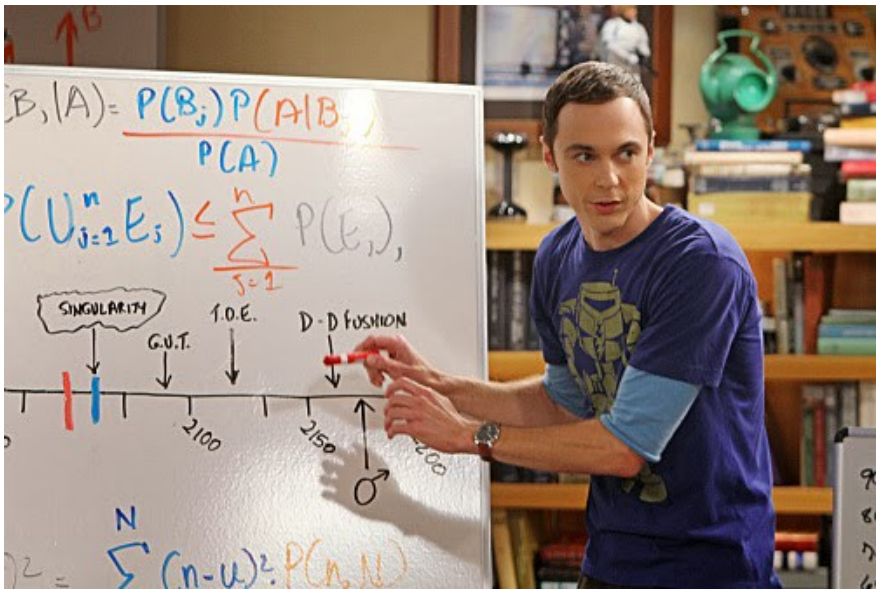
Yash Sharma Follow
Mar 30, 2017 · 6 min read

Naive Bayes. Unfolded.

A few days back, when I told my friend that having no girlfriend in college and having no girlfriend in school are two independent events, I was told that I am being ***Naive***.
The same happened to Naive Bayes Algorithm when it *assumed* that all the features are independent of each other.



As celebrated as it could be!

So, with the 9th blog in our 12A12D series, we explore the genius of this algorithm.

> ***Every true genius is bound to be naive.***

. . .

What is Naive Bayes?

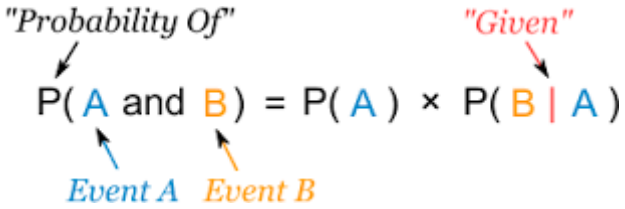
- Collection of classification algorithms based on ***Bayes Theorem***.
- Classifies given different instances (object/data) into predefined classes(groups), assuming there is no interdependency of features (class conditional independence).

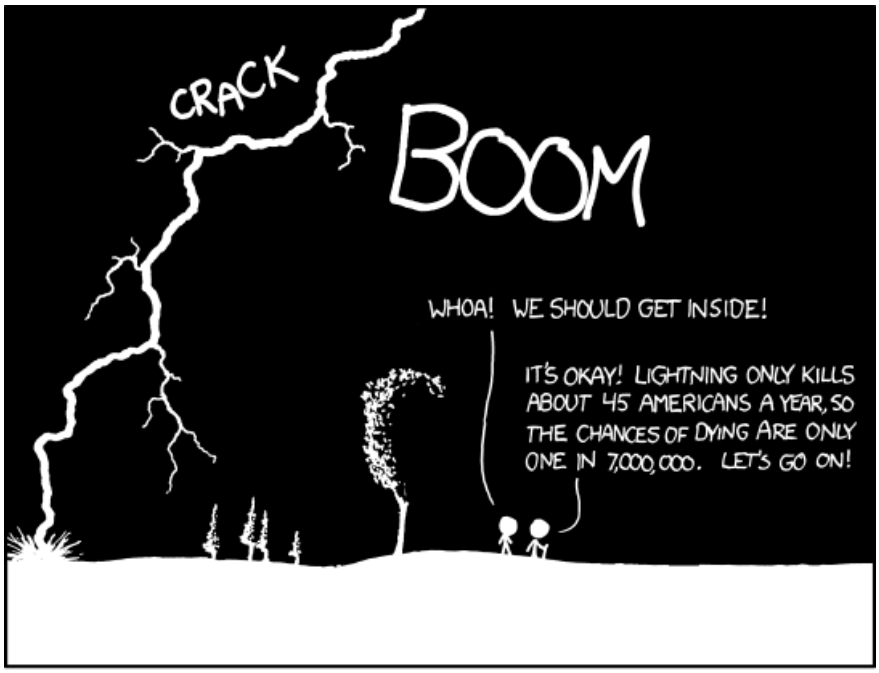
Just before exploring NB in details, let's understand few *basic concepts* first:

1. **Conditional Probability**

- This gives us the chance that something will happen given that something else has already happened.

