Counts

We've built a lot of fancy predictive models to, in part, determine which words (or parts of speech) are associated with a particular set of textual documents. What if we think about this problem directly, and just build a table showing how often a document occurs with a given label and a given term.

	Spam	Not Spam
!	178	40
no !	211	333

Counts

How strong of an associatation is there between the two? To see, let's compute the row and column counts.

	Spam	Not Spam	
!	178	40	218
no !	211	333	544
	389	373	**762**

Probabilities

Now, we will erase (for the moment) the counts and compute the probabilities associated with each category on its own.

	Spam	Not Spam		
!			218	**28.6%**
no !			544	**71.4%**
	389	373	**762**	
	51.0%	**49.0%**		

Expected Probabilities

Given just the totals, what the expected proportion of entries that should be in each cell? We can get these by multiplying the associated row and column probabilities.

	Spam	Not Spam		
!	0.510×0.286	0.490×0.286	218	**28.6%**
no !	0.510×0.714	0.490×0.714	544	**71.4%**
	389	373	**762**	
	51.0%	**49.0%**		

Expected Probabilities

Given just the totals, what the expected proportion of entries that should be in each cell? We can get these by multiplying the associated row and column probabilities.

	Spam	Not Spam		
!	14.5%	14.0%	218	**28.6%**
no !	36.4%	35.1%	544	**71.4%**
	389	373	**762**	
	51.0%	**49.0%**		

Expected Counts

Multiplying the probabilities by the number of documents (762) gives the expected counts.

	Spam	Not Spam		
!	111.2	106.8	218	**28.6%**
no !	277.5	266.6	544	**71.4%**
	389	373	**762**	
	51.0%	**49.0%**		

Measure Pe

Bear with this idea for a moment. With these proportions, we can compute the probability of observering the exact values (yes, it will be small) that we would get the observered data.

	Spam	Not Spam
!	14.5%	14.0%
no !	36.4%	35.1%

	Spam	Not Spam
!	178	40
no !	211	333

Probability(Right | Left) = Pe

Measuring Po

Similarly, we can compute the probability of observing the data given the observered proportions. This will also be very small, but higher than the other number. The big question is: how much larger?

	Spam	Not Spam
!	23.4%	5.25%
no !	27.7%	43.7%

	Spam	Not Spam
!	178	40
no !	211	333

Probability(Right | Left) = Po

G-Score

To measure the difference between these models, we compute what is called the g-score (or log-likelihood ratio). Higher values will correspond to words that are more strongly associated with a given label.

$$G = \log \left[\frac{\text{Probability(Left | Right) = Po}}{\text{Probability(Left | Right) = Pe}} \right]$$

We can compute the G score for many terms and look at those that are the largest. We can extend this to multiclass classification by computing G scores for each specific category.

Naïve Bayes

We can turn a similar idea into a predictive model. Consider the probability of observing spam given the presence of two difference words. Using Bayes' Rule, we can write this as a equation on the right:

$$\text{Prob(spam | ! and £)} \propto \text{Prob(! and £ | spam)} \times \text{Prob (spam)}$$

If we assume, naïvely, that the probabilities of "**!**" and "**£**" given spam are independent, we get the following equation:

$$\propto \text{Prob(! | spam)} \times \text{Prob(£ | spam)} \times \text{Prob (spam)}$$

Computing the Probability

We can compute an estimate of the probability by returning to our table.

	Spam	Not Spam
!	178	40
no !	211	333

Prob(! | spam) = (178) / (178 + 211)
= 0.457

Computing the Probability

And again, for £

	Spam	Not Spam
£	128	1
no £	261	372

Prob(£ | spam) = (128) / (128 + 261)
= 0.329

Naïve Bayes

The overall probability of being spam is 51% (see previous table) and so we have:

$$\text{Prob(spam } | \text{ ! and £)} \propto \text{Prob(! } | \text{ spam)} \times \text{Prob(£ } | \text{ spam)} \times \text{Prob (spam)}$$

$$= 0.457 \times 0.329 \times 0.51$$

$$= 0.0767$$

That seems too small, but it's because we also need to compute the probability the other way around.

Computing the Probability

We can compute an estimate of the probability by returning to our table.

	Spam	Not Spam
!	178	40
no !	211	333

Prob(! | not spam) = (40) / (40 + 333)
= 0.107

Computing the Probability

And again, for £

	Spam	Not Spam
£	128	1
no £	261	372

Prob(£ | no spam) = (1) / (1 + 372)
= 0.00268

Naïve Bayes

The overall probability of being spam is 49% (see previous table) and so we have:

$$\text{Prob(ham}|\ \text{! and £)} \propto \text{Prob(! | ham)} \times \text{Prob(£ | ham)} \times \text{Prob (ham)}$$

$$= 0.107 \times 0.00268 \times 0.49$$

$$= 0.0001405124$$

Naïve Bayes

And finally, we can get the actual probability of the message being spam using the Näive Bayes algorithm.

$$\text{Prob(spam | ! and £)} = \frac{0.00268}{0.00268 + 0.0001405124}$$

$$= 0.95$$

$$= 95\%$$