5-Minute Machine Learning

Bayes Theorem and Naive Bayes

Andre Violante Jan 11 · 5 min read ★

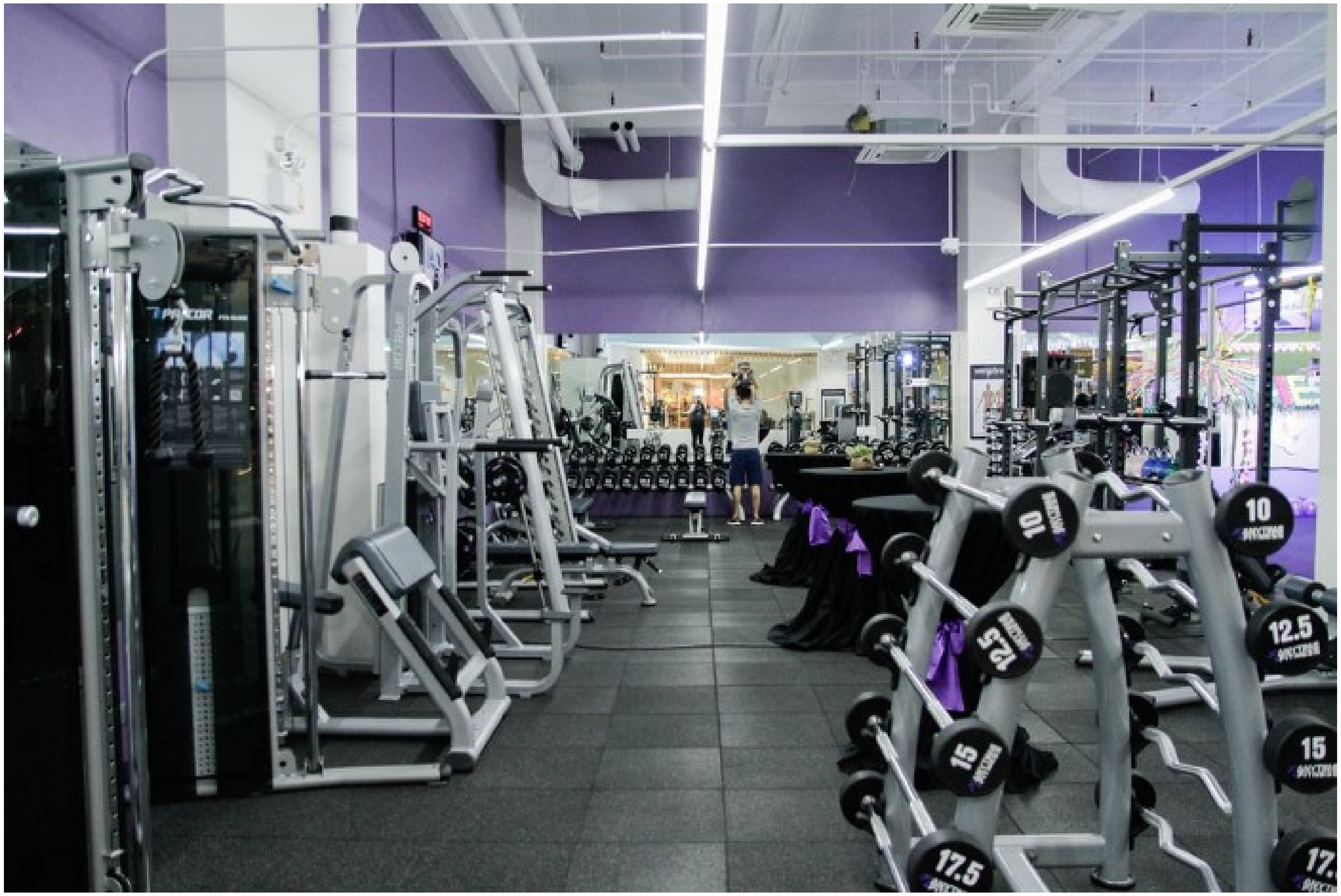

Image from Unsplash.com by @markbertulfo

Naive Bayes is a set of simple and efficient machine learning algorithms for solving a variety of classification and regression problems. If you haven't been in a stats class for a while or seeing the word "bayesian" makes you uneasy then this is may be a good 5-minute introduction. I'm going to give an explanation of Bayes theorem and then Naive Bayes within 5 minutes using a fitness gym new years resolution example. I'll also include some simple python code using Scikit-learn in my GitHub. Let's get started!

