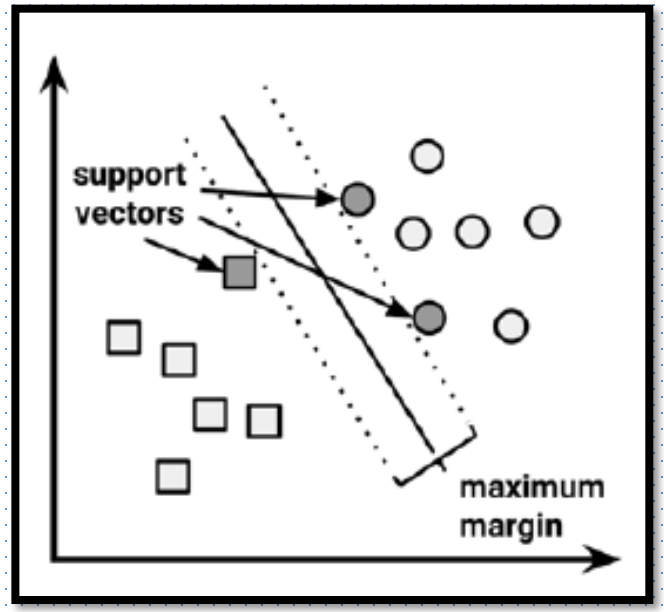


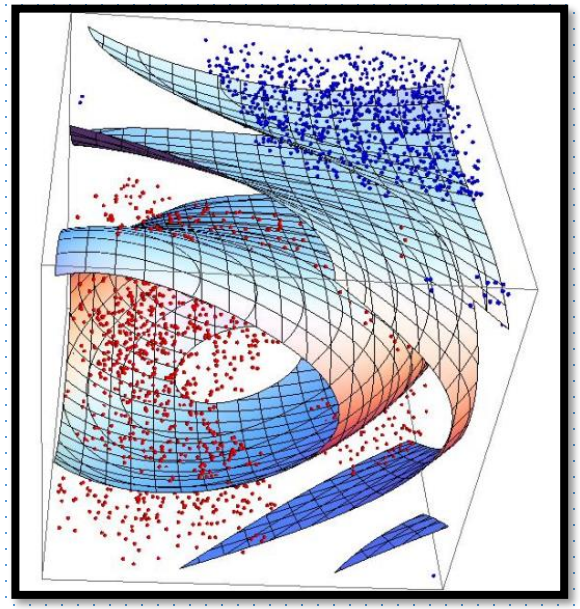
Regression trees

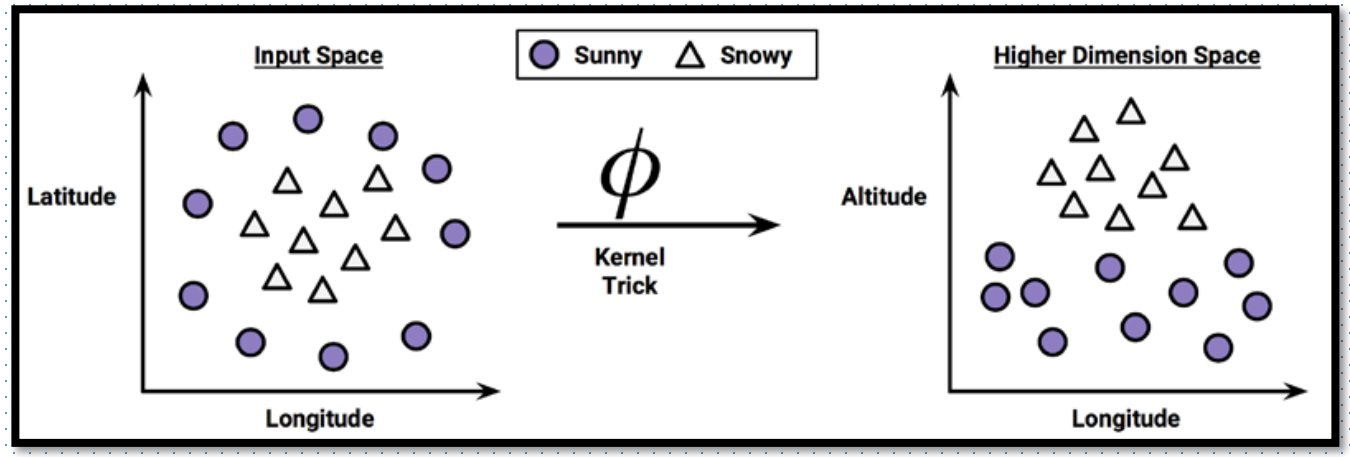




Support vector machines

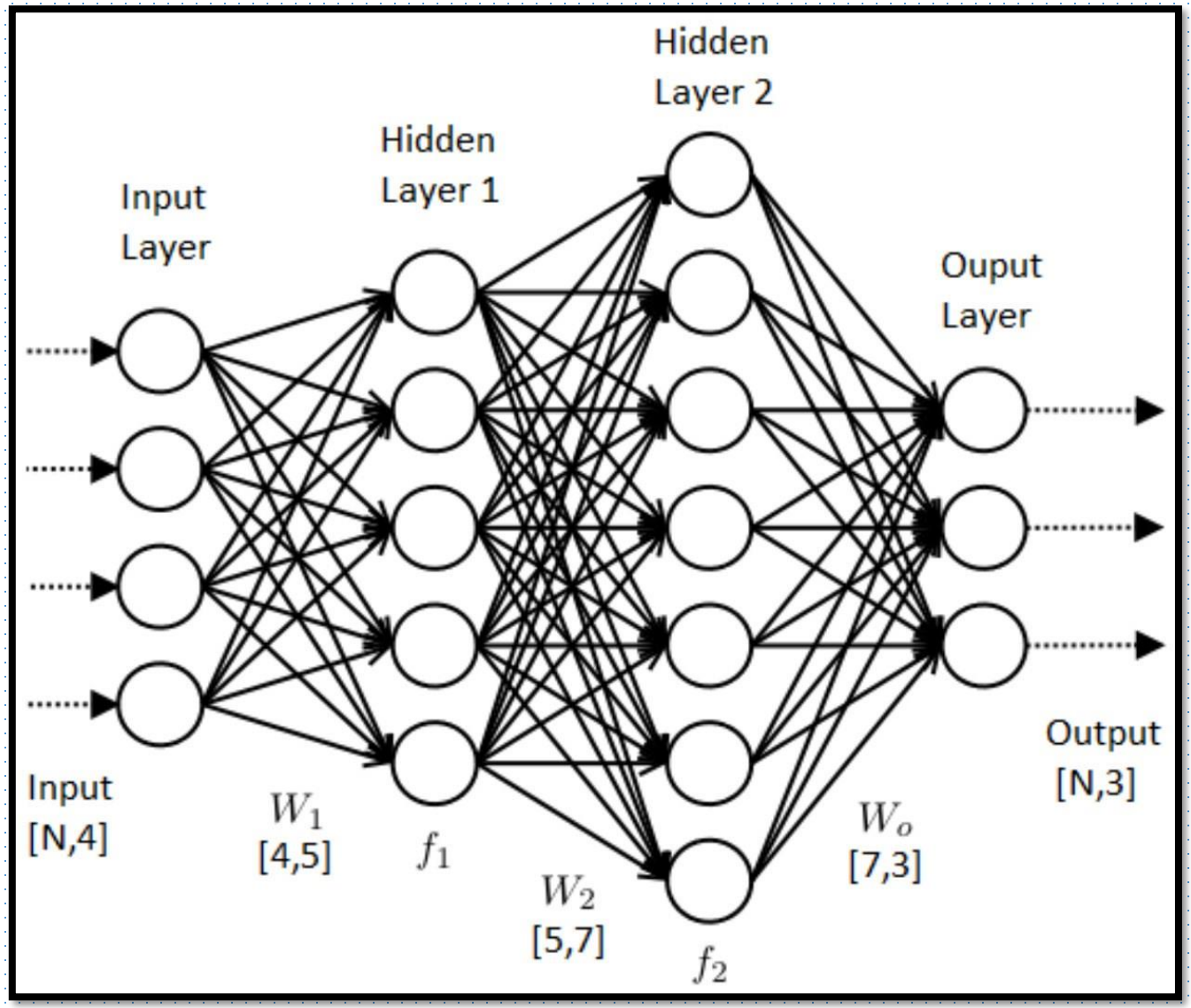




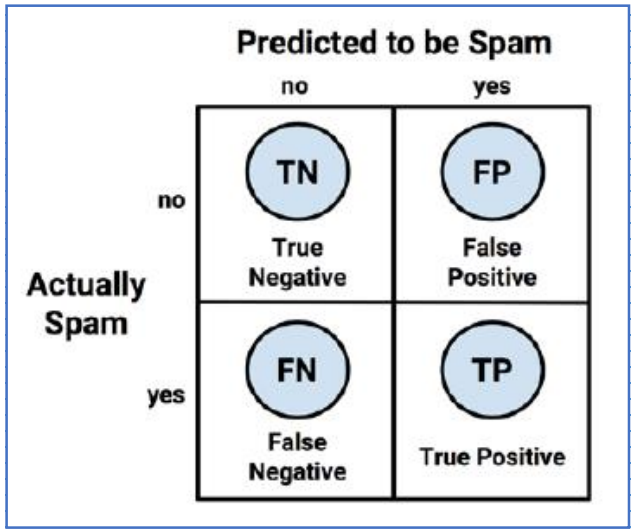


Essentially, the kernel trick involves a process of constructing new features that express mathematical relationships between measured characteristics.

