DataInquest

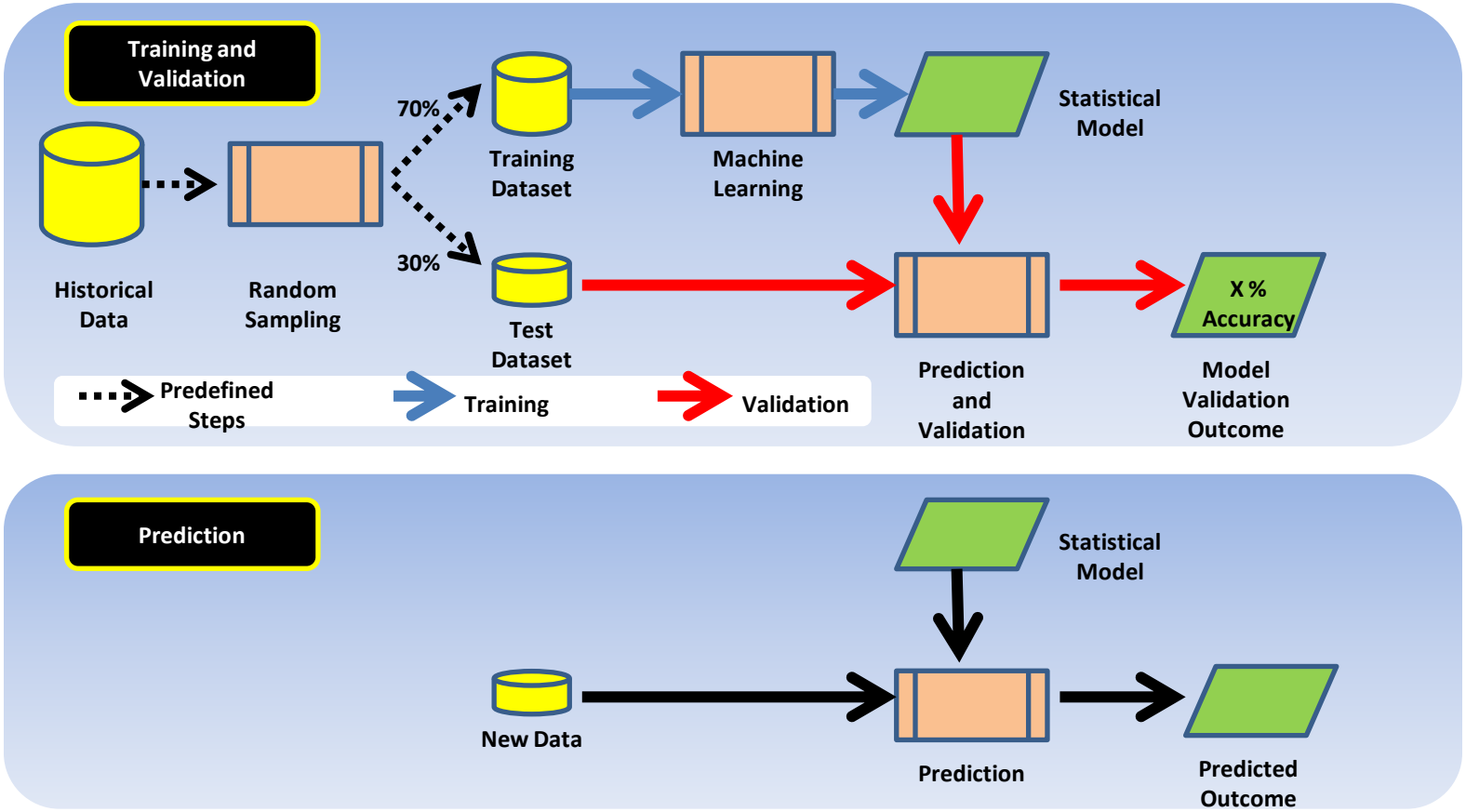
Big Data Technologies and Data Science training for all!!