Let's say, there is an outcome 'B' and some evidence 'A' of that outcome. From the way these probabilities are defined: The probability of having **both** the outcome 'B' and the evidence 'A' is:



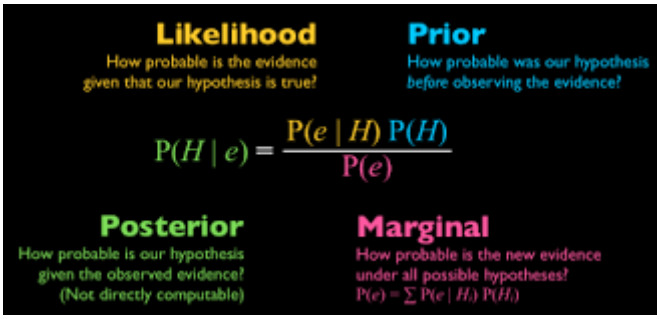


Damn.

2. **Bayes Rule**

- Conceptually, this is the way to go from ***P(Evidence|known Outcome)*** to ***P(Outcome|known Evidence).***

Often, we know how frequently some particular evidence is observed, given a known outcome. We have to use this known fact to compute the reverse, i.e. to compute the chance of that outcome happening given the evidence.



Eg. Consider a population where a disease D has broken out. The municipality in order to test the disease uses a machine which gives a positive output with some probability given the person has disease.

Probability of disease D given test-positive =
*[P(Test is +ve | disease) * P (disease)] / P (+ve test, with or without disease)*

. . .

Is it Mathematical? You bet.

So far we have talked about only a single piece of evidence. However, in real life situations there are multiple pieces of evidence that confirm the occurrence or nonoccurrence of an event.

Mathematics tends to get complicated as these are often correlated to each other. Quite intuitively, one such approach is to 'uncouple' multiple pieces of evidence, and treat each piece of evidence as independent. Hence, the name!

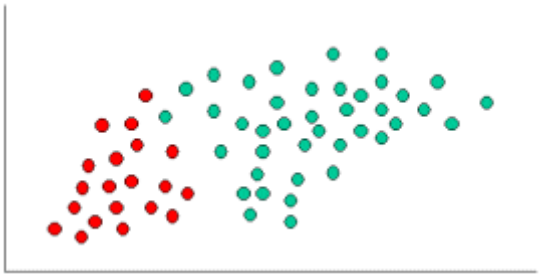
The mathematical interpretation of independence of features is illustrated by the fact that the class conditional probabilities can be computed as a product of individual probabilities:

$$P\left(\mathbf{x}|\omega_j\right) = P\left(x_1 \mid \omega_j\right) \cdot P\left(x_2 \mid \omega_j\right) \cdot \ldots \cdot P\left(x_d \mid \omega_j\right) = \prod_{k=1}^{d} P\left(x_k \mid \omega_j\right)$$

Here, *'d'* evidences were observed for the occurrence of the event *'wj'*. The ***naive assumption of independence*** of variables allowed us to write the probabilities as the product of individual class-conditional probabilities

. . .

Example. Explained.



Given set.

As indicated, the objects can be classified as either `GREEN` **or** `RED` . Our task is to classify new cases as they arrive, i.e., decide to which class label they belong, based on the currently existing objects.

Since there are twice as many `GREEN` objects as `RED` , it is reasonable to believe that a new case (which hasn't been observed yet) is twice as likely to have membership `GREEN` rather than `RED` .
In the Bayesian analysis, this belief is known as the ***prior probability***. Prior probabilities are based on previous experience. In this case, the percentage of `GREEN` and `RED` objects, are often used to predict outcomes before they actually happen.

Since, there are a total of `60` objects, `40` of which are `GREEN` and 20 `RED` , our prior probabilities for class membership are:

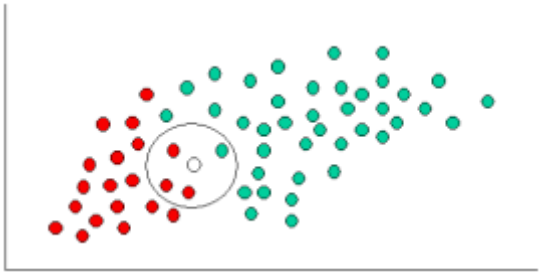
Prior Probability of `GREEN` : `number of GREEN objects / total number of objects` = `40 / 60`

Prior Probability of `RED` : `number of RED objects / total number of objects` = `20 / 60`

Having formulated our prior probability, we are now ready to classify a new object (`WHITE` circle in the diagram below).

