

# Naive Bayes and K Mean Clustering

## Naive Bayes

Naïve Baye is Classification (Supervise) Learning Algorithm.

Naïve Bayes uses Bayes' Theorem, combined with a ("naive") presumption of conditional independence, to predict the value of a target (output), from evidence given by one or more predictor (input) fields.

The Naive Bayes Classifier technique is based on Bayesian theorem and is particularly suited when the dimensionality of the inputs is high.

Despite its simplicity, Naive Bayes can often outperform more sophisticated classification methods.

Based on concept of Prior Probability and Likelihood and Posterior Probability

events knowledge guides subsequent similar events.

In this algorithm, it is assumed that every trait of the data being classified is independent of all other traits given

the class. This assumption of independent dataset traits grounds it's naming with the word naive. Bayes'

Theorem is denominated after prominent statistician Thomas Bayes.

It may be represented as:

$$P(h/e) = P(e/h) \cdot P(h) / P(e)$$

$$P(h) = P(e/h) \cdot P(h) / P(e)$$

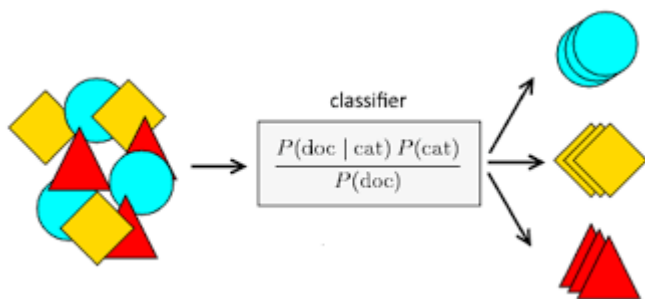
Eqn 1: Baye's Theorem [4]

- Where  $P(h)$  is the prior probability of hypothesis  $h$ .
- $P(h/e)$  is the posterior probability of hypothesis  $h$  (in presence of the evidence  $e$ ).
- $P(e/h)$  is the likelihood of evidence  $e$  on hypothesis  $h$ .

A. Proceedings

The elucidated equation for classification may be put in writing as:

$$P(\text{Class A} | \text{Trait 1}, \text{Trait 2}) = \frac{P(\text{Trait 1} | \text{Class A}) \cdot P(\text{Trait 2} | \text{Class A}) \cdot P(\text{Class A})}{P(\text{Trait 1}) \cdot P(\text{Trait 2})}$$



The diagram shows the Naive Bayes formula:  $P(c|x) = \frac{P(x|c)P(c)}{P(x)}$ . Arrows point from labels to the corresponding parts of the formula: 'Likelihood' points to  $P(x|c)$ , 'Class Prior Probability' points to  $P(c)$ , 'Posterior Probability' points to  $P(c|x)$ , and 'Predictor Prior Probability' points to  $P(x)$ .

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

## K means

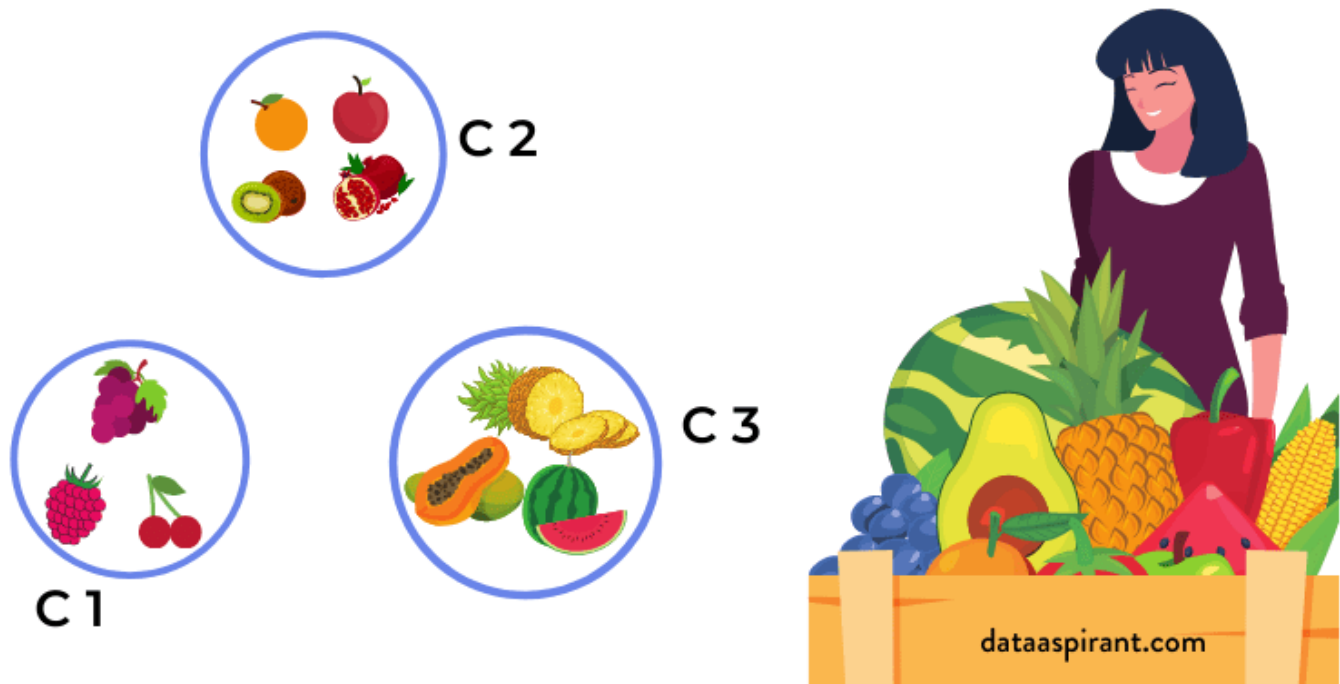
This algorithm divides M data points (which are in N dimensions) into K clusters in order to minimize the within-cluster sum of squares. We try to achieve the "local" optima solutions such that no inter-cluster data point manoeuvre reduces the within-cluster sum of squares. [1]  
Basically, this algorithm creates k clusters and pairs similar type of objects in a unique cluster. Thus k clusters are formed in such a way that the constituents of a certain cluster are similar as compared to the non-cluster constituents of a certain data set.

### A. Proceedings

Initially, k initial cluster centres are selected and then iteratively refined as:

1. Each instance  $d_i$  is assigned to its closest cluster centre.
2. Each cluster centre  $C_j$  is then updated and this becomes equivalent to the mean of its elemental instances. [2]

These steps are iterated until no further change is there in the apportionment of instances to clusters. Simply we may say that iterations are continued till cluster memberships are stabilized. This is called convergence.



# K Means Clustering

