Bayes' theorem

Consider the formula for the conditional probability:

$$P(A|B) = \frac{P(A \cup B)}{P(B)} \tag{1}$$

which is equivalent to

$$P(A|B)P(B) = P(A \cup B) \tag{2}$$

which tells us that the probability of A and B is the probability of B multiplied by the probability of A given B. This makes lots of sense, but it is also notable that the left hand side doesn't look symmetric in A and B while the right hand side clearly is. Obviously this means we can write

$$P(A|B)P(B) = P(B|A)P(A) \tag{3}$$

or

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{4}$$

This formula is called the Bayes' rule and is surprisingly useful because there are lots of interesting problems where we are told $P(B|A)$ but would like to know $P(A|B)$.

Often the example given is related to testing. Lets say 5% of steaks sold as beef steak are actually made of horse and imagine we have a horsiness test which is positive 90% of the time when tested on horse and 10% of the time when tested on beef. If a piece of steak tests positive for horse, what is the chance it is horse? Let H be the event of being horse and Y the event of testing positive for horsiness. Now we know $P(H) = 0.05$ and $P(Y|H) = 0.9$; what we want is $P(H|Y)$ and this is what Bayes' rule is useful for:

$$P(H|Y) = \frac{P(Y|H)P(H)}{P(Y)} \tag{5}$$

We don't have $P(Y)$ but we can work it out:

$$P(Y) = P(Y|H)P(H) + P(Y|\bar{H})P(\bar{H}) \tag{6}$$

since $P(Y \cup H) = P(Y|H)P(H)$ and so on. Hence

$$P(Y) = 0.9 \times 0.05 + 0.1 \times 0.95 = 0.14 \tag{7}$$

Thus

$$P(H|Y) = \frac{0.9 \times 0.05}{0.14} = 0.32 \tag{8}$$

Hence, surprisingly, if a steak tests positive for horsiness it is still more likely to be beef. Basically, because there are so many more beef steaks than horse steaks, the relatively small false positive rate for beef still leads to a reasonably high chance a piece of steak that tests positive for horse is nonetheless beef.

There is a particular terminology associated with Bayes' rule; it is sometimes written:

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}} \tag{9}$$

The *posterior* is the probability estimated after the evidence is gathered, for example, the chance of horsiness after we have found the test is positive. The *likelihood* is how likely the evidence is given the event, in the example above, it is $P(Y|H)$; the *prior* is the probability estimated before the evidence is gathered, that is $P(H)$, finally *evidence* measure the probability of the evidence, $P(Y)$.

Naïve Bayes estimator

Many learning algorithms can be thought of have machines for estimating probabilities, often in the face of insufficient data to estimate the probabilities required. A common example used to illustrate this is a spam filter. Let W represent an ordered list of words that may be in a email, say:

$$W = (\text{enlargement}, \text{xxx}, \text{cheapest}, \text{pharmaceuticals}, \text{satisfied}, \text{leeds}) \tag{10}$$

and say $\mathbf{w}$ is a vector of zeros and ones indicating the presence or absense of different potential spam words in an email. Thus, an email that includes the words 'enlargment', 'xxx' and 'leeds' but not 'cheapest', 'pharmaceuticals' and 'satisfied' would be represented by

$$\mathbf{w} = (1, 1, 0, 0, 0, 1) \tag{11}$$

Now let S represent the event of an email being spam. The objective with a spam filter is to estimate $P(S|\mathbf{w})$ for every possible vector $\mathbf{w}$ and then use a cut-off to label any email with a high probability of being spam as 'spam'.

Obviously if you have a truely huge amount of data you could estimate this probability by counting:

$$P(S|(1, 1, 0, 0, 0, 1)) = \frac{\#\{\text{spam emails with the words enlargement, xxx and leeds}\}}{\#\{\text{all emails with the words enlargement, xxx and leeds}\}} \tag{12}$$

However there are $2^6 = 64$ possible $\mathbf{w}$ vectors, and of course in a real example you'd need many more than six words, thus, for anything but an unfeasibly large data set, the amount of emails with the precise combination of words represented by a given $\mathbf{w}$ will be tiny, leading to a poor estimate of the probabilities.

An alternative approach is to use Bayes's rule to get

$$P(S|\mathbf{w}) = \frac{P(\mathbf{w}|S)P(S)}{P(\mathbf{w})} \tag{13}$$

This doesn't look any better, $P(\mathbf{w}|S)$ is no easier to estimate than $P(S|\mathbf{w})$. However, in the naïve Bayes estimator it is additionally assumed that the different words are independent so that

$$\begin{aligned} P((1, 1, 0, 0, 0, 1)|S) &= P(\text{enlargement}|S)P(\text{xxx}|S)[1 - P(\text{cheapest}|S)]\times \\ &\quad [1 - P(\text{pharmaceuticals}|S)][1 - P(\text{satisfied}|S)]P(\text{leeds}|S) \end{aligned} \tag{14}$$

This is clearly inaccurate, a spam email is 'enlargement' is more likely to contain 'satisfied' than one that doesn't, that is why it is a 'naïve' classifier. The advantage though is that the individual probabilities are much easier to estimate, there will be more emails with 'leeds' in that there will be emails with the exact combination of words represented by $(1, 1, 0, 0, 0, 1)$ and so counting occurances will be much more accurate.