> *Since the objects are well clustered, it is reasonable to assume that the more* `GREEN` *(or* `RED` *) objects in the vicinity of X, the more likely that the new cases belong to that particular color.*

To measure this likelihood, we draw a circle around X which encompasses a number (to be chosen apriori) of points irrespective of their class labels. Then, we calculate the number of points in the circle belonging to each class label:



$$Likelihood\ of\ X\ given\ GREEN \propto \frac{Number\ of\ GREEN\ in\ the\ vicinity\ of\ X}{Total\ number\ of\ GREEN\ cases}$$

$$Likelihood\ of\ X\ given\ RED \propto \frac{Number\ of\ RED\ in\ the\ vicinity\ of\ X}{Total\ number\ of\ RED\ cases}$$

From the illustration above, it is clear that Likelihood of `X` given `GREEN` is smaller than Likelihood of `X` given `RED` , since the circle encompasses `1GREEN` object and `3RED` ones. Thus:

$$Probability\ of\ \ X\ given\ GREEN \propto \frac{1}{40}$$

$$Probability\ of\ \ X\ given\ RED \propto \frac{3}{20}$$

Although the prior probabilities indicate that `X` may belong to `GREEN` (given that there are twice as many `GREEN` compared to `RED`) the likelihood indicates otherwise; that the class membership of `X` is `RED` (given that there are more `RED` objects in the vicinity of `X` than `GREEN`).

> *In the Bayesian analysis, the final classification is produced by combining both sources of information,*

> *i.e., the prior and the likelihood, to form a posterior probability using the so-called Bayes' rule.*

$$\text{Posterior probability of } X \text{ being GREEN} \propto$$
$$\text{Prior probability of GREEN} \times \text{Likelihood of } X \text{ given GREEN}$$
$$= \frac{4}{6} \times \frac{1}{40} = \frac{1}{60}$$
$$\text{Posterior probability of } X \text{ being RED} \propto$$
$$\text{Prior probability of RED} \times \text{Likelihood of } X \text{ given RED}$$
$$= \frac{2}{6} \times \frac{3}{20} = \frac{1}{20}$$

> *Finally, we classify X as* `RED` *since its class membership achieves the largest posterior probability.*

. . .

Is it really that good?

One of the basic ones, this algorithm is most researched upon. Let's see if people actually use it in 2017!

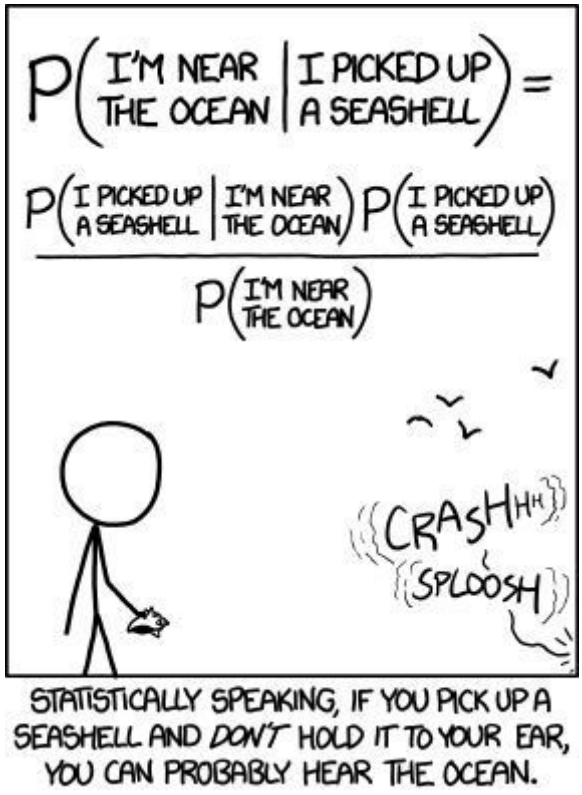
> ***Pros***

- **Easy** and **fast** to predict class of test data set. Also, performs well in *multi-class prediction*.
- When **assumption of independence** holds, a Naive Bayes classifier performs better compare to other models like logistic regression and you need *less training data*.
- It perform well in case of **categorical input variables** compared to numerical variable(s). For numerical variable, normal distribution is assumed (bell curve, which is a strong assumption).

> ***Cons***

- ***Zero Frequency:*** If categorical variable has a category (in test data set), which was not observed in training data set, then model will assign a zero probability and will be unable to make a prediction. To solve this, we can use the smoothing technique. One of the simplest smoothing techniques is called Laplace estimation.

- **Bad estimator:** Probability outputs from predict_proba are not to be taken too seriously.
- **Assumption of independent predictors:** In real life, it is almost impossible that we get a set of predictors which are completely independent.



Yes, we are vella.

. . .

Implementation in Python

https://github.com/meetvora/mlp-classifier/blob/master/models/naiveScratch.py

^This has been implemented from scratch. You can obviously use SKL.

References

1. AV's blog
2. Stack OverFlow - Naive Bayes Classification
3. Sklearn module

Footnotes

Used in spam filtering or document classification, it is widely known to all data scientists. You are ***one of them*** now!

Coming tomorrow, 12A12D shall cover Regression techniques. Be ready.

Thanks for reading. :)
And, ❤if this was a good read. Enjoy!

Co-Authors: Abhinaba Bala and Palash Jain
Editor: Akhil Gupta