Quick Intro to Bayes Theorem

In order to explain Naive Bayes we need to first explain Bayes theorem. The foundation of Bayes theorem is conditional probability (figure 1). In fact, Bayes theorem (figure 1) is just an alternate or reverse way to calculate conditional probability. When the joint probability, ***P(A∩B)***, is hard to calculate or if the inverse or Bayes probability, ***P(B|A)***, is easier to calculate then Bayes theorem can be applied.

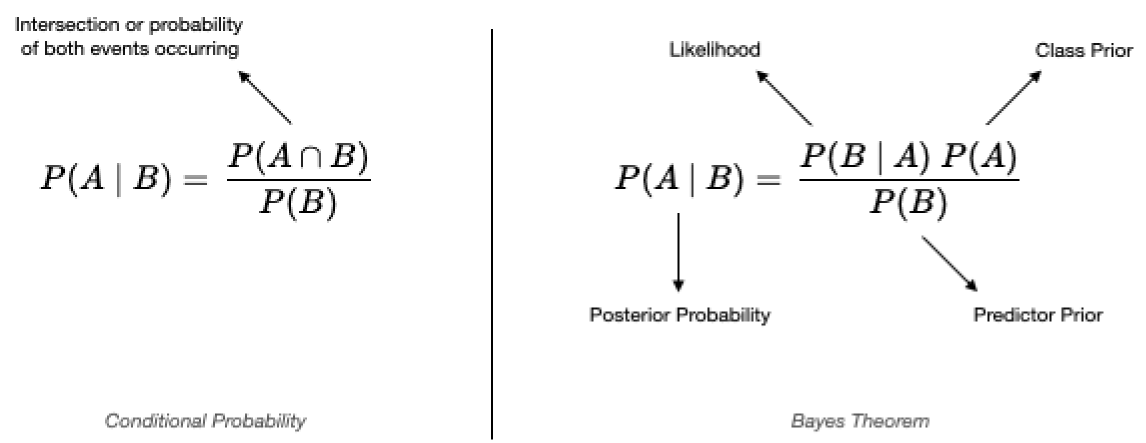

Figure 1 — Conditional probability and Bayes theorem

Let's quickly define some of the lingo in Bayes theorem:

- ***Class prior or prior probability:*** probability of event A occurring before knowing anything about event B.
- ***Predictor prior or evidence:*** same as class prior but for event B.
- ***Posterior probability:*** probability of event A after learning about event B.
- ***Likelihood:*** reverse of the posterior probability.

Bayes Theorem vs Naive Bayes: Whats the Difference?

What does all this have to do with Naive Bayes? Well, you need to know that the distinction between Bayes theorem and Naive Bayes is that Naive Bayes assumes conditional independence where Bayes theorem does not. This means the relationship between all input features are independent. Maybe not a great assumption, but this is is why the algorithm is called "naive". This is also one reason the the algorithm is very fast. Even though the algorithm is "naive" it can still outperform complex models so don't let the name dissuade you. I'll show notation difference between Bayes theorem and Naive Bayes below. Let's first work through yet another Bayes theorem example for our friends at Globo Gym.

Steps to Apply Bayes Theorem

Here's a simple example that will be relevant with all the New Years resolutions. Globo Gym wants to predict if a member will attend the gym given the weather conditions ***P(attend = yes | weather)***.

Step 1- View or collect "raw" data.