Data Science

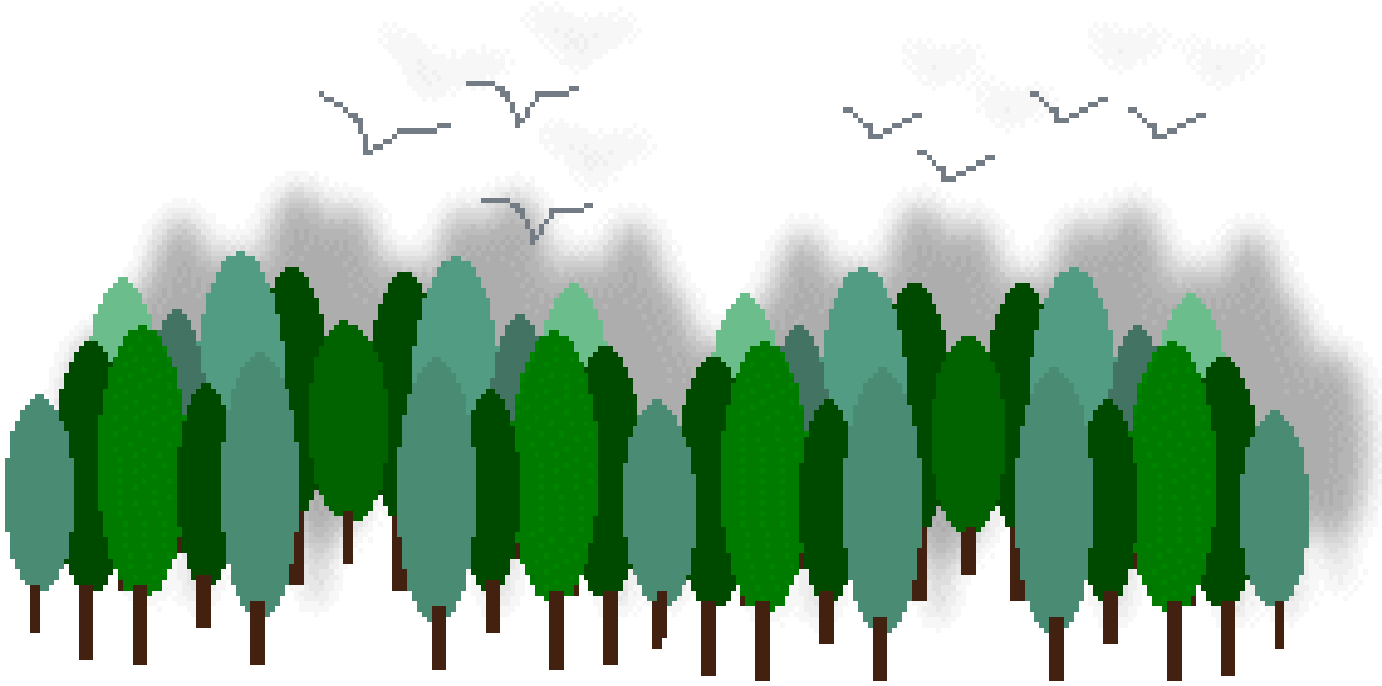
Topics for Today

- ✓ Random Forest Classifier
  - What is Random Forests
  - How Random Forest work?
  - Features of Random Forest
  - Out of Box Error Estimate and Variable Importance
  - Application of Technique on smaller datasets for better understanding using R software.
- ✓ Naïve Bayes Classifier
  - Introduction to Bayes Theorem
  - Understanding Naive Bayes Classifier with Example.
  - Application of Technique on smaller datasets for better understanding using R software.

Supervised Learning- Process Flow



Random Forest



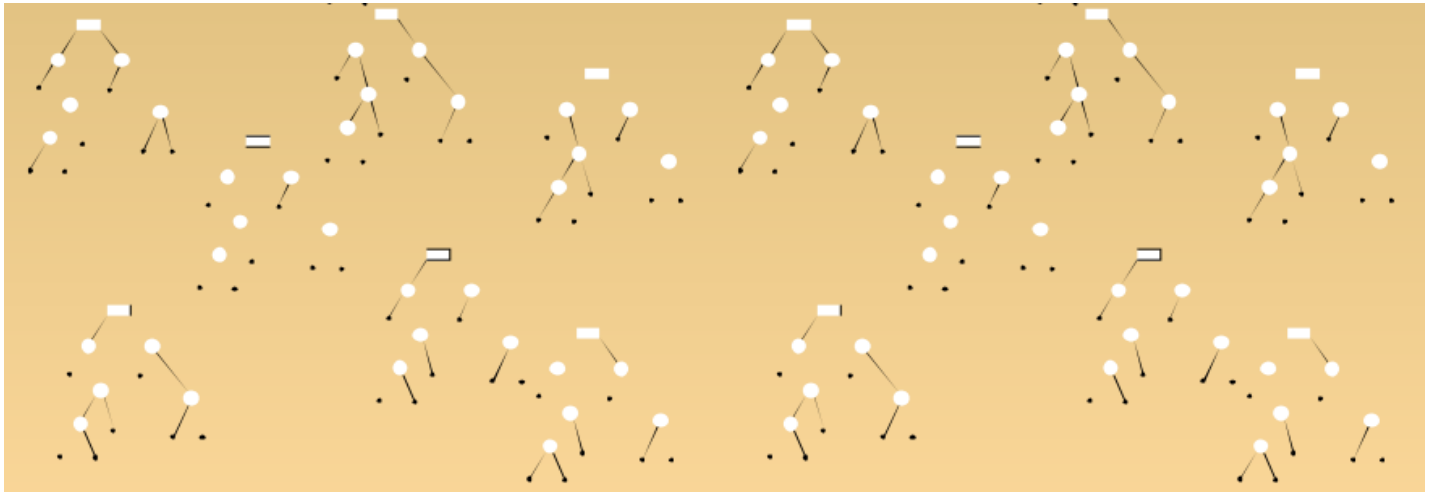
http://webclipart.about.com/library/weekly/aa041105.htm

