Naïve Bayes Classifier

JELENA JOVANOVIĆ

Email: jeljov@gmail.com

Web: http://jelenajovanovic.net

WHY NAÏVE BAYES?

Naïve Bayes (NB) is often cited as an algorithm that is among the first to be considered for any classification task

Rationale:

- Simplicity*
- Good performance
- High scalability
- Adaptable to almost any kind of classification task

*Occam’s Razor: “Other things being equal, simple theories are preferable to complex ones”

To recall: Bayes rule

$$P(H|E) = \frac{P(E|H) * P(H)}{P(E)}$$

- H – hypothesis
- E – evidence related to the hypothesis H, i.e., the data to be used for validating (accepting/rejecting) the hypothesis H
- P (H) – probability of the hypothesis (*prior probability*)
- P (E) – probability of the evidence i.e., the state of the world described by the gathered data
- P (E | H) – (conditional) probability of evidence E given that the hypothesis H holds
- P (H | E) – (conditional) probability of the hypothesis H given the evidence E

BAYES RULE – AN EXAMPLE

Let us suppose the following:

- one morning, you wake up with a high temperature
- the previous day, you heard that some virus infection had started spreading through the city, though the infection rate was still rather low, namely 2.5%
- you've also heard that in 50% of cases, the virus went with a high temperature
- you have a high temperature only a couple of times over a year, so, let's say that the probability that you have a high temp. is 5%

Question: what is the probability that, since you have a high temperature, you've caught the virus?

BAYES RULE

Theory	Example
Hypothesis (H)	One has caught a virus infection
P(H)	0.025
Evidence (E)	One has a high temperature
P(E)	0.05
(conditional) probability of E given H P(E|H)	Probability that the virus infection causes high temperature 0.50
(conditional) probability of H given E: P(H|E)	Probability that given one has a high temperature, he/she also has the virus ?

$$P(H|E) = P(E|H) * P(H) / P(E)$$

$$P(H|E) = 0.50 * 0.025 / 0.05 = 0.25$$

NB CLASSIFIER

If there is a class ***c*** and an observation ***o***, following the Bayes rule, the probability that the observation ***o*** is of class ***c*** is:

$$P(c|o) = P(o|c) * P(c) / P(o) \qquad (1)$$

For the given set of classes ***C*** and an observation ***o***, we want to find class ***c**,* from the set ***C**,* with the highest conditional probability for the observation ***o***; this leads to the function:

$$f = argmax_{c \in C} P(c|o) \qquad (2)$$

By applying the Bayes rule, we get:

$$f = argmax_{c \in C} P(o|c) * P(c) \qquad (3)$$

NB CLASSIFIER

$$f = argmax_{c \in C} P(o|c) * P(c) \qquad (3)$$

Now, we need to *estimate* the probabilities P(c) and P(o|c)

P(c) can be computed rather easily: by counting the number of occurrences of the class ***c*** in the training set

P(o|c) – probability that in the class ***c*** one would “find” the observation ***o*** – not that easy to determine, so we introduce an assumption that gave this algorithm the epithet “naïve”

NB CLASSIFIER

How do we determine $P(o|c)$?

- we represent the observation ***o*** as a vector of its attributes $(x_1, x_2, ..., x_n)$, also known as *feature vector*
- so, instead of $P(o|c)$, we'll have $P(x_1, x_2, x_3, ... x_n|c)$
- to compute $P(x_1, x_2, x_3, ... x_n|c)$, we introduce the following naïve assumption:
  - attributes that describe observation ***o*** are *mutually independent*, i.e., ***o*** can be considered as a simple set (bag) of attributes

NB CLASSIFIER

Based on the introduced assumption, $P(x_1, x_2,...,x_n |c)$ can be represented as a product of individual conditional probabilities

$$P(x_1, x_2,...,x_n |c) = P(x_1|c) * P(x_2|c) * \dots * P(x_n|c)$$

Thus, we arrive to the general equation of the NB algorithm:

$$f = argmax_{c \in C} P(c) * \prod_{i=1}^{n} P(x_i|c)$$

NB CLASSIFIER

The introduced assumption

- (–) often is invalid
- (–) often leads to a loss of information that could have been derived from the data
- (+) simplifies the computation of $P(x_1, x_2,...,x_n|c)$, and thus simplifies the overall classification task

NB CLASSIFIER

Probabilities $P(c)$ and $P(x_i|c)$ are estimated on the training set in the following way:

$P(c)$ is the ratio of the number of observations of the class ***c*** and the total number of observations in the training set

$P(x_i|c)$ is determined from the distribution of the attribute x_i in the observations of the class ***c***; the computation depends on the type of attribute (nominal or numeric)

An example of $P(x_i|c)$, in the context of the Titanic classification task:

P(Pclass='1st'| Survived='Yes') =
count(Pclass='1st' & Survived='Yes') / count(Survived='Yes')

Characteristics of the NB algorithm

- Very fast and efficient
- Often produces good results
  - often turns out to be better or at least equally good as other, more sophisticated algorithms
- Does not require much memory
- Has low affinity to over-fitting
- Performs well even with small training set

CHARACTERISTICS OF THE NB ALGORITHM

- "Resistant" to the low-importance attributes
  - attributes that are equally distributed through the overall training set, and thus do not have significant impact on the class label
- Primarily suitable for use with nominal attributes; in the case of numerical attributes
  - Discretize the attribute values, or
  - Use probability distribution of the attributes to estimate the probability of each attribute value
    - e.g., if attribute *x* is normally distributed, we use density f. of the Normal distribution to compute probabilities:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$