1.Naive Bayes classification by hand, without R built in functions, only math or dfm functions

Use Naïve Bayes to discover that the following email was sent by the respective parties: “immigration voter aliens help jobs” and provide estimates

republican1: immigration aliens wall country take

republican2: voter economy president jobs security

republican3: healthcare cost socialism unfair help

democrat1: immigration country diversity help security

democrat2: healthcare universal preconditions unfair help

democrat3: jobs inequality pay voter help

democrat4: abortion choice right women help court

1b. add Laplace smoothing to it on and re-estimate each party’s respective posterior probability. Give your estimate findings

For each task (3) through (6), begin with the raw version of the text

2a. Divide the reviews at the empirical median star rating and give each review a label as being positive if the star rating is greater than the median or negative if less than

2b. Create a variable “anchor” that has value “positive” if the user star rating = 5, “neutral” if less than 5 but greater than 1 ,otherwise “negative” if the user rating = 1

3. Use the dictionaries of positive and negative words “negative-words.txt” and “positive-words.txt” not any from R packages for question 3

3b. Generate a sentiment score for each review based on the number of positive words minus the number of negative words

3c. Create a vector of dichotomous variables, of equal length to the number of reviews, in which texts that have a positive sentiment score are labeled “positive,” while those with a negative score are labeled “negative”; if the sentiment score is equal to 0, score them as negative.

3d. Identify positive or negative reviews by creating a confusion matrix with the positive and negative values assigned by the sentiment on the vertical axis and the binary “true” classifications from 2(a) on the horizontal axis.

3e. Use the non-anchor texts for the following, use the predicted sentiment score to rank the reviews, where rank 1 is the most positive review and N is the most negative

Now, rank the non-anchor reviews by their star rating. Compute the sum of all of the absolute differences between the predicted rank (from the sentiment score) and the star rating rank of each review.

(b) Use the “textmodel” function in quanteda to train a smoothed Naive Bayes classifier with uniform priors, using 75% of the reviews in the training set and 25% in the test set,features in the test set should match the set of features in the training set use quanteda’s dfm match function.. Use +1 smoothing. Report the accuracy, precision, recall and F1 score of your predictions. Include the confusion matrix in your answer.

(d) Re-estimate Naive Bayes with the “docfreq” prior and +1 smoothing. Report the accuracy, precision, recall and F1 score of these new results. Include the confusion matrix in your answer.

(e) Fit the model without smoothing and a uniform prior. Report the accuracy, precision, recall and F1 score of your predictions. Include the confusion matrix in your answer.

5a) Use functions in base R and quanteda, but not the built-in wordscores function.

Create a vector of wordscores for the words that appear in the “anchor negative” and “anchor positive” . That is, you should fit a wordscores model to the anchor texts. What are the 10 lowest and 10 highest wordscores?

(5b) Apply your wordscores model to the non-anchor documents. This should generate a wordscores estimate for each document. Calculate the RankSum statistic of the reviews as scored by wordscores versus the true star rating.

6. restrict your analysis to the first 1000 reviews using the original ordering of the review data.

(c) In this step, you will train SVM models with a linear kernel. Your goal is to maximize out-of-sample accuracy by fitting models with 5-fold cross-validation.You should fit 3 models, using 20%, 50%, and 70% of the data for cross-validation. The remaining data is the validation set

Report which model has the highest accuracy for out-of-sample predictions made on the validation set.

(d) Choose the best hyperparameters from the previous question part, and fit an SVM model with those hyperparameters, but with a radial kernel.

7. For this question use the first 500 reviews in the dataset.

(a) Split the dataset into a training (75%) and a test set (25%) and construct a document feature matrix for each ,features in the test set should match the set of features in the training set.

(b) Using the randomForest package fit a random forest model to the training set using the package’s default values for ntree and mtry After fitting the model, extract the mean decrease in Gini index for the feature set and order from most important to least important.

(c) Using the fitted model, predict the sentiment values for the test set and report the confusion matrix along with accuracy, precision, recall and F1 score.

(d) Now you will do some tuning of a model parameter. The package’s default value for the argument mtry is sqrt(# of features). Estimate two more models, one for each of these values of mtry: 0.5\*sqrt(# of features) and 1.5\*sqrt(# of features). As you did above, use each of the fitted models to predict the sentiment values for the test set. Report the respective accuracy scores.