We have data where each row represents member attendance to Globo Gym given the weather. So observation 3 is a member that attended the gym when it was cloudy outside.

```
  weather attended
0   sunny      yes
1   rainy       no
2   snowy       no
3  cloudy      yes
4  cloudy       no
```

Step 2 - Convert long data to a frequency table

This provides the sum of attendance by weather condition.

```
         attended
          no  yes
weather
```

```
cloudy     1    3
rainy      2    1
snowy      3    1
sunny      1    3
```

Step 3 - Row and column sums to get probabilities

```
weather probabilities
cloudy = 4/15 or 0.267
rainy  = 3/15 or 0.20
snowy  = 4/15 or 0.267
sunny  = 4/15 or 0.267

attendance probabilities
no  = 7/15 or 0.467
yes = 8/15 or 0.533
```

Looking at our **class prior probability** (probability of attendance), on average a member is 53% likely to attend the gym. Just FYI, Thats the exact business model for most gyms: hope a lot of people sign up but rarely attend. However, our question is whats the probability a member will attend the gym given the weather condition.

Step 5 - Apply probabilities from frequency table to Bayes theorem

Figure 2 shows our question put into Bayes theorem notation. Let's assign each of the probabilities in figure 2 a value from our fequency table above and then rewrite the equation so its clear.

$$P(yes \mid sunny) = \frac{P(sunny \mid yes)\, P(yes)}{P(sunny)}$$

Figure 2 - Bayes theorem for probability of attending given the weather is sunny

Likelihood*: P(sunny | yes)* = 3/8 or 0.375 (Total sunny **AND** yes divided by total yes)

Class Prior Probability*: P(yes)* = 8/15 or 0.533

Predictor Prior Probability*: P(sunny)* = 4/15 or 0.267

$$P(yes \mid sunny) = \frac{(0.375 \cdot 0.533)}{0.267} = 0.749$$

Figure 3 - Bayes theorem with values

Figure 3 shows that a random member is 75% likely to attend the gym given its sunny. Thats higher than the overall average attendance of 53%!

On the opposite spectrum, the probability of attending the gym when its snowy out is only 25% (0.125 · 0.533 / 0.267).

Since this is a binary example (attend or not attend) and ***P(yes | sunny)*** = 0.75 or 75%, then the inverse ***P(no | sunny)*** is 0.25 or 25% since probabilities have to sum to 1 or 100%.

Thats how to use Bayes theorem to find the posterior probability for classification. The Naive Bayes algorithm is similar to this which we'll show next. Just to be clear, one obvious problem with our example is that given the weather we apply the same probability to all members which doesn't make sense, but this is just a fun example. Now, let's discuss additional features and using Naive Bayes.

Multiple Features and Using Naive Bayes

In nearly all cases you'll have many features in a model. Example features for Globo Gym could be: age bins, membership type, gender, etc. Lets show how you would incorporate those features into Bayes theorem and Naive Bayes.

Figure 4 below shows Bayes theorem simplified into the Naive Bayes algorithm incorporating multiple features. In Bayes theorem you would calculate a single conditional probability given all features (top). With Naive Bayes we simplify it by calculating the conditional probability for each feature and then multiply them together. Remember, this is why it's called "naive" since all the features conditional probabilities are calculated independently of each other. The Naive Bayes algorithm is literally simplified by the help of independence and dropping the denominator. You can follow the steps above from Bayes theorem to apply these, now easy, calculations and hence the relationship between Bayes theorem and Naive Bayes!

Figure 4 — Bayes theorem simplified into Naive Bayes

Conclusion

That was a quick 5-minute intro to Bayes theorem and Naive Bayes. We used the fun example of Globo Gym predicting gym attendance using Bayes theorem. We explained the difference between Bayes theorem and Naive Bayes, showed the simplified notation, and showed why it's "naive" through the assumption of independence. There's so much more to add here, but hopefully this gives you a some understanding of Bayes theorem and the Naive Bayes algorithm. I'll add some good reading in the references below. I hope your curiosity of Naive Bayes has grown and that you'll incorporate it into your next project.

References

1. This guy is awesome! Articles by Jason Brownlee: Bayes Theorem for Machine Learning, Probability, and Develop Naive Bayes from Scratch
2. Free pdf of *Think Bayes* book here
3. GitHub repo with simple code
4. Scikit-learn documentation for Naive Bayes

Statistics Machine Learning Data Science Python