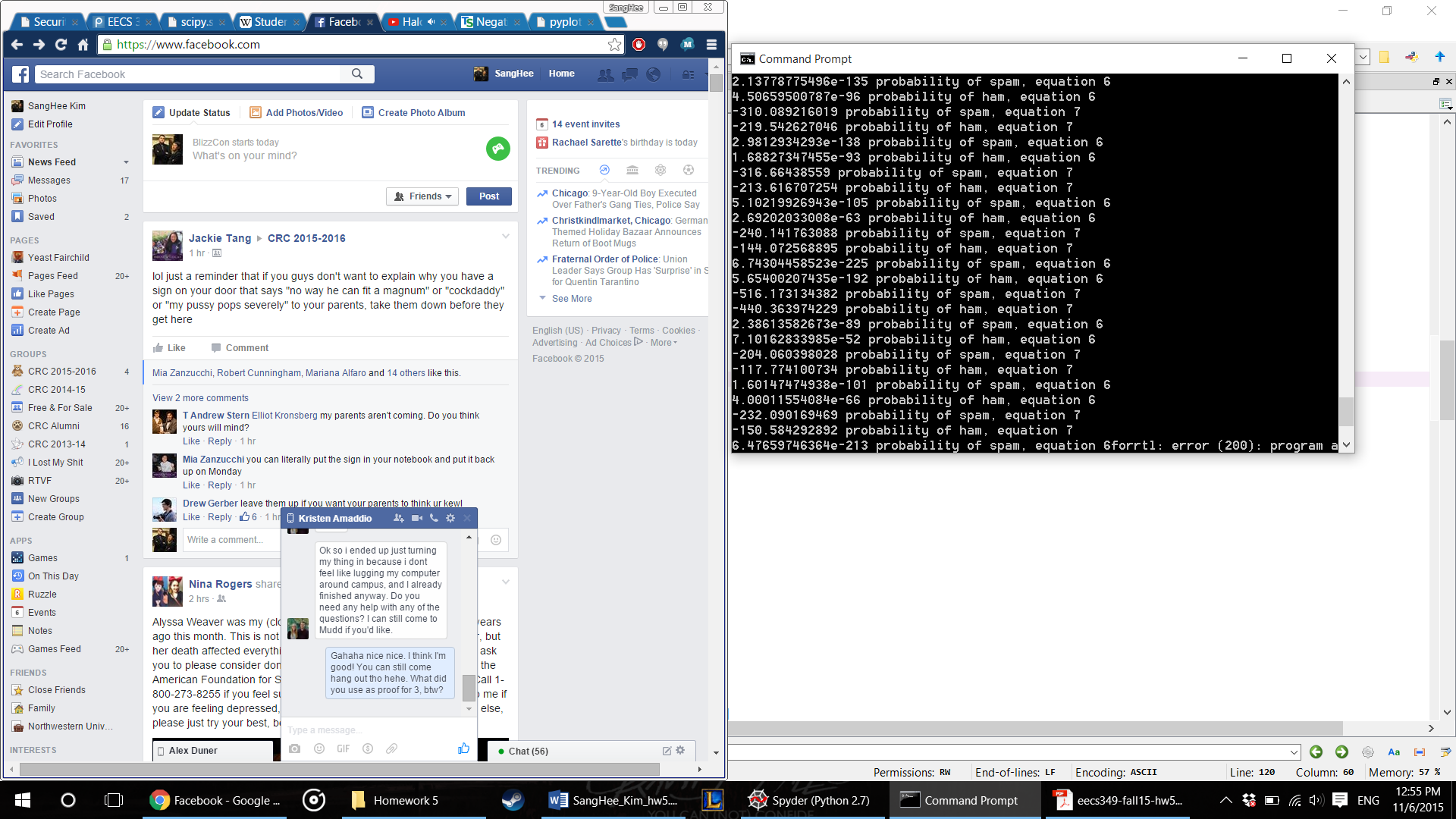
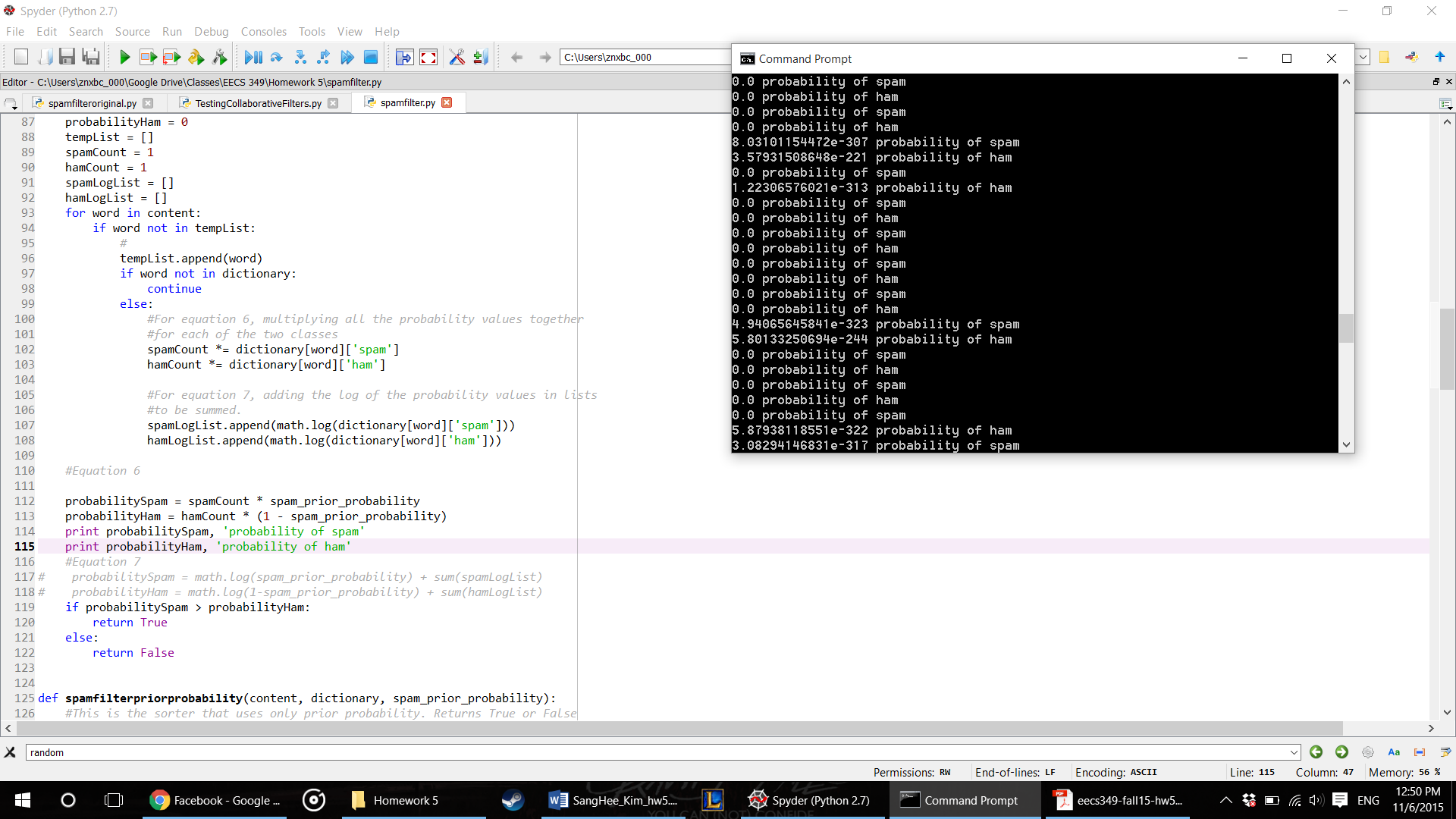
SangHee Kim  
EECS 349  
Homework 5  
11/4/15

