Lesson_7: Naive Bayes Classifiers

Ali Aburas, PhD

Slide credit:Andrew Ng

Naïve Bayes Machine Learning

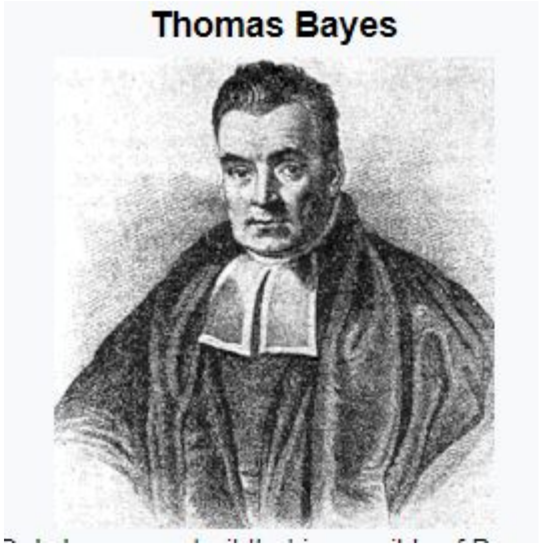
- The **Naive Bayes** classifier is a supervised machine learning algorithm designed to assign a label or category to an object based on its features.
- The **Naive Bayes** classifier is a simple but effective probabilistic learning algorithm based on the Bayes theorem with strong independence assumptions among the features.
- **Naive Bayes** has been proved a great classifier not only for text related tasks like spam filtering, and sport events classification but also to predict medical diagnosis.
- **Why is it called Naïve Bayes?**
    - **Naïve**: It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features.
    - **Bayes**: It is called Bayes because it depends on the principle of Bayes' Theorem.

Bayes' Theorem

1. Theory behind the Naive Bayes classifiers

- **Bayes' Theorem**, named after 18th-century British mathematician Thomas Bayes, is a mathematical formula for determining conditional probability.

- **Conditional probability** is the likelihood of an outcome occurring, based on a previous outcome having occurred in similar circumstances.



Key theoretical concepts

- **Prior Probability**: The initial probability of an event occurring before observing any evidence. For example, the probability that a patient has the flu without considering their symptoms.
- **Conditional (Posterior) Probability**: The probability of an event occurring given that another event has already occurred. For instance, the probability that a patient has a fever given that they have the flu.
- **Bayes' Theorem statement**: Simply put, the probability of A given B equals the probability of B given A, multiplied by the probability of A, divided by the probability of B.

- **P(A)** is the **priori** of A (the prior probability, i.e. Probability of event before evidence is seen).

- **P(A|B)** is a posteriori probability of B, i.e. probability of event after evidence is seen.

Probability of B occurring given evidence A has already occurred

Probability of A occurring

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

Probability of A occurring given evidence B has already occurred

Probability of B occurring

The formula of Bayes' Theorem.

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

- **P(A|B)**: This is the probability that event A is true given that B is true. In other words, if we know B has occurred (e.g., the email contains the word "offer"), we want to determine how likely it is that A is true (the email is spam).
- **P(B|A)**: This is the probability that B is true if A is true. So, if we know the email is spam (A), this probability tells us how likely it is to contain the word "offer" (B).
- **P(A):** This represents the prior probability of A, meaning how likely it is that an email is spam before considering specific features (B).
- **P(B):** This is the prior probability of B, which refers to how likely it is to observe feature B in general ("offer"), regardless of A.

Naive Bayes Classifier

- The **fundamental Naïve Bayes assumption** is that each feature makes an:
  - **Independent** : We assume that no pair of features are dependent.
  - **Equal**: each feature is given the same influence(or importance).

Bayes theorem can be rewritten as:

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)}$$

$$X = (x_1, x_2, x_3, \ldots.., x_n)$$

Here x_1, x_2…, x_n represent the features. By substituting for X and expanding using the chain rule we get,

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

Naive Bayes Classifier

$$P(y|x_1,...,x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

We can conveniently simplify the previous Eq. by dropping the denominator *P(x)*:

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

Now, we find the probability of given set of inputs for all possible values of the class variable *y* and pick up the output with maximum probability. This can be expressed mathematically as:

$$y = argmax_y P(y)\prod_{i=1}^{n} P(x_i|y)$$

Naive Bayes Classifier Example

- Consider the car theft problem with attributes Color, Type, Origin, and the target, Stolen can be either Yes or No.
- Here in our dataset, **we need to classify whether the car is stolen, given the features of the car.**
- The columns represent these **features (X)** and the rows represent individual **entries(Y)**.

