Naïve Bayes Algorithm

Naïve Bayes algorithm is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods. Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c).

$$P(c|x) = \frac{\boldsymbol{P(x|c)} \ \ P(c)}{P(x)}$$

$$P(c|x) = \frac{\boldsymbol{P(x_1|c)\, P(x_2|c)\, P(x_3|c)\, P(x_4|c) \ldots\ldots P(x_n|c)} \ \ P(c)}{P(x)}$$

$P(c|x)$ is the posterior probability of *class* (c, *target*) given *predictor* (x, *attributes*)
$P(x|c)$ is the likelihood which is the probability of *predictor* given *class*
$P(c)$ is the prior probability of *class*
$P(x)$ is the prior probability of *predictor*

$$P(c_1|x) = \frac{P(x|c_1) \ \ P(c_1)}{P(x)}$$

$$P(c_2|x) = \frac{P(x|c_2) \ \ P(c_2)}{P(x)}$$

Object belongs to C_1 if $P(c_1|x) > P(c_2|x)$; otherwise, object belongs to C_2.

$$P(c_1|x) = P(x_1|c_1)\, P(x_2|c_1)\, P(x_3|c_1)\, P(x_4|c_1) \ldots\ldots P(x_n|c_1)\, P(c_1)$$

$$P(c_2|x) = P(x_1|c_2)\, P(x_2|c_2)\, P(x_3|c_2)\, P(x_4|c_2) \ldots\ldots P(x_n|c_2)\, P(c_2)$$

Example: Playing Tennis

PlayTennis: training examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Learning Phase

P(Play=*Yes)* = 9/14

P(Play=*No)* = 5/14

Outlook	Play=*Yes*	Play=*No*
Sunny	*P* (Outlook= *Sunny* | Play=*Yes)* = 2/9	*P* (Outlook= *Sunny* | Play=*No)* = 3/5
Overcast	*P* (Outlook= *Overcast* | Play=*Yes)* = 4/9	*P* (Outlook= *Overcast* | Play=*No)* = 0/5
Rain	*P* (Outlook= *Rain* | Play=*Yes)* = 3/9	*P* (Outlook= *Rain* | Play=*No)* = 2/5

Temperature	Play=*Yes*	Play=*No*
Hot	*P* (Temperature= *Hot* | Play=*Yes)* = 2/9	*P* (Temperature= *Hot* | Play= *No)* = 2/5
Mild	*P* (Temperature= *Mild* | Play=*Yes)* = 4/9	*P* (Temperature= *Mild* | Play= *No)* = 2/5
Cool	*P* (Temperature= *Cool* | Play=*Yes)* = 3/9	*P* (Temperature= *Cool* | Play= *No)* = 1/5

Humidity	Play=*Yes*	Play=N*o*
High	*P* (Humidity= *High* | Play=*Yes)* = 3/9	*P* (Humidity= *High* | Play= *No)* = 4/5
Normal	*P* (Humidity= *Normal* | Play=*Yes)* = 6/9	*P* (Humidity= *Normal* | Play= *No)* = 1/5

Wind	Play=*Yes*	Play=*No*
Strong	*P* (Wind= *Strong* | Play=*Yes)* = 3/9	*P* (Wind= *Strong* | Play= *No)* = 3/5
Weak	*P* (Wind= *Weak* | Play=*Yes)* = 6/9	*P* (Wind= *Weak* | Play= *No)* = 2/5

Test Phase

Given a new instance,
x' = (Outlook=*Sunny,* Temperature=*Cool,* Humidity=*High,* Wind=*Strong*)

P(Outlook=*Sunny*|Play=*Yes*) = 2/9
P(Temperature=*Cool*|Play=*Yes*) = 3/9
P(Huminity=*High*|Play=*Yes*) = 3/9
P(Wind=*Strong*|Play=*Yes*) = 3/9
P(Play=*Yes*) = 9/14

P(Outlook=S*unny*|Play=*No*) = 3/5
P(Temperature=*Cool*|Play=*No*) = 1/5
P(Huminity=*High*|Play=*No*) = 4/5
P(Wind=*Strong*|Play=*No*) = 3/5
P(Play=*No*) = 5/14

P(*Yes*|**x**'): [P(*Sunny*|Yes) P(*Cool*|*Yes*) P(*High*|Yes) P(*Strong*|*Yes*)] P(Play=*Yes*) = 0.0053
P(*No*|**x**'): [P(*Sunny*|No) P(*Cool*|No) P(*High*|*No*) P(*Strong*|*No*)] P(Play=*No*) = 0.0206

Given the fact P(*Yes*|**x**') < P(*No*|**x**'), we label **x**' to be "*No*".