HW4 – Solution/Rubric

a. **(30 points)** The data cleaning process greatly depends on who is doing the process. However, there are some necessary techniques that we saw in class:

- Turn to lower case
- Remove punctuation
- Remove stopwords
- Stem document
- Remove infrequent terms

Additionally, it is important to:

- Remove HTML
- Remove frequent code (such as “\n” or “\t”, for example).

Grading criteria:

- **12 points:** For carrying out steps discussed in class (first list: 4 points for removing infrequent terms, 2 points each for others).
- **5 points:** For addressing html code within text.
- **5 points:** For clarifying that body and title were processed separately.
- **8 points**: For general clarity of response (was each step briefly described within homework?)

Note: If process is not clearly described, check code to see if it’s there. If it is, penalize with (-5) for clarity of response and (-2) for each of the other three entries (where it applies).

b. **(40 points)**: There is total freedom on choosing which modelling techniques are to be used.

Test Accuracy using a 0.5 threshold for reference (Answers may vary with different random seed/different R version, etc.) Full credit if the code and models are correct.

Model	**Accuracy**
Logistics Regression	0.5916816
LDA	0.5875
Stepwise Regression	0.5883929
CART	0.5669643

Grading Criteria:

- **4 points:** For clearly showing the four techniques used and showing that at least 4 (and no more than 10) models were evaluated on the test set.
  Discount 2 points if some of the techniques do not apply (e.g. applying linear regression to this binary classification problem).
- **20 points (5 points for each technique):**
  For explaining the details of the training procedures (e.g. model tuning using cross-validation, selection criteria for parameter values).
- **4 points:** Summary for selection metric of different models.
- **8 points:** Summary for bootstrap results of the final model (ideally, a confidence interval). Explanation of why final model was chosen. Explanation of why final model was chosen.
- **4 points:** For general clarity of response (if after reading the student's answer once or twice, it is clear what they did).

c. **(30 points)** This question is purposefully open-ended and there are at least a few reasonable approaches that we can think of that should be awarded full credit. Here, we present one possibility.

From the training set, we see that roughly 1103/(1103+1137) ≈ 49% of the questions are useful. Therefore, the probability that the top question is useful is 49% if we follow Stack Overflow's current approach and assign questions based on their timestamp.

Now suppose that we order the questions such that the questions predicted to be useful by our model are on top and those that are predicted to not be useful are below. Then, the probability that the top question is useful is the probability that a generic question is actually useful given that our model predicted it to be useful. This can be estimated based on the confusion matrix by the following ratio: $\frac{\#\ true\ positives}{\#\ total\ true\ predictions} = \frac{\#\ TP}{\#\ TP+FP}$, which is also referred to as the precision of the model.

Now consider using a logistic regression model as the final model. Checking the confusion matrix,

```
     FALSE TRUE
0      745  392
1      530  573
```

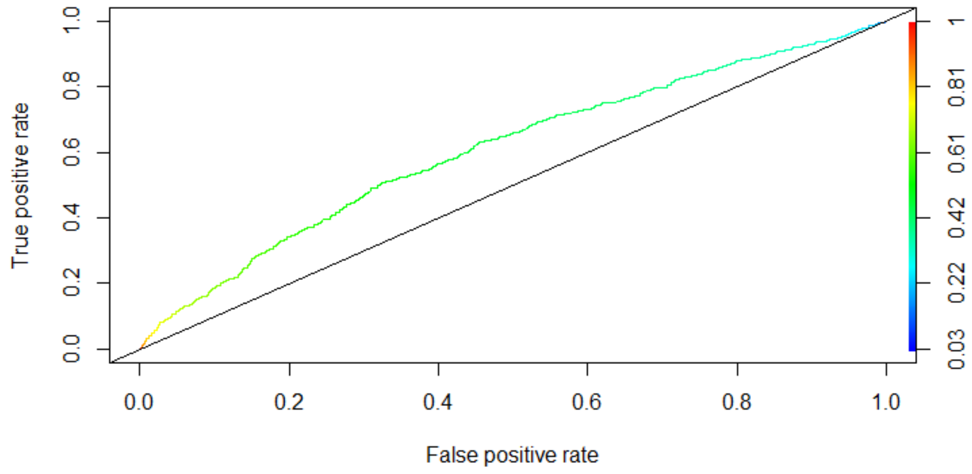
we see that setting the threshold probability $\mathrm{p} = 0.5$, yields a precision value $\approx$ 59%

