NAIVE BAYES ALGO

Naive Bayes classifier is a probabilistic machine learning model that's used ~~for~~ for classification task.

The crux of the classifier is based on Bayes theorem.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

hypothesis. evidence.

⇒ The assumption made here is that the predictors / features are independent. ie. the presence of one particular feature doesn't affect the others. ⇒ Hence, it is called Naive.

⇒ Or we can say that all the predictors have an equal effect on the outcome.

y = class variable output

$x_1, x_2 \ldots x_n \rightarrow$ input features.

$$P(y | x_1, x_2 \ldots x_n) = \frac{P(x_1|y) \cdot P(x_2|y) \ldots P(x_n|y) \cdot P(y)}{P(x_1) \cdot P(x_2) \cdot P(x_3) \ldots P(x_n)}$$

for all the entries, denominator doesn't change, it remains static. ∴, the denominator can be removed and a proportionality can be used.

$$P(y | x_1 \ldots x_n) \propto P(y) \cdot \prod_{i=1}^{n} P(x_i | y)$$

we need to find the class 'y' with the max. prob.

Scanned by CamScanner

$y = \underset{y}{\operatorname{argmax}}\ P(y) \cdot \prod_{i=1}^{n} P(x_i | y)$ → using this function we obtain the class given the predictors.

Types of Naive Bayes classifiers:

① **Multinomial Naive Bayes:**

mostly used for document classification problem.
whether a document belongs to a category of sports, politics, technology etc.

The features / predictors used by the classifier are the freq. of the words present in the document.

② **Bernoulli Naive Bayes:**

Similar to multinomial, here the predictors are boolean variables.

The parameters that we use to predict the class variable take up only values yes or no.

ex: a word occurs in a text or not.

③ **Gaussian Naive Bayes:**

when the predictors take up a continuous value and are not discrete, we assume that these values are sampled from gaussian distribution.

$$P(x_i | y) = \frac{1}{\sqrt{2\pi}\,\sigma_y} e^{-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}}$$

Scanned by CamScanner

⇒ Naive Bayes algo. are mostly used in sentimental analysis, spam filtering, recommendation systems etc.

They are fast and easy to implement but their biggest disadv. is that the requirement of predictors to be independent.

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

likelihood

class prior prob.

posterior prob.

predictor prior prob.

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

Note:

① It performs well in case of categorical input variables compared to numerical variables.

② For numerical variables, normal distribution is assumed (a very strong assumption).

⇒ ③ If the categorical variable has a category (test data) which was not observed in the training dataset then the model will assign 0 prob. and will be unable to make a prediction.

This is often called as "zero frequency". To solve this we can use smoothing technique. ex: Laplace Estimation.

Applications:

① Real time prediction - eager learning classifier.

② Multi class prediction.

③ Text classification / sentimental analysis.

④ Spam filtering.

⑤ Recommendation systems and CF

Scanned by CamScanner

ex3 will the player play Tennis.

$\boxed{\text{Outlook}}$ ← input feature - 1

	Yes	No	P(Yes)	P(No)
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0
Rainy	3	2	3/9	2/5
Total :	9	5	1	1

$\boxed{\text{Play}}$ ← output class

Yes	9	$P(\text{Yes}) = \frac{9}{14}$
No	5	$P(\text{No}) = \frac{5}{14}$

$\boxed{\text{Temperature}}$ ← input feature - 2

	Yes	No	P(Yes)	P(No)
Hot	2	2	2/9	2/5
Mild	4	2	4/9	2/5
Cool	3	1	3/9	1/5
Total :	9	5	1	1

Today (Sunny, Hot) (x_1, x_2)

$$P(\text{Yes} \mid \text{Today}) = P(\text{Sunny} \mid \text{yes}) \times P(\text{Hot} \mid \text{yes}) \times P(\text{yes})$$

Note: deno. could be ignored.

$$= \frac{2}{9} \times \frac{2}{9} \times \frac{9}{14} = 0.031$$

$$P(\text{No} \mid \text{Today}) = P(\text{Sunny} \mid \text{No}) \times P(\text{Hot} \mid \text{No}) \times P(\text{No})$$

$$= \frac{3}{5} \times \frac{2}{5} \times \frac{5}{14} = 0.085$$

$\boxed{\text{output} = \text{No}}$

$$P(\text{Yes}) = \frac{0.031}{0.031 + 0.085} = 0.27$$

$$P(\text{No}) = 1 - 0.27 = 0.73$$

$P(\text{No}) > P(\text{Yes})$

Scanned by CamScanner