3) Given the right amount of data, i.e. small enough amount of data that will not cause an underflow, the two functions will always give the same result. The first picture shows how the relationship between the outputs of each pairs of probabilities for the two equations is always the same; if probability of ham is lower in equation 6, it will be as well for equation 7.



If there is an underflow, the computer eventually reached a point where it was not be able to register the miniscule value and automatically round it to 0. Once that happens, the resulting probability becomes much more erratic, causing considerable amount of error. This affected the probability of ham much more, since it had much larger number of emails in its training set. This would often cause the probability of ham be 0, which caused an issue with the True/False if statement that used probabilitySpam > probabilityHam.



Notice how the values are calculated as 0 after the numbers reach beyond negative exponents in 300 range.

4A)

The dictionary was built with spamasssasin’s ‘20030228\_spam’ and ‘20021010\_easy\_ham’ as they were recommended on the assignment, and also used the emails from those two same data set to test the resulting spam filter. I chose to use the same two data set for building and testing the spam filter not only because I was more comfortable in using consistent data sets, but also because I felt that there were sufficiently enough emails on both data sets (2500 on ham, 500 on spam). The difference in two numbers also allow for the spam filter to test the situation in which the prior probability will not be 50-50 and thus closer to real-life situation where there will be more ham emails than spam.

After looking through the emails, it seemed like the general method for spamming was pretty similar to those used in recent spamming; some used baiting technique, asking the users to check a link out, or pretended to be an acquaintance, etc. However, the problem lies with the encoding method itself; it does not take into consideration all the headings in the email, which takes up almost 80%, if not more, of the sample emails. This means that the classifier is classifying based on those headers, not the actual contents of the email, which may also impact the real-life usage of this code.

4B) The experiment had 100 emails in each of the 100 samples. The errors were calculated for each of the samples, where I had the program filter the same 100 emails twice using prior probability and Naïve Bayes classifier by counting how many emails were misclassified and divided that by 100. An average error for the two methods were then calculated by dividing the sum of all the errors by 100. The errors were appended into two lists as they were calculated, and I then used the related T-test with the function scipy.stats.ttest\_rel to see which method worked better instead of using the cross validation. This is because the low p-value would allow for me to void the null hypothesis of the t-test (that there is no significant difference and thus the two methods are essentially related, which they are not) and use the calculated errors to compare the two methods.

4C)

The t-test for the two methods showed that their results had p-value ranging from anywhere from roughly 5.023-53 to 2.499-54, showing that they have statistically significant differences. With this, I compared the average errors of the two filtering methods, and it seemed that the Naïve Bayes Classifier performed much better than the prior probability filter; NBC had 0.0349 error, or 3.49% error, whereas prior probability method had .1624, or 16.24% error, three times higher than NBC.

The below graph shows the distribution of errors of each of the samples between the two methods. It can be seen that the prior probability method (the blue line) is consistently above the Naïve Bayes classifier (the red line) in all samples, which would mean that, now that the two have been calculated to have statistically significant differences, the Naïve Bayes Classifier can be said to have better performances.

