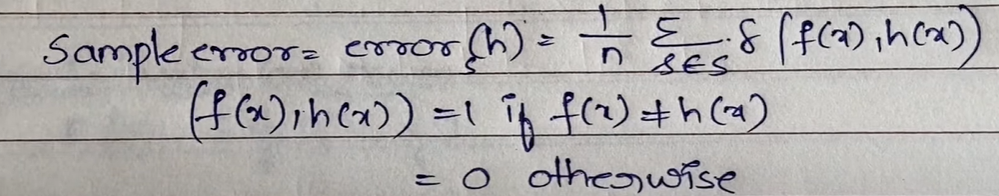
**Errors:**

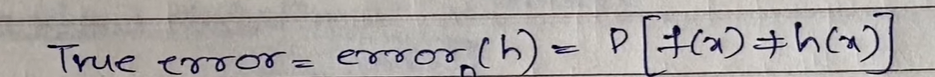
* Sample error
* True error
* Sample error:

This error rate of hypothesis over a sample of data

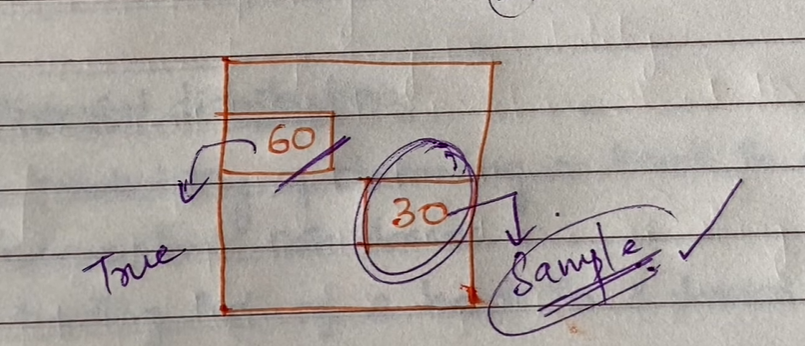


True error:

This error rate of hypothesis over the entire distribution



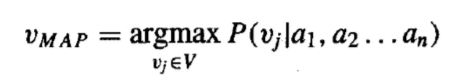
* Explain an example to understand



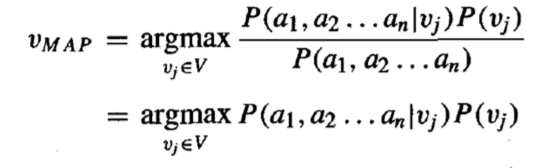
Calculating only 30 students data results to sample error where as calculating whole class students data leads to true error.

**Naive Bayes Classifier**

* One highly practical Bayesian learning method is Naive Bayes Learner (Naive Bayes Classifier).
* The naive Bayes classifier applies to learning tasks where each instance x is described by a conjunction of attribute values and where the target function f (x) can take on any value from some finite set V.
* A set of training examples is provided, and a new instance is presented, described by the tuple of attribute values (al, a2 ...an).
* The learner is asked to predict the target value (classification), for this new instance
* The Bayesian approach to classifying the new instance is to assign the most probable target value vMAP, given the attribute values (al, a2 ... an) that describe the instance.

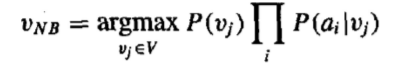


* By Bayes theorem: Naive Bayes Classifier



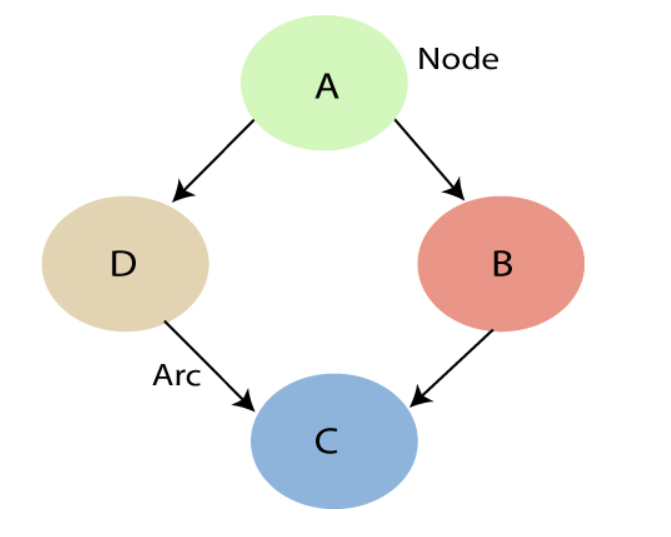
* It is easy to estimate each of the P(vj ) simply by counting the frequency with which each target value vj occurs in the training data.
* However, estimating the different P(al,a2…an | vj ) terms is not feasible unless we have a very, very large set of training data.
* The naive Bayes classifier is based on the simplifying assumption that the attribute values are conditionally independent given the target value.
* For a given the target value of the instance, the probability of observing conjunction al, a2...an, is just the product of the probabilities for the individual attributes:



* Naive Bayes classifier: 

**Bayesian Belief Network?**

* The naive Bayes classifier makes significant use of the assumption that the values of the attributes a1 . . .a, are conditionally independent given the target value v.
* However, in many cases this conditional independence assumption is clearly overly restrictive.
* Bayesian Belief Network or Bayesian Network or Belief Network is a Probabilistic Graphical Model (PGM) that represents conditional dependencies between random variables through a Directed Acyclic Graph (DAG).
* It is also called a **Bayes network, belief network, decision network**, or **Bayesian model**.
* Bayesian Network can be used for building models from data and experts opinions, and it consists of two parts:
  + Directed Acyclic Graph
  + Table of conditional probabilities.
* In contrast to the naive Bayes classifier, which assumes that all the variables are conditionally independent given the value of the target variable, Bayesian belief networks allow stating conditional independence assumptions that apply to subsets of the variables.
* In general, a Bayesian belief network describes the probability distribution over a set of variables.
* Consider an arbitrary set of random variables Yl . . . Y,, where each variable Yi can take on the set of possible values V(Yi).
* We define the joint space of the set of variables Y to be the cross product V(Yl) x V(Y2) x. . . V(Y,).
* The probability distribution over this joint' space is called the joint probability distribution.
* **A Bayesian network graph is made up of nodes and Arcs (directed links), where:**
* Each node corresponds to the random variables, and a variable can be continuous or discrete.
* **Arc or directed arrows** represent the causal relationship or conditional probabilities between random variables. These directed links or arrows connect the pair of nodes in the graph.



* In the above diagram, A, B, C, and D are random variables represented by the nodes of the network graph.
* If we are considering node B, which is connected with node A by a directed arrow, then node A is called the parent of Node B.
* Node C is independent of node A.
* If we have variables x1, x2, x3,....., xn, then the probabilities of a different combination of x1, x2, x3.. xn, are known as Joint probability distribution.

**EM Algorithm**

* Expectation Maximization
* Used to find latent variable
* Latent variables are those which can’t be observed
* Basic for many unsupervised clustering algorithm

Steps:

1. Initially, a set of initial values are considered

A set of incomplete data is given to the system

1. Next step - expectation step – E step

Here, we use observed data to estimate or guess the values of the missing data

1. Maximisation step or M – step

Some missing data will be there so we need to update the data

Here, we use the complete data generated in preceding e-step to update the values

1. We check if values are converging/not

If converging – stop

Otherwise repeat step 2 and step 3 till the convergence occurs

**Usage:**

* Used to fill missing data
* Used for unsupervised learning algorithm
* Used to discover values of latent variables