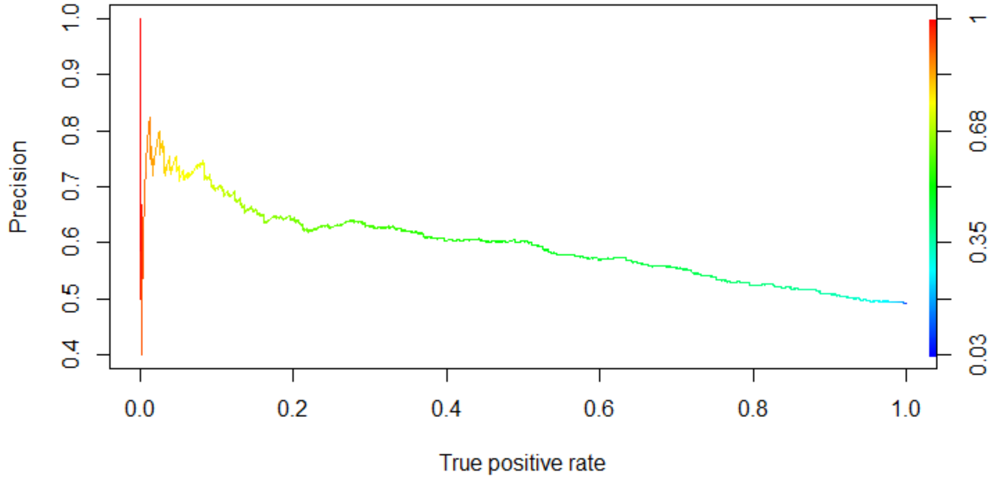
To get a better precision, we can consider changing the threshold. We have to be careful to not set the threshold too high though (close to 1), otherwise the precision becomes undefined as there will not be anything predicted as positive by the model. A reasonable way to ensure that the threshold is big enough is to make sure that, among the 15 questions on the page, at least one of them is a true positive. A back of the envelope argument tells us that, within a given list of 15 questions at any given point in time, roughly 7 questions are not useful, and 8 questions are useful. We would like to have at least 1 positive prediction out of 7 useful questions. Thus, we want to constrain the TPR $\geq \frac{1}{7} \approx 0.14$.

Typically, when TPR is very low, precision is unstable; when TPR is high, precision might decrease (see the precision-TPR plot below). So, it is a good idea to round the TPR constraint up to 0.2 (this is also a "safer" choice as it increases the expected number of true positives). Looking at the ROC (or precision-TPR plot) then guides us to a threshold value of about p = 0.6, which results in a TPR of about 0.2. Now we compute the confusion matrix

```
FALSE TRUE
0    973  164
1    827  276
```

for this threshold and we get a precision value of ≈ 62%. Thus, based on our analysis, the increase in the probability of the top question being useful is about 62% - 49% = 13%.





Grading criteria:

- **(5 points)** For general coherence of argument (after reading the answer twice, can we understand the argument?).
- **(10 points)** For including metric from model (precision in this case, or other reasonable metrics (highest predicted probability from logistic regression for example)). 0 point for accuracy.
- **(5 points)** For recognizing base case must use distribution of useful/not useful from dataset (in this case, the 49% found from the test set).
- **(10 points)** For using the result of the model on the test set to estimate the increase in the probability that the top question is useful (in this case, from 49% to 62%)