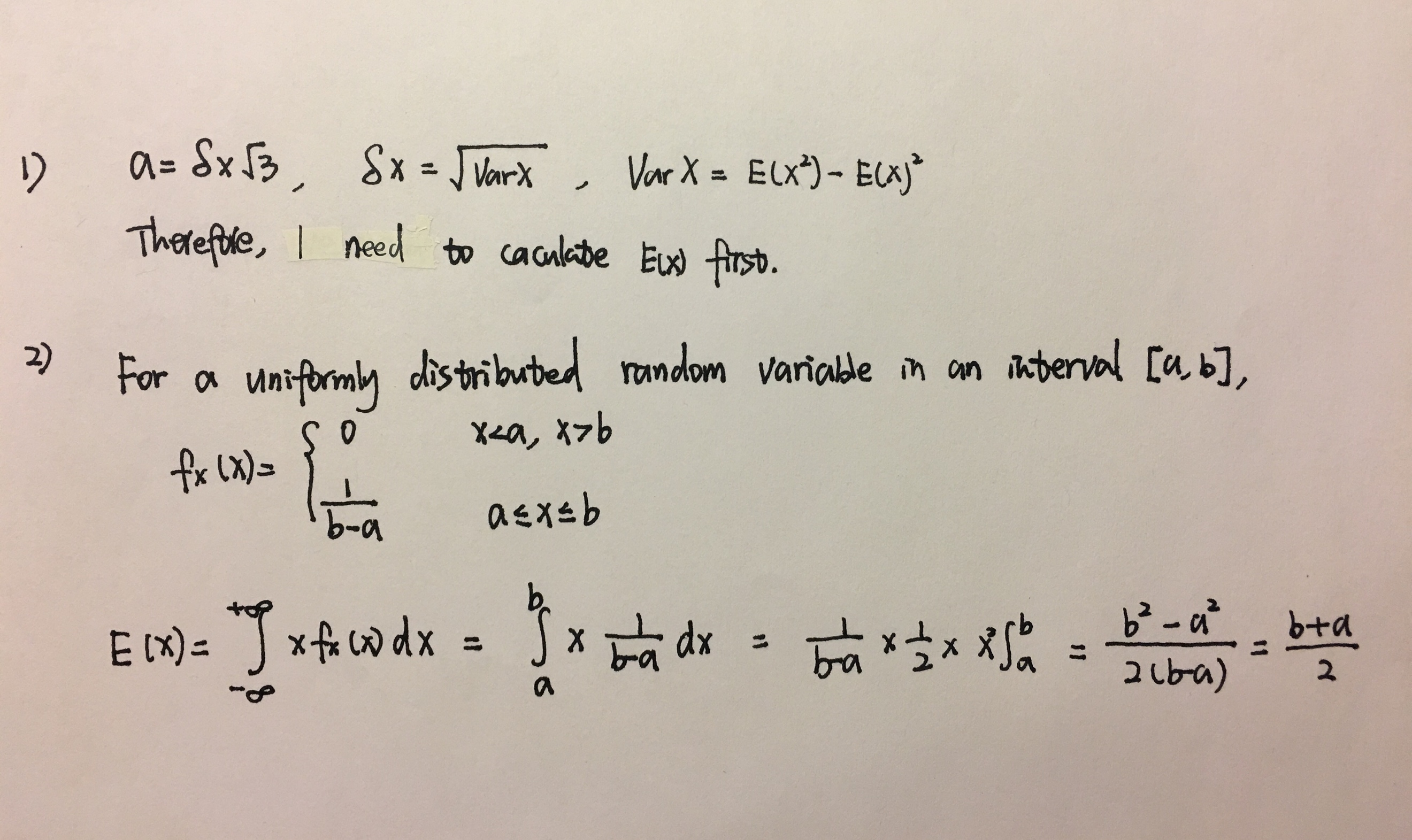
# **Report**

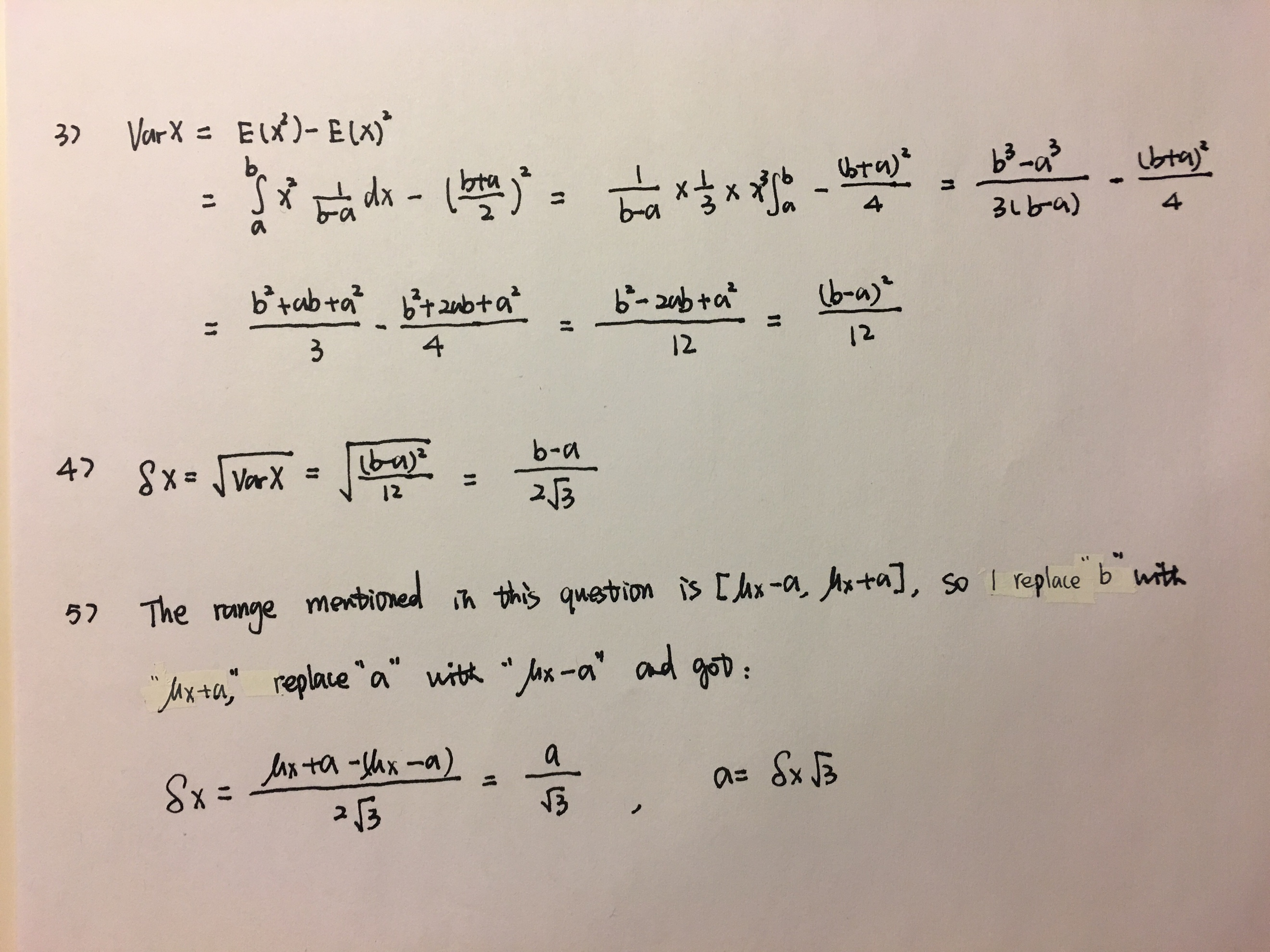
**Name: Ying Xu**

**E-mail:** [**xuying1991@hotmail.com**](mailto:xuying1991@hotmail.com)

**Last four digits of student ID: 9859**

Question1:





Question2a:

filename: hw1item2a.R

Question2b:

filename: hw1item2b.R

graphs: hw1item2b\_1.jpeg, hw1item2b\_2.jpeg, hw1item2b\_3.jpeg

Question2c:

filename: hw1item2c.R

graphs: hw1item2c\_1.jpeg, hw1item2c\_2.jpeg, hw1item2c\_3.jpeg

Question3a:

filename: hw1item3a.R

graph: hw1item3a.jpeg

I plotted a histogram with y axis as density to see the true distribution. When bandwidth is equal to 1.5, I found the estimate distribution match best with true distribution, in which the highest density appears around 4 and there is a drop around 10. If I increase the bandwidth, these features will more subtle and inaccurate. I also cut off negative values because %Hbf cannot be negative.

Question3b:

filename: hw1item3b.R

graph: hw1item3b.jpeg

I choose bandwidth is equal to 2 because it is the best match with true distribution, and it will have a very clear intersection when I plot two pdfs together. I also set range in the density function from 0 to 25, because %Hbf cannot be negative.

