Firstly, parsed both the files and stored the positive and negative reviews in two separate lists. Later, I generated word probabilities against positive and negative class(observation table). Then implemented the naive bayesian method with the assumption that prior probability of both classes are same. Then, evaluated the model by computing the confusion matrix and wrote a method called “accuracy” to compute the accuracy.

Initially used ‘/w/S*’ regular expression to extract words from the review but the accuracy was not great. Later, I tried different regular expressions and finally figured out split() function provided by python is performing better than all. I also stripped off comma and periods from the review before using the split function which further improved the accuracy.

After that, I applied laplace add-1 smoothing to the observation probabilities(a table which has probability of a word with respect to both positive and negative classes) and compared accuracy of both the models that is with smoothing and without smoothing, and observed that smoothing is improving the accuracy.

Whenever I encounter unknown word, I skipped it in the probability computation of the review as I don’t have the probability of that word w.r.t both positive and negative classes. Since I am skipping it for both classes probability computation, it won’t affect the model performance. Later, implemented K-fold cross-validation to measure my accuracy.