What is Random Forests?

An ensemble classifier using many decision tree models

What are ensemble models?

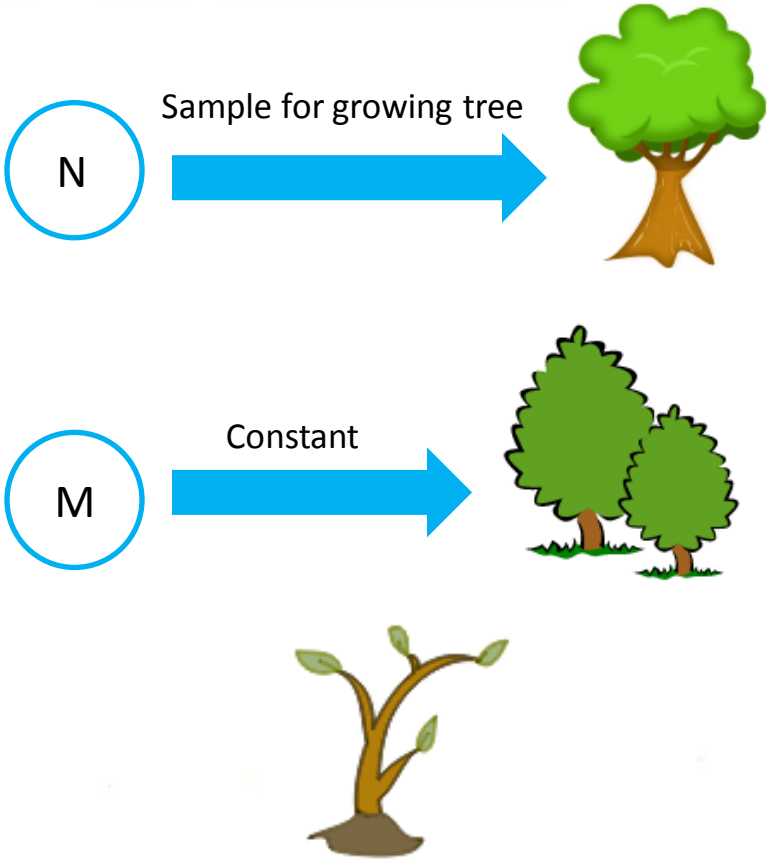
- ✓ Ensemble models combine the results from different models
- ✓ The result from an ensemble model is usually better than the result from one of the individual models



How Random Forests Work?

Each tree is grown as follows:

- ✓ If the number of cases (observations/ records) in the training set is N, sample N cases at random - but with replacement, from the original data. This sample will be the training set for growing the tree.
- ✓ If there are M input variables, a number m<<M is specified such that at each node, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held constant during the forest growing.
- ✓ Each tree is grown to the largest extent possible.

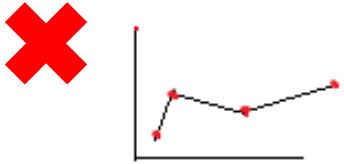


How Random Forests Work?

✓ A different subset of the training data are selected (~2/3), with replacement, to train each tree

Random forests

✓ Remaining training data (OOB) are used to estimate error and variable importance.



✓ Class assignment is made by the number of votes from all of the trees.