Artificial Neural Networks



Model evaluation metrics



- **True Positive (TP):** Correctly classified as the class of interest
- **True Negative (TN):** Correctly classified as not the class of interest
- **False Positive (FP):** Incorrectly classified as the class of interest
- **False Negative (FN):** Incorrectly classified as not the class of interest

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{error rate} = \frac{\text{FP} + \text{FN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} = 1 - \text{accuracy}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}|$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y})^2$$

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y})^2}$$

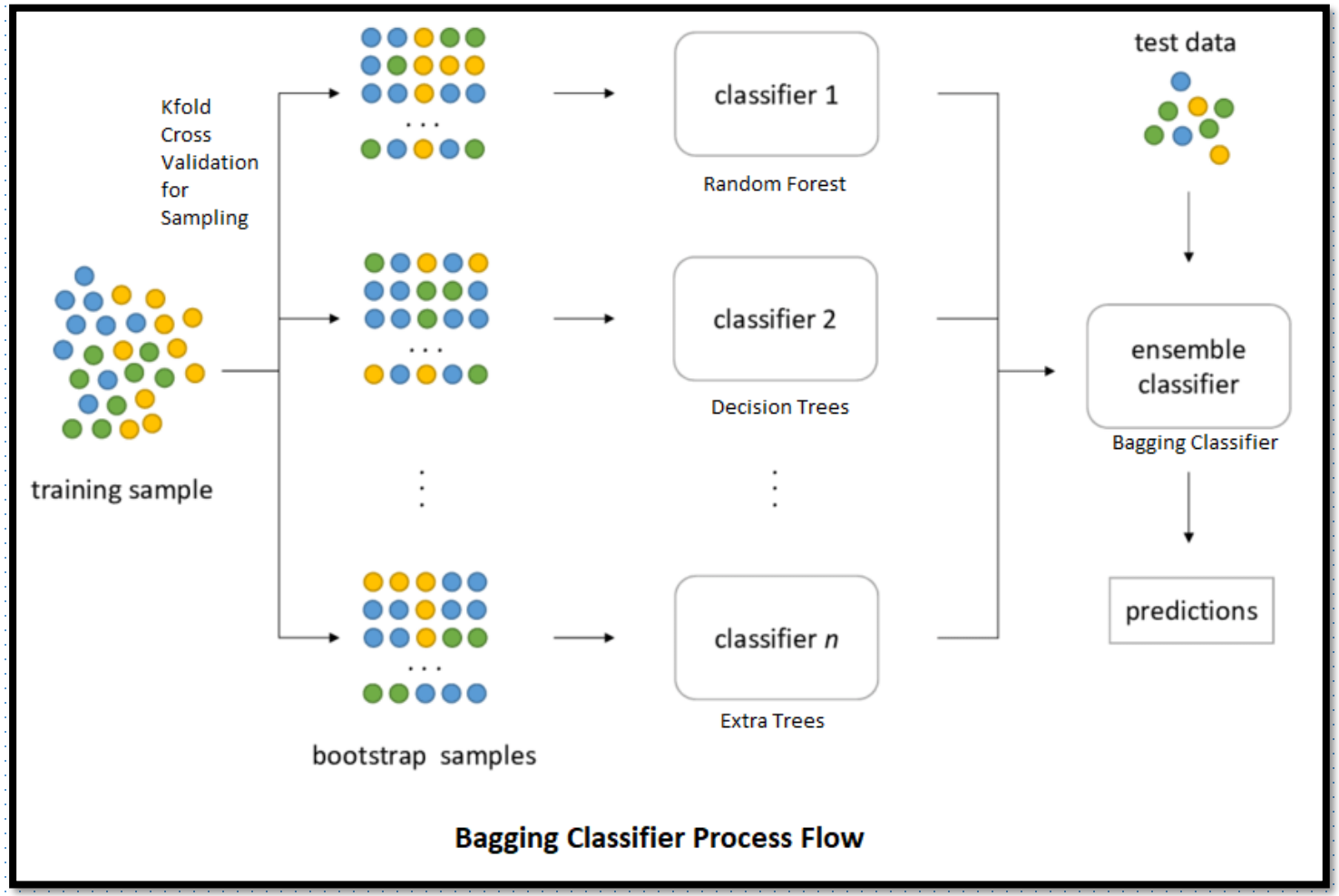
$$R^2 = 1 - \frac{\sum(y_i - \hat{y})^2}{\sum(y_i - \bar{y})^2}$$

