Matt Forbes
February 2011
CS 400 - AI Indep. Study
Learning With Naive Bayes Classification

Overview

Situation: We want to classify pieces of data in to categories, but there is no formal definition describing how this is done. Given some number of test cases, we can come up with an efficient way of classifying unknown instances. Ideally, we could use the Bayes optimal classifier algorithm, but it is computationally infeasible. Making some simplifying (usually incorrect) assumptions we can devise an "easier" way of doing this classification.

If we let x be some instance composed of conjoined attributes $a_i \ldots a_n$, and V be the set of possible values an instance can be classified as, then the probability of x being classified as v_j is:

$$P(v_j|a_1, a_2, \ldots, a_n).$$

To find the most probable v_j, we take the argmax of the above probability. With Bayes rule, we can simplify this expression and obtain:

$$P(v_j)P(a_1, a_2, \ldots, a_n|v_j)$$

Up until this point, we have not made any assumptions, and this expression would indeed give us the most probable v_j for this instance. However, coming up with these probabilities is completely hopeless.

If we were so lucky that each a_i was conditionally independent of the others, we would be in business. From rules of probability we know that $P(a, b|c) = P(a|c)P(b|c)$ when a and b are conditionally independent given c. Using this, we can express the most probable v_j as:

$$\underset{v_j \in V}{\arg\max} \quad P(v_j) \prod_{i=1}^{n} P(a_i|v_j)$$

This whole situation just got much easier computationally. $P(v_j)$ is easy; it is just $\frac{1}{|V|}$ unless we know some values are inherently more probable. Each value of $P(a_i|v_j)$ can also be calculated fairly inexpensively. Based on the training data, this would be the number of times a_i occurred in instances classified as v_j divided by the total number of these v_j instances.

The Goal