Features of Random Forests

- ✓ It is unexcelled in accuracy among current algorithms.
- ✓ It runs efficiently on large data bases.
- ✓ It can handle thousands of input variables without variable deletion.
- ✓ It gives estimates of what variables are important in the classification.
- ✓ It generates an internal unbiased estimate of the generalization error as the forest building progresses.

Features of Random Forests

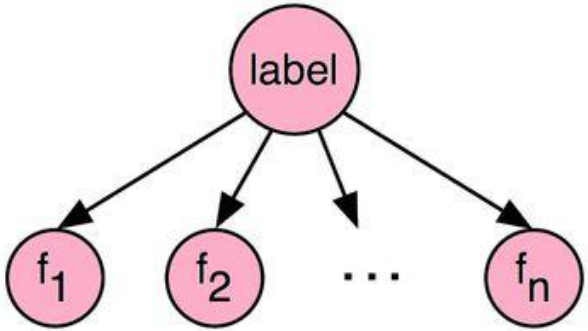
- ✓ It has an effective method for estimating missing data and maintains accuracy when a large proportion of the data are missing.
- ✓ It has methods for balancing error in class population unbalanced data sets.
- ✓ Generated forests can be saved for future use on other data.
- ✓ Prototypes are computed that give information about the relation between the variables and the classification.

The out-of-bag (oob) error estimate

In random forests, there is no need for cross-validation or a separate test set to get an unbiased estimate of the test set error. It is estimated internally, during the run, as follows

- ✓ Each tree is constructed using a different bootstrap sample from the original data.
- ✓ About one-third of the cases are left out of the bootstrap sample and not used in the construction of the kth tree.
- ✓ Put each case left out in the construction of the kth tree down the kth tree to get a classification.
- ✓ In this way, a test set classification is obtained for each case in about one-third of the trees.
- ✓ At the end of the run, take j to be the class that got most of the votes every time case n was oob.
- ✓ The proportion of times that j is not equal to the true class of n averaged over all cases is the oob error estimate. This has proven to be unbiased in many tests.

Naive Bayes Classifier

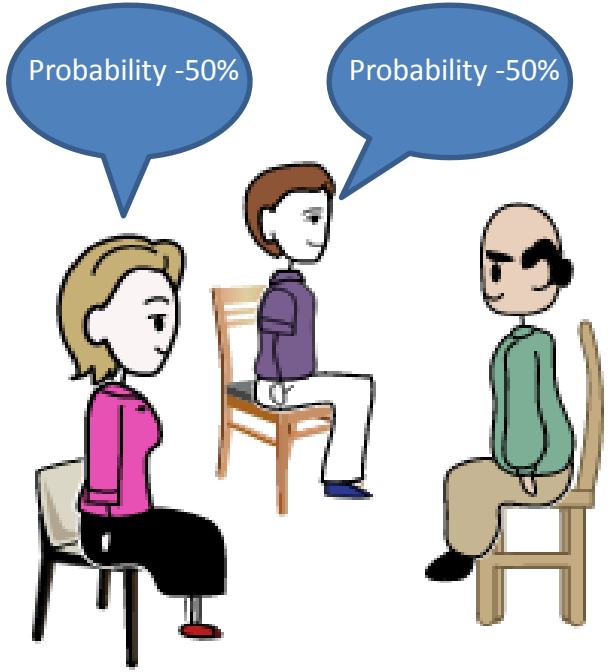


Naive Bayes Classifier Introductory Overview

The Naive Bayes Classifier technique is based on the so-called 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.

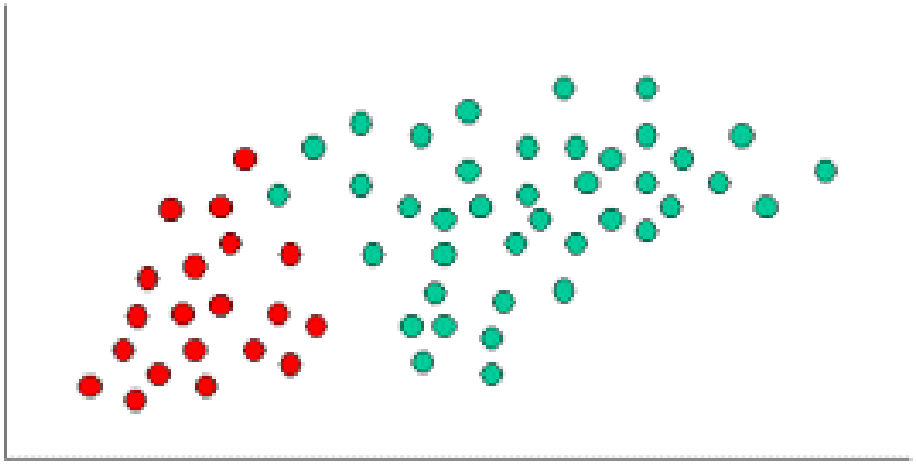
Introductory example to Bayes' Theorem:

Suppose a man told you he had had a nice conversation with someone on the train. Not knowing anything about this conversation, the probability that he was speaking to a woman is 50% (assuming the train had an equal number of men and women and the speaker was as likely to strike up a conversation with a man as with a woman). Now suppose he also told you that his conversational partner had long hair. It is now more likely he was speaking to a woman, since women are more likely to have long hair than men. Bayes' theorem can be used to calculate the probability that the person was a woman.



Naive Bayes Classifier- Example

As indicated, the objects can be classified as either GREEN or RED.



Our task is to classify new cases as they arrive, i.e., decide to which class label they belong, based on the currently exiting objects.

Naive Bayes Classifier-Example (Cont…)- Determine Prior Probability

Since there are twice as many **GREEN** objects as **RED**, it is reasonable to believe that a new case (which hasn't been observed yet) is twice as likely to have membership **GREEN** rather than **RED**. In the Bayesian analysis, this belief is known as the prior probability. Prior probabilities are based on previous experience, in this case the percentage of **GREEN** and **RED** objects, and often used to predict outcomes before they actually happen.

Thus, we can write:

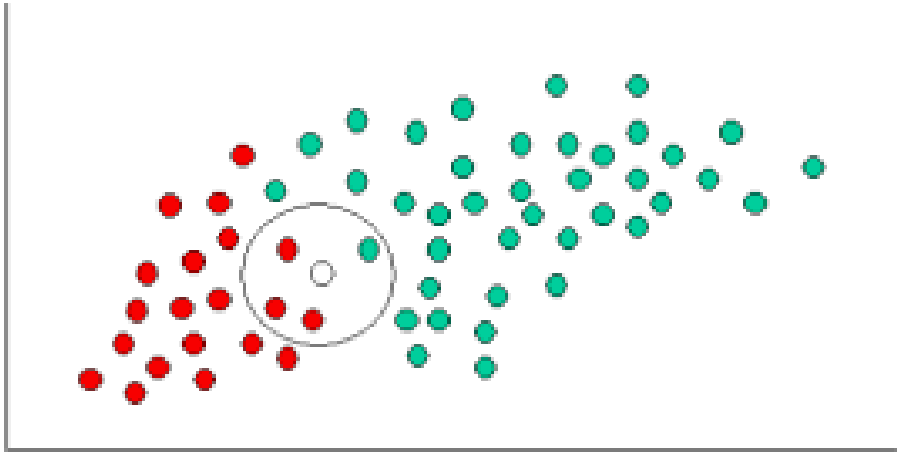
- ✓ Prior Probability of **GREEN**: $\frac{\text{No of green objects}}{\text{Total number of Objects}}$
- ✓ Prior Probability of **RED**: $\frac{\text{No of Red objects}}{\text{Total number of objects}}$
- ✓

Since there is a total of 60 objects, 40 of which are GREEN and 20 RED, our prior probabilities for class membership are:

- ✓ Prior Probability for **GREEN**: 40 / 60
- ✓ Prior Probability for **RED**: 20 / 60

Naive Bayes Classifier-Example (Cont…)- Determine Prior Probability

Having formulated our prior probability, we are now ready to classify a new object (WHITE circle in the diagram below). Since the objects are well clustered, it is reasonable to assume that the more GREEN (or RED) objects in the vicinity of X, the more likely that the new cases belong to that particular color. To measure this likelihood, we draw a circle around X which encompasses a number (to be chosen a priori) of points irrespective of their class labels. Then we calculate the number of points in the circle belonging to each class label. From this we calculate the likelihood:



Naive Bayes Classifier- Example (Continued) - Determine Likelihood

$$Likelihood\ of\ X\ given\ GREEN \propto \frac{Number\ of\ GREEN\ in\ the\ vicinity\ of\ X}{Total\ number\ of\ GREEN\ cases}$$

$$Likelihood\ of\ X\ given\ RED \propto \frac{Number\ of\ RED\ in\ the\ vicinity\ of\ X}{Total\ number\ of\ RED\ cases}$$

From the illustration above, it is clear that Likelihood of X given GREEN is smaller than Likelihood of X given RED, since the circle encompasses 1 GREEN object and 3 RED ones.

