1. Movie review classification using Naive Bayes
   Assume that you have trained a Naive Bayes classifier for the task of sentiment classification (please refer to Chapter 4 in the J&M book). The classifier uses only bag-of-word features. Assume the following parameters for each word being part of a positive or negative movie review, and the prior probabilities are 0.4 for the positive class and 0.6 for the negative class.

	pos	neg
I	0.09	0.16
always	0.07	0.06
like	0.29	0.06
foreign	0.04	0.15
films	0.08	0.11

Question: What class will Naive Bayes assign to the sentence “I always like foreign films”? Show your work.

$$p(\text{class}|\text{I always like foreign films}) = P(\text{I always like foreign films}|class) * p(class)$$

Due to bag-of-word,
p(I always like foreign films | class) =
p(I | class) *
p(always | class) *
p(like | class) *
p(foreign | class) *
p(films | class) *

Substituting for the pos and neg classes,
p(I always like foreign films | pos) = 0.09 * 0.07 * 0.29 * 0.04 * 0.08 * $\underline{0.4}$ = 0.00000233856
p(I always like foreign films | neg) = 0.16 * 0.06 * 0.06 * 0.15 * 0.11 * $\underline{0.6}$ = 0.0000057024

In this binary classification instance,

$$y^{\wedge} = argmax(\text{p(pos|I always like foreign films)}, \text{p(neg|I always like foreign film)})$$

$$y^{\wedge} = neg$$

The predicted sentiment of the sentence is **negative** under this Naive Bayes classifier.

2.

a) Naive Bayes classifier pseudo code

i) Create a hashmap of words and their vector-index for entire vocabulary

ii) Cycle through all train files vectorizing a histogram of word occurrences for each class

iii) Turn histogram vectors into conditional probability vectors for each class

iv) Cycle through test files predicting for each review

v) Report accuracy

b) Training for the small movie review

Organizational steps taken

i) I created comedy and action folders, adding the respective training data

ii) I created the vocab file containing the 7 words in this corpus

iii) I created a file with the new document

iv) I created an output folder to store both output files

This allowed me to use the same functions from the preprocess and NB scripts

c) Testing the small movie review

i) I set the prior probabilities as 0.6 for action and 0.4 for comedy.

ii) The classifier predicted the genre was action

iii) log probability of comedy = -9.764237458311019

iv) log probability of action = -8.887384158769597

d) Big movie review

i) I followed the pseudocode in part a)

ii) The accuracy was around 81 percent

Discussion:

The model had 3199 false negatives (true positives predicted as negative) and 1533 false positives (true negatives predicted as positives). This implies that the model may have some bias towards classifying reviews as negative.

It may also suffer from an overly extensive vocabulary. Shortening the vocabulary to exclude less ambiguous words may help the model generalize better.

When I investigated the reviews marked incorrectly, I found a few recurring themes.

> i) “I usually hate Madonna **but**…” These reviews use negative language in a positive review (or vice versa) to qualify how this movie is unique. Due to the BOW, this syntactical context is lost.
>
> ii) Complementing movies by describing their content. This includes many reviews of the movie Jackass where the content is praised by calling it shocking, disgusting, etc. This also applies to sad or dystopian movies where the plot is discussed. Because the words used are negative, the classifier thinks the sentiment is negative. This problem seems much harder to solve. One possible solution is to have genre specific models.