1. **What was the estimated value of P(C) for C = 1?**

   $P(C = 1) = 0.4018006002000667$

2. **What was the estimated value of P(C) for C = 0?**

   $P(C = 0) = 0.5981993997999333$

3. **What were the estimated values for (μˆ, σˆ2) for the Gaussian corresponding to attribute capital run length longest and class 1 (Spam).**

   $\mu_{X_8|C=1} = 97.2091286307054$
   $\sigma^2_{X_8|C=1} = 36369.99111261217$

4. **What were the estimated values for (μˆ, σˆ2) for the Gaussian corresponding to attribute char freq; and Class 0.**

   $\mu_{X_1|C=0} = 0.048425863991081425$
   $\sigma^2_{X_1|C=0} = 0.08830560325706123$

5. **Which classes were predicted for the first 5 examples in the test set?**

X1	X2	X3	X4	X5	X6	X7	X8	X9	y	***y_pred***
char_freq_;	char_freq_(	char_freq_[	char_freq_!	char_freq_$	char_freq_#	capital_run_length_average	capital_run_length_longest	capital_run_length_total	label	***predicted label***
0	0	0	0	0	0	2	4	6	0	***0***
0	0	0	0.102	0	0	2.531	30	81	0	***0***
0	0.492	0	0	0	0	1.89	11	138	0	***0***
0.203	0.195	0.05	0	0.014	0	2.88	45	1080	0	***0***
0	0	0	0.874	0	0	5.114	107	179	1	***0***

6. Which classes were predicted for the last 5 examples in the test set?

X1	X2	X3	X4	X5	X6	X7	X8	X9	y	***y_pred***
char_freq_;	char_freq_(	char_freq_[	char_freq_!	char_freq_$	char_freq_#	capital_run_length_average	capital_run_length_longest	capital_run_length_total	label	***predicted label***
1.204	0	0	0	0	0	1.285	2	9	0	***0***
0	0.309	0	0.309	0	0	3.973	34	151	0	***0***
0	0.279	0	2.001	0.093	0	3.706	63	341	1	***0***
0	0.109	0	0.414	0.021	0	5.955	65	667	1	***0***
0	0	0	0	0	0	5.888	29	53	0	***0***

7. **What was the percentage error on the examples in the test file?**

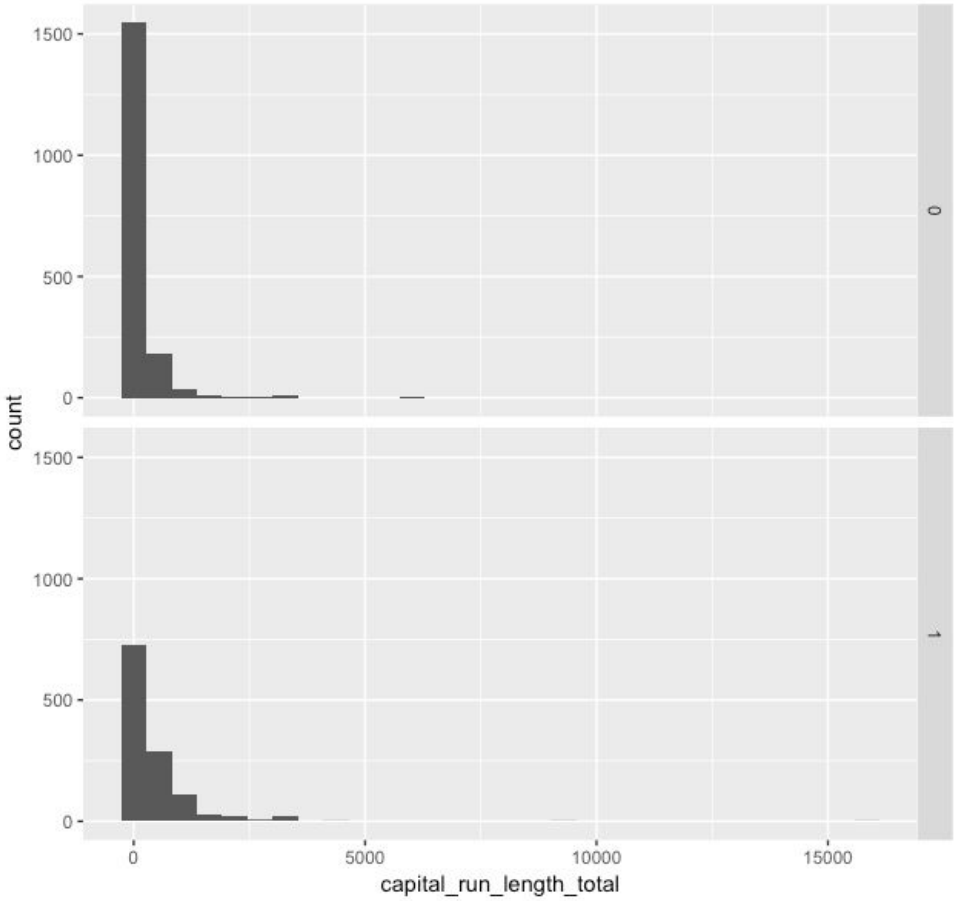
   20%

8. **Sometimes a not-very-intelligent learning algorithm can achieve high accuracy on a particular learning task simply because the task is easy. To check for this, you can compare the performance of your algorithm to the performance of some very simple algorithms. One such algorithm just predicts the majority class (the class that is most frequent in the training set). This algorithm is sometimes called Zero-R. It can achieve high accuracy in a 2-class problem if the dataset is very imbalanced (i.e., if the fraction of examples in one class is much larger than the fraction of examples in the other). What accuracy is attained is you use Zero-R instead of Gaussian Naive Bayes?**

   59%

9. **Gaussian Naive Bayes is based on two assumptions: (1) the conditional independence assumption, and (2) the assumption that the pdf for p(xj|C) are Gaussian. These assumptions are more reasonable for some datasets than for others. Do you think these assumptions are reasonable for the spam dataset you just used? Why or why not? In answering this question, you can give a common-sense argument and/or show relevant plots, graphs, or statistical information.**

- X_7 takes the average of uninterrupted sequences of capital letters, X_8 the length of the longest uninterrupted sequence of capital letters, and X_9 the total number of capital letters in the email. X_7, X_8, X_9 are variables derived from the number or order of capital letters in the documents. One may argue that if there is a random process generating capital letters in the document, quantities derived from it may not be independent of each other.
- Capital letters usually follow punctuation symbols such as "!". So there might be a dependence between X_4 and X_7, X_8, X_9.
- We could plot a histogram for each variable by class to check if they look normally distributed. Here we give an example for X_9, above it's for class 0 (not-spam) and below for class 1 (spam). We can see that the distributions don't seem to display the typical bell-shaped form of the Gaussian distribution.



- If the variables are independent and normally distributed, they should show a low or near-zero pairwise correlation.
  Calculating correlations, we have some example of correlations that aren't immaterial:
  $Corr(X_7, X_8) = 0.637482673$;
  $Corr(X_8, X_9) = 0.433454816$;
  $Corr(X_5, X_9) = 0.1665018817$.