Thus:

$$Probability\ of\ X\ given\ GREEN \propto \frac{1}{40}$$

$$Probability\ of\ X\ given\ RED \propto \frac{3}{20}$$

Naive Bayes Classifier-Example(Cont..) - Determine Posterior Probability

Although the prior probabilities indicate that X may belong to **GREEN** (given that there are twice as many **GREEN** compared to **RED**) the likelihood indicates otherwise; that the class membership of X is **RED** (given that there are more **RED** objects in the vicinity of X than **GREEN**). In the Bayesian analysis, the final classification is produced by combining both sources of information, i.e., the prior and the likelihood, to form a posterior probability using the so-called Bayes' rule (named after Rev. Thomas Bayes 1702-1761).

$$\text{Posterior probability of } X \text{ being } GREEN \propto$$
$$\text{Prior probability of } GREEN \times \text{Likelihood of } X \text{ given } GREEN$$
$$= \frac{4}{6} \times \frac{1}{40} = \frac{1}{60}$$
$$\text{Posterior probability of } X \text{ being } RED \propto$$
$$\text{Prior probability of } RED \times \text{Likelihood of } X \text{ given } RED$$
$$= \frac{2}{6} \times \frac{3}{20} = \frac{1}{20}$$

Finally, we classify X as **RED** since its class membership achieves the largest posterior probability.

Naive Bayes Classifier- Fruit Example

Let's say that we have data on 1000 pieces of fruit. They happen to be Banana, Orange or some Other Fruit. We know 3 characteristics about each fruit:

- ✓ Whether it is Long
- ✓ Whether it is Sweet and
- ✓ If its color is Yellow.

This is our 'training set.' We will use this to predict the type of any new fruit we encounter.

Type	Long	Not Long	Sweet	Not Sweet	Yellow	Not Yellow	Total
Banana	400	100	350	150	450	50	500
Orange	0	300	150	150	300	0	300
Other Fruit	100	100	150	50	50	150	200
Total	500	500	650	350	800	200	1000

Naive Bayes Classifier- Fruit Example

We can pre-compute a lot of things about our fruit collection.

Prior Probabilities

The so-called "Prior" probabilities. (If we didn't know any of the fruit attributes, this would be our guess.) These are our base rates.

P(Banana) = 0.5 (500/1000)
P(Orange) = 0.3
P(Other Fruit) = 0.2

Probability of "Evidence"

p(Long) = 0.5
P(Sweet) = 0.65
P(Yellow) = 0.8

Probability of "Likelihood"

P(Long/Banana) = 0.8
P(Long/Orange) = 0 [Oranges are never long in all the fruit we have seen.]
....
P(Yellow/Other Fruit) = 50/200 = 0.25
P(Not Yellow/Other Fruit) = 0.75

Naive Bayes Classifier- Fruit Example - Given a Fruit, how to classify it?

Let's say that we are given the properties of an unknown fruit, and asked to classify it. We are told that the **fruit is Long, Sweet and Yellow.** Is it a Banana? Is it an Orange? Or Is it some Other Fruit?

We can simply run the numbers for each of the 3 outcomes, one by one. Then we choose the highest probability and 'classify' our unknown fruit as belonging to the class that had the highest probability based on our prior evidence (our 1000 fruit training set):

Naive Bayes Classifier- Fruit Example - Given a Fruit, how to classify it?

$$P(\text{Banana}/\text{Long, Sweet and Yellow}) = \frac{P(\text{Long}/\text{Banana})\, p(\text{Sweet}/\text{Banana}).P(\text{Yellow}/\text{Banana}) \times P(\text{banana})}{P(\text{Long}).\, P(\text{Sweet}).\, P(\text{Yellow})}$$

$$= \frac{0.8 \times 0.7 \times 0.9 \times 0.5}{P(\text{evidence})}$$

$$= 0.252/P(\text{evidence})$$

P(Orange/Long, Sweet and Yellow) = 0

P(Other Fruit/Long, Sweet and Yellow) = P(Long/Other fruit) x P(Sweet/Other fruit) x P(Yellow/Other fruit) x P(Other Fruit)
= (100/200 x 150/200 x 50/150 x 200/1000) / P(evidence)
= 0.01875/P(evidence)

By an overwhelming margin (0.252 >> 0.01875), we classify this Sweet/Long/Yellow fruit as likely to be a Banana.