Benjamin Limgoes
Assignment 5

1. The Laplace smoothing prevents there from being degenerate probabilities in our dataset. If we were to use no smoothing, then there could be probabilities that were equal to 0 or 1. If we tried to generalize from the training dataset with the 0 or 1 probabilities we could be missing vital features (or adding irrelevant ones.) For instance, suppose I were training on academic journal articles and every single paper of the "Economics Journal Article" class in my training used the word "Hessian" to talk about matrices of second order derivatives. Then I would overfit my data when I went to test it; there are plenty of Economics Journal articles that never use the phrase "Hessian" but are certainly Economics articles. Laplace smoothing prevents this from being a by mapping the probabilities between the empirical rule and a uniform distribution, rather than [0,1].
2. In this problem, the $P(|d_i|)$ serves as an adjustment factor since our domain is not over the documents themselves, but rather over the counts of certain words in each document type. A typical multinomial distribution is defined on the domain of probabilities of the unit of analysis; in our case though, we need the probability that a word appears in a certain document type, and the probability of looking at that document type. This is why there is there is an adjustment factor; it says the probability of that document class appearing.
3. Imagine I were trying to classify document types from a bunch of newspapers (to build on the Reuters Newswire example in the paper.) Document types could be the different types of newspaper pieces, such as op-eds, letters to the editor, feature pieces, and headlines. Many editors might place a strict cap on the word count for "Letters to the Editor" (perhaps 500 words or less), and also impose a 10 word maximum on headlines. The independence assumption between length and document type is then violated; if I were to randomly choose a document from a population and the number of words was less than 500, then the probability it is a headline or a letter to the editor is not the same as if there were more than 500 words. Or more simply:
   P("Letters to the Editor" | Words <= 500) > P("Letters to the Editor" | Words > 500).
4. In general, the Laplace smoothing function is as follows: $\frac{x_i + a}{N + (a * d_i)}$ where a is the smoothing parameter, N is the population size, x_i is the specific observation, and d is the word class. In the specific example of equation 6, the term |V| is the number of words in the vocabulary. This is the same as the d in the general Laplace smoothing function. The size of the vocabulary serves as the size of the number of items in the i-th document class. This is part of the adjustment factor in the smoothing function; since a=1, the vocabulary size helps move our data away from degenerate

probabilities.

5. The multi-variate Bernoulli performs better with smaller test vocabularies if the smaller vocabulary implies that there it is an easier classification problem. With an easier classification problem, it may be true that only a couple of words may be important in classification (such as classifying as financial news, the word “earnings” would most certainly be necessary). So if one of the “trigger” words appears, it is easy to pick out as one class or another.

   Another way to view this idea of an easy classification problem is to imagine the data are from a “narrow subject with limited vocabulary.” If each class has limited overlap between the classes' vocabularies, the multi-variate Bernoulli can quite easily assign high (but non degenerate) probabilities to certain words if they only occur in one of the classes. If the data were not from “a narrow subject with limited vocabulary” then it is advantageous to know how frequently a word was used in document class, rather than simply if it appeared.
6. If I increase the document length, then there is a greater probability of any given word occurring in the document, which would make the multi-variate Bernoulli method significantly worse. The way I would tackle this issue is to randomly sample words from each document class. If I can be assured in my preprocessing stage that there will be at least X number of words, then I can randomly sample a fixed number of words Y. Now that I've created training documents with a fixed length, I can use the multi-variate Bernoulli method, so long as I preprocess my test data similarly. If you are worried about high misclassification costs, I could bootstrap the process and repeatedly resample each test document a fixed number of times.