Question3c:

filename: hw1item3c.R

graph: hw1item3c.jpeg

According to Bayes’ classifier, the threshold should be the %Hbf where two conditional pdfs intersect. In the code, I calculated where two pdfs have the same y value, and then extract the corresponding x value from either one of those two pdfs (because it’s the same). There are several intersect values generated, but in the %Hbf range of (0, 20.7), there is only one intersection at 4.94. Therefore, the threshold is 4.94.

Questions3d:

filename: hw1item3d.R

graph: hw1item3d.jpeg

I set CBA = 5, CAB = 1 according to what mentioned in the question, therefore the cost function will be *5\*fhbf|H(x|HR)* and *1\*fhbf|H(x|HNR).* I took all values from the conditional pdf in which the patients are responders and multiply that by 5. Then I plot that as y values and the responder %Hbf as x values. After that I combined the *fhbf|H(x|HNR)* plot from question 3b (since multiply by 1 doesn’t change that function). However, I did not get an intersection, and interpret that the cost of classifying as a non-responder is always higher than classifying as responder.

Question3e:

filename: hw1item3e.R

graph: hw1item3e.jpeg

The standard deviation I choose for both conditional probabilities is 1.5. I made a histogram density plot to see the true distribution of the data, the conditional probabilities fit best when standard deviation is 1.5.

Question3f:

filename: hw1item3e.R

graph: hw1item3e.jpeg

I did the same as question 3d. However, I still did not get an intersection, and I would interpret that the cost of classifying as a non-responder is always higher than classifying as responder.

Question4:

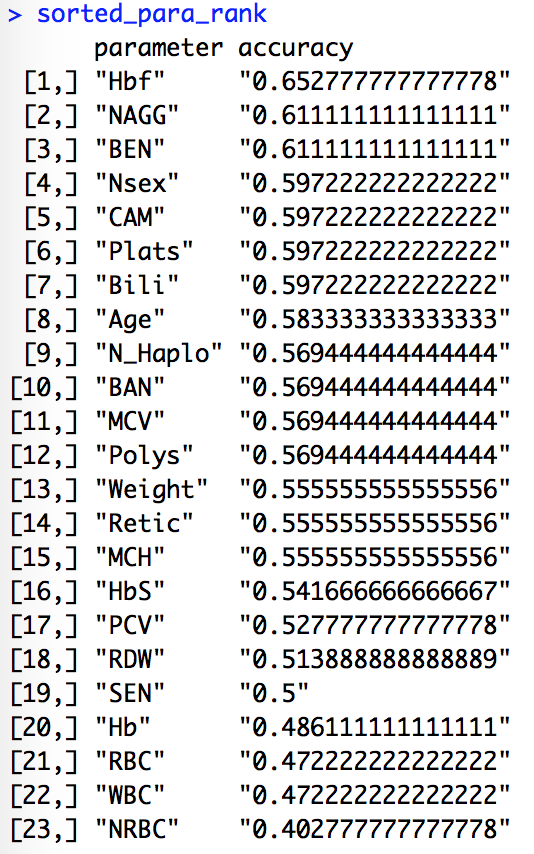
filename: hw1item4.R

graph: hw1item4\_1.jpeg ~ hw1item4\_23.jpeg

Some criteria I used in this question:

1. I used Gaussian kernel for density estimation, and used default bandwidth for each density estimation, since the default bw.nrd0 implements a rule-of-thumb for choosing the bandwidth of a Gaussian kernel density estimator.
2. If I got more than one intersections, I choose the one that is closest to median value of the parameter dataset and define that as the threshold.

Below screenshot shows the result I got. (The ranking result is stored in parameter: sorted\_para\_rank). I would choose Hbf, NAGG and BEN for the subset of parameters used for classification, since they got accuracies above 60%.



Question5:

filename: hw1item5.R

My criteria on classifying test data using KNN is if the total count of responder is greater than or equal to non-responder, then classify test data as a responder, otherwise classify test data as a non-responder. Below screenshots show the results I got, the accuracy for each parameter is different when k is different. (The accuracy ranking when k=1 is stored in parameter: sorted\_k1\_para\_rank, k=2 in sorted\_k2\_para\_rank, k=3 in sorted\_k3\_para\_rank, k=5 in sorted\_k5\_para\_rank)

The best k I would choose is k=5, it gives the highest accuracy and have the most parameters of which accuracies are higher than 60%. Additionally, the lowest accuracy is highest among k1,2 and 3. When compare k5 result with question 4, Hbf has highest accuracy in both models.