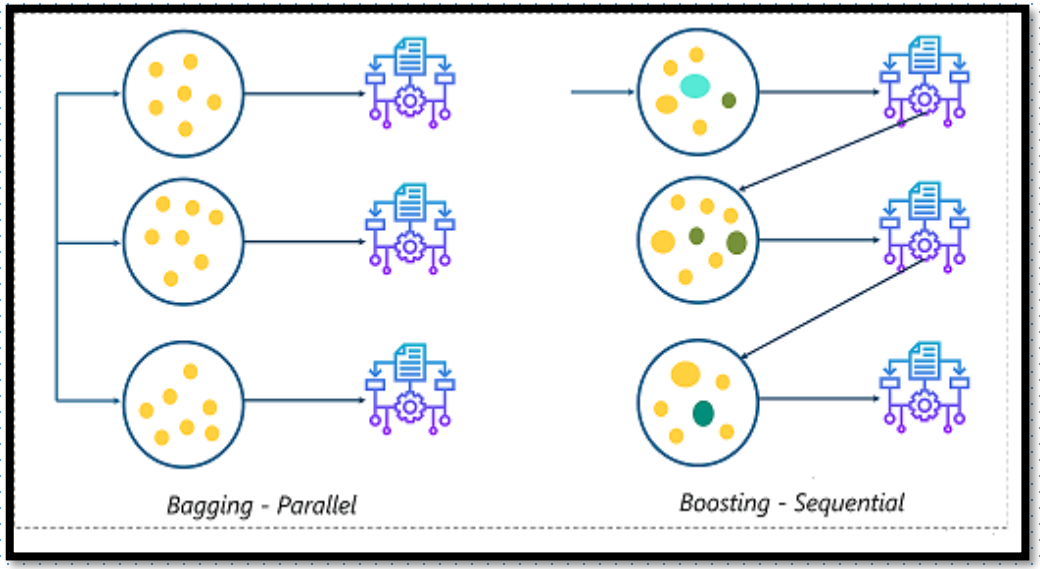
Where,

$\hat{y}$ – *predicted value of* y

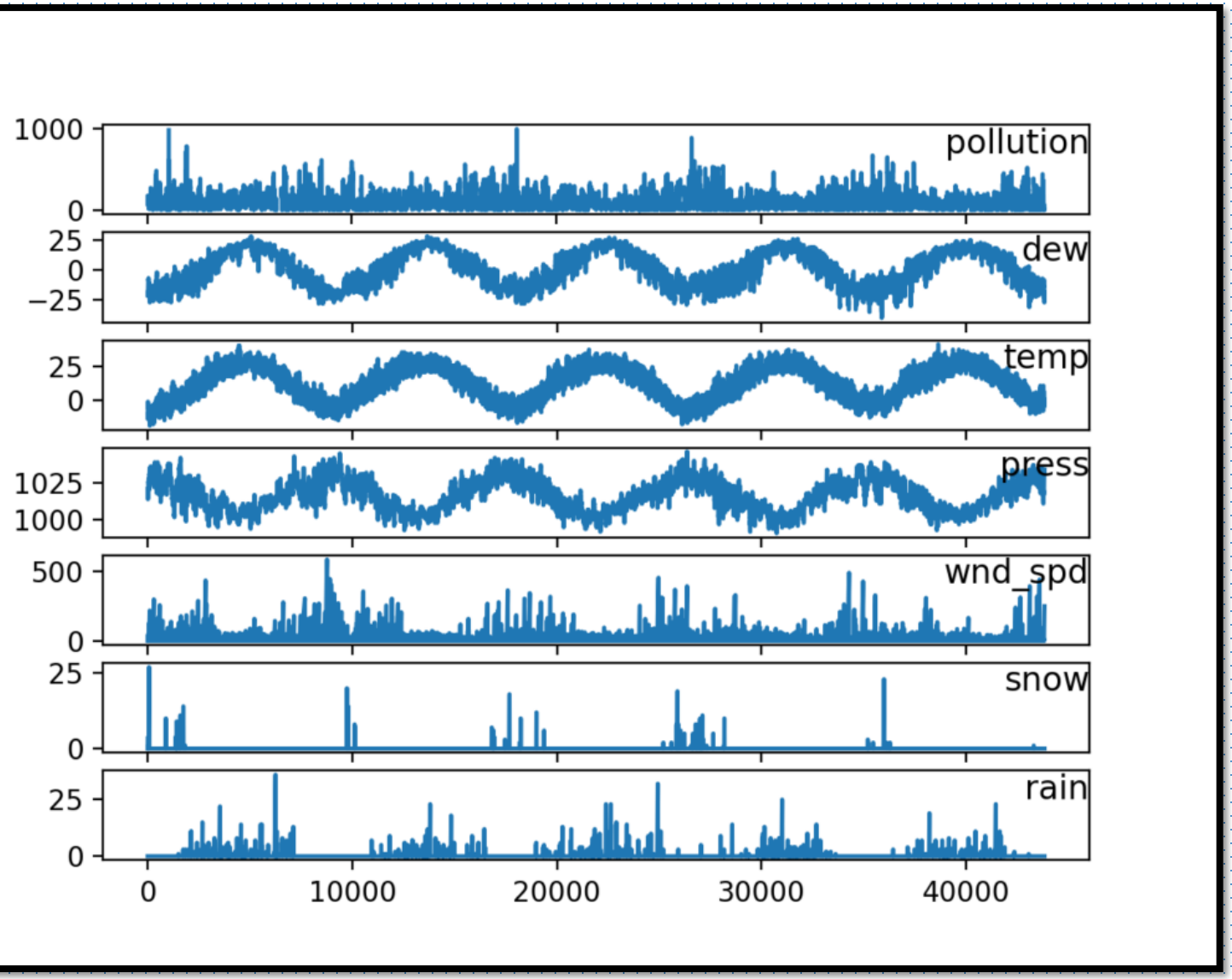
