Yihong Ye
Artificial Intelligence final report
Dr.Simon

Purpose: My final project is using the Naive Bayes Algorithm to build a spam email filter model and test out the accuracy.

Bayers Theorem: states as Probability of the event B given A is equal to the probability of event A given B multiplied by the probability of A upon the probability of B

Naive Bayes Algorithm: P(A|B) = P(B|A)P(A) / P(B)
A: proposition (evidence)
P(A|B) = conditional probability of occurrence of event A given event B is true (posterior)
P(B|A) = Probability of the occurrence of event B given the event A is true (likelihood)
P(A) = Prior probability of the proposition
P(B) = Prior probability of evidence

Posterior = likelihood * proposition prior probability / evidence prior probability

Ex: let draw a card from the deck. (no jokers)

The probability of drawing a queen is 4/52 = 1/13

P(Queen | Face) = P(Face | Queen). P(Queen) / P(Face)

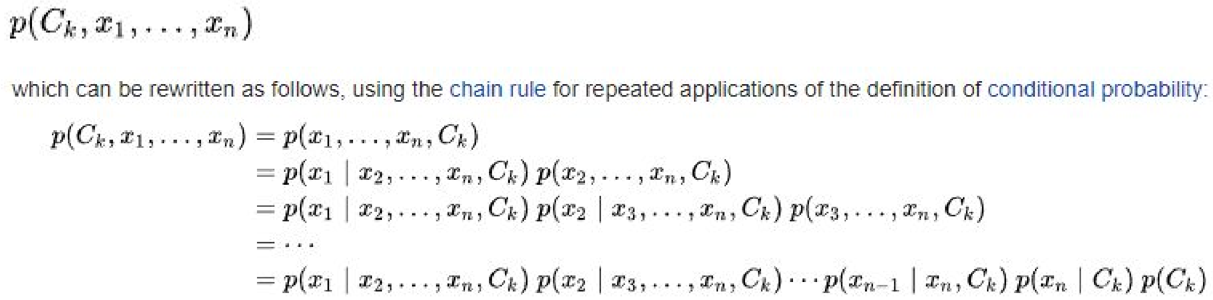
P(Queen | Face) = 1 because if we get a queen it’s a face card
P(Queen) will be 1/13
P(Face) = 3/13 (J,Q,K / 13)
P(Queen | Face) = 1/13 * 13/3 = 1 / 3
Which means if we draw a face card, there is 1 / 3 chance it will be queen

Coin Example:
To toss two coins, let A be the event that the second coin is head and B be the event that first coin is tail
Possible output: {Head Head, Head Tail, Tail Head, Tail, Tail}

P(Second coin being head given the first coin is tail)
P(A | B) = [P(B| A) * P(A)] / P(B)
= P(the first coin being tail given second coin is head) * P(the second coin being head) / P(First coin being tail)
= [(1 / 2) * (1 / 2)] / (1 / 2) = 1 / 2
So there is 50% chance that second coin will be head given the situation of first coin is tail

Naive Bayers Classifiers and Bayesian Tan are widely used for constructing models to predict accuracy such as Face Recognition, Weather Prediction, Medical Diagnosis, News classification

Bayer's Theorem for Naive Bayers Algorithm
The joint probability model:

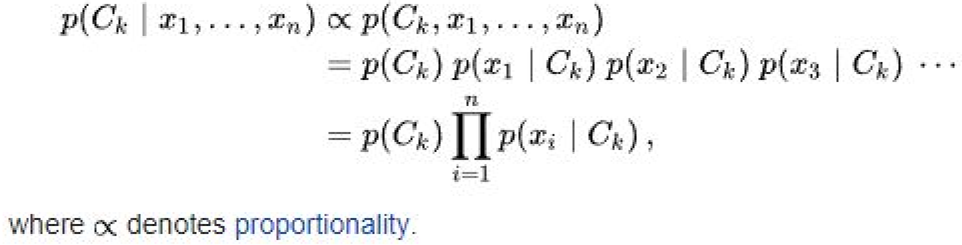


$$p(C_k, x_1, \dots, x_n)$$

which can be rewritten as follows, using the chain rule for repeated applications of the definition of conditional probability:

$$\begin{aligned} p(C_k, x_1, \dots, x_n) &= p(x_1, \dots, x_n, C_k) \\ &= p(x_1 \mid x_2, \dots, x_n, C_k)\, p(x_2, \dots, x_n, C_k) \\ &= p(x_1 \mid x_2, \dots, x_n, C_k)\, p(x_2 \mid x_3, \dots, x_n, C_k)\, p(x_3, \dots, x_n, C_k) \\ &= \cdots \\ &= p(x_1 \mid x_2, \dots, x_n, C_k)\, p(x_2 \mid x_3, \dots, x_n, C_k) \cdots p(x_{n-1} \mid x_n, C_k)\, p(x_n \mid C_k)\, p(C_k) \end{aligned}$$

And it can be express as :



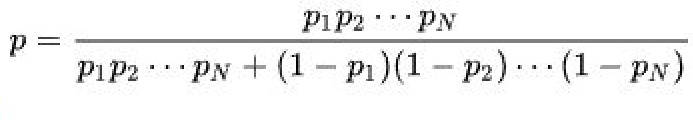
$$
\begin{aligned}
p(C_k \mid x_1, \dots, x_n) &\propto p(C_k, x_1, \dots, x_n) \\
&= p(C_k)\, p(x_1 \mid C_k)\, p(x_2 \mid C_k)\, p(x_3 \mid C_k)\, \cdots \\
&= p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)\,,
\end{aligned}
$$

where $\propto$ denotes proportionality.

Computing the probability that a message containing a given word is spam:
Given the condition that the suspected message contains the word “discount”
P(S|W) = P(W|S) * P(S) / (P(W|S) * P(S) + P(W|H) * P(H))

P(S|W): the probability that a message is a spam
P(S): the overall probability that any message is a spam
P(W|S): the probability that the word “discount” appears in spam messages
P(H): the overall probability that any message is not a spam
P(W|H): the probability that the word “discount” appear in ham messages

This is just one word but reality email is combined with lots of individual probabilities. So we need to use Bayer’s theorem which is below:



$$
p = \frac{p_1 p_2 \cdots p_N}{p_1 p_2 \cdots p_N + (1 - p_1)(1 - p_2) \cdots (1 - p_N)}
$$

P: the probability that suspects message is spam
P1: the probability P(S|W1) of spam knowing it contains a word1
P2: the probability P(S|W2) of spam knowing it contains a word2

PN: the probability P(S|Wn) of spam knowing it contains a word n

This is the formula referenced by Paul Graham

Some other heuristics:
Neutral words like “the”, “a” “some” can be ignored
Some words have a higher probability of being spam emails like “Viagra”, “good”, “credit”, “insurance”

In this project, I’m using Pandas and Scikit-Learn packages

Pandas is the data framework for manipulation
Scikit-Learn is used to build machine learning models

Step 1: read the data from the CSV file which contains two columns

Step 2: filter out the data into training data and testing data

Step 3: Add in features such as count words occurrence

Step 4: Using Scikit to build a model

Step 5: output accuracy of the model

Advantage of Naive Bayes Classifier:

1. Very simple and easy to implement
2. Needs less training data
3. Handles both continuous and discrete data
4. High scalable with number of predictors and data points
5. It can be used for real-time predictions
6. Not sensitive to irrelevant features

Disadvantage of Naive Bayes Classifier:

1. It makes a very strong assumption on the shape of your data distribution
2. Naive Bayers considers that features are independent of each other. However, in the real world, features depend on each other