Example No.	Color	Type	Origin	Stolen?
1	Red	Sports	Domestic	Yes
2	Red	Sports	Domestic	No
3	Red	Sports	Domestic	Yes
4	Yellow	Sports	Domestic	No
5	Yellow	Sports	Imported	Yes
6	Yellow	SUV	Imported	No
7	Yellow	SUV	Imported	Yes
8	Yellow	SUV	Domestic	No
9	Red	SUV	Imported	No
10	Red	Sports	Imported	Yes

$$\textit{New Instance} = (\textit{Red}, \textit{SUV}, \textit{Domestic})$$

Example No.	Color	Type	Origin	Stolen?
1	Red	Sports	Domestic	Yes
2	Red	Sports	Domestic	No
3	Red	Sports	Domestic	Yes
4	Yellow	Sports	Domestic	No
5	Yellow	Sports	Imported	Yes
6	Yellow	SUV	Imported	No
7	Yellow	SUV	Imported	Yes
8	Yellow	SUV	Domestic	No
9	Red	SUV	Imported	No
10	Red	Sports	Imported	Yes

$$y = argmax_y P(y) \prod_{i=1}^{n} P(x_i|y)$$

$$p(Yes) = \frac{5}{10} = 0.5$$

$$p(No) = \frac{5}{10} = 0.5$$

Color	Yes	No
Red	3/5	2/5
Yellow	2/5	3/5

Origin	Yes	No
Domestic	2/5	3/5
Imported	3/5	2/5

$$New\ Instance = (Red, SUV, Domestic)$$

Type	Yes	No
Sports	4/5	2/5
SUV	1/5	3/5

$$P(Yes \mid New\,Ins\tan ce) = P(Yes) \cdot P(Color = Red \mid Yes) \cdot P(Type = SUV \mid Yes) \cdot P(Origin = Domestic \mid Yes)$$

$$P(Yes \mid New\,Ins\tan ce) = \frac{5}{10} \cdot \frac{3}{5} \cdot \frac{1}{5} \cdot \frac{2}{5} = \frac{3}{125} = 0.024$$

$$P(No \mid New\,Ins\tan ce) = P(No) \cdot P(Color = Red \mid No) \cdot P(Type = SUV \mid No) \cdot P(Origin = Domestic \mid No)$$

$$P(No \mid New\,Ins\tan ce) = \frac{5}{10} \cdot \frac{2}{5} \cdot \frac{3}{5} \cdot \frac{3}{5} = \frac{9}{125} = 0.072$$

Since **0.077 > 0.024**, Which means given the features **RED SUV and Domestic**, our example gets classified as 'NO' **the car is not stolen**.

Types of Naïve Bayes Model:

- **Gaussian** : It is a straightforward algorithm used when the attributes are continuous. This assumes that the X are continuous and have been sampled from a Gaussian distribution.
    - The data near the mean are more frequent in occurrence than data far from the mean and **bell-shaped**. Examples include: Heights of humans, IQ scores, Quality Control.
- **Multinomial** : Multinominal Naive Bayes is used on documentation classification issues. The features needed for this type are the frequency of the words converted from the document. It is basically used for document classification, like a particular document which deals with what category like Sports, Politics or education etc.

- **Bernoulli** : Bernoulli Naive Bayes is an algorithm that is useful for data that has binary or boolean attributes. The attributes will have a value of yes or no, useful or not, granted or rejected, etc. For example if some word exists in a document or not so. This model is also one of the most popular for document classification tasks.

- **For more information, look** at naive bayes scikit learn

Naïve Bayes Classifier

Advantages of Naïve Bayes Classifier:

- Simple and easy to implement.
- Requires only a small amount of training data.
- Works well with high-dimensional data.

Disadvantages of Naïve Bayes Classifier:

- Assuming feature independence. This assumption doesn't hold in any actual data.
  - Naive Bayes assumes that all features are independent or unrelated, so it cannot learn the relationship between features.
- If the independent assumption isn't met, it can be not as accurate as more-complicated models.

Real-world problems

- **Text classification**: The Naive Bayes Algorithm is used as a probabilistic learning technique for text classification. It is one of the best-known algorithms used for document classification of one or many classes.
- **Sentiment analysis**: The Naive Bayes Algorithm is used to analyze sentiments or feelings, whether positive, neutral, or negative.
- **Spam filtering**: It is also similar to the text classification process. It is popular for helping you determine if the mail you receive is spam.
- **Medical diagnosis**: This algorithm is used in medical diagnosis and helps you to predict the patient's risk level for certain diseases.
- **Weather prediction**: You can use this algorithm to predict whether the weather will be good.
- **Face recognition**: This helps you identify faces.
- **Recommendation system**: The Naive Bayes Algorithm is a collection of collaborative filtering issued for building hybrid recommendation systems that assist you in predicting whether a user will receive any resource.