Bayesian Classification

A statistical classifier: performs probability prediction.

Baye's Theorem: Basic

- Let X be a data sample ('evidence') : class label is unknown.
- Let H be a hypothesis that x belongs to class C.
- Classification is to determine P(H|X), (posteriori probability) : the probability that the hypothesis holds given the observed data sample x.
- P(H) (priori probability): the initial probability. E.g., x will buy the computer, regardless of age, income……
- P(X): probability that sample data is observed.
- P(X|H) (likelihood): the probability of observing sample x, given the hypothesis holds.

Given training data x, posteriori probability of a hypothesis H, P(H|X) follows the Bayes'.

P(H|X) = P(X|H)P(H)/P(X)

Posteriori = likelihood * prior*evidence

Example:

Imagine we want to predict if someone would love the 1990 movie Troll2 or not:

Love Troll 2	Popcorn(grams)	Soda(ml)	Candy(grams)
	24.3	750.2	0.2
	28.2	533.2	50.5
	etc	etc.	etc
Not Love Troll 2	Popcorn(grams)	Soda(ml)	Candy(grams)
	2.1	120.5	90.7
	4.8	110.9	102.3
	etc	etc.	etc

We calculated the means and standard distribution of popcorn, soda, and candy for love movie and not love movie. Then draw the Gaussian Distribution for each column at the center of mean and sd.

Now, someone new shows up….

Says they eat 20 grams popcorn, 500ml soda, and 25 grams candy every day. Let's use Guassian Naïve Bayes to predict if they love Troll 2 or not.

Initial Guesses: P(Love troll 2) = 0.5; P(not love troll 2) = 0.5

Terminology: The initial guesses are called prior probabilities.

Love Troll 2 score = P(loves Troll 2)* L(popcorn = 20|loves)*L(soda = 500|loves)*L(candy = 25|loves)

= 0.5 * 0.06*0.004* really small value.

Notes: When we get a really small number. It's a good idea to take the log() of everything to prevent something called Linderflow.

Love Troll 2 score = Log(0.5*0.06*0.004*small) = -124

Similarly:

Not Love Troll 2 score = -48

The score of not loving is greater than love troll 2. Thus, we classify the person as someone who does not love troll2.

If X_k is categorical, $P(X_k|C_i)$ is the # of tuples in C_i having value X_k divided by $|C_{i,d}|$(# of tuples of C_i in D).

Pros:

- Easy to implement
- Good results obtained in most of the cases
- Often considered 'the base line' for data mining operation.

Cons:

- Assumption: class conditional independence
- Practically, dependencies exist among variables