$\bar{y}$ – *mean value of* y

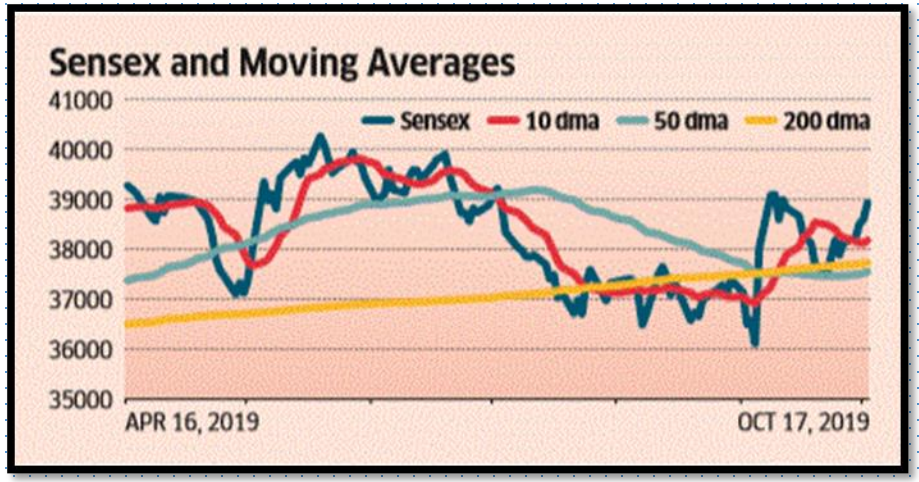
Ensemble methods





Time series data





Thank You