Q11

KO/MC/IKM

5/16/2021

Part a)

With a split of 70% of the spam_stats315B_train.csv for training and 30% for validation (we do not perform hyperparameter tuning over a grid of parameters however, use the validation set to obtain an estimate of the misclassification rate), we get an estimate of 4.77%. The test set misclassification rate obtained is 4.04%. Within the test set, of all the spam emails, 4.69% were misclassified, whereas, of all non-spam or "good" emails, 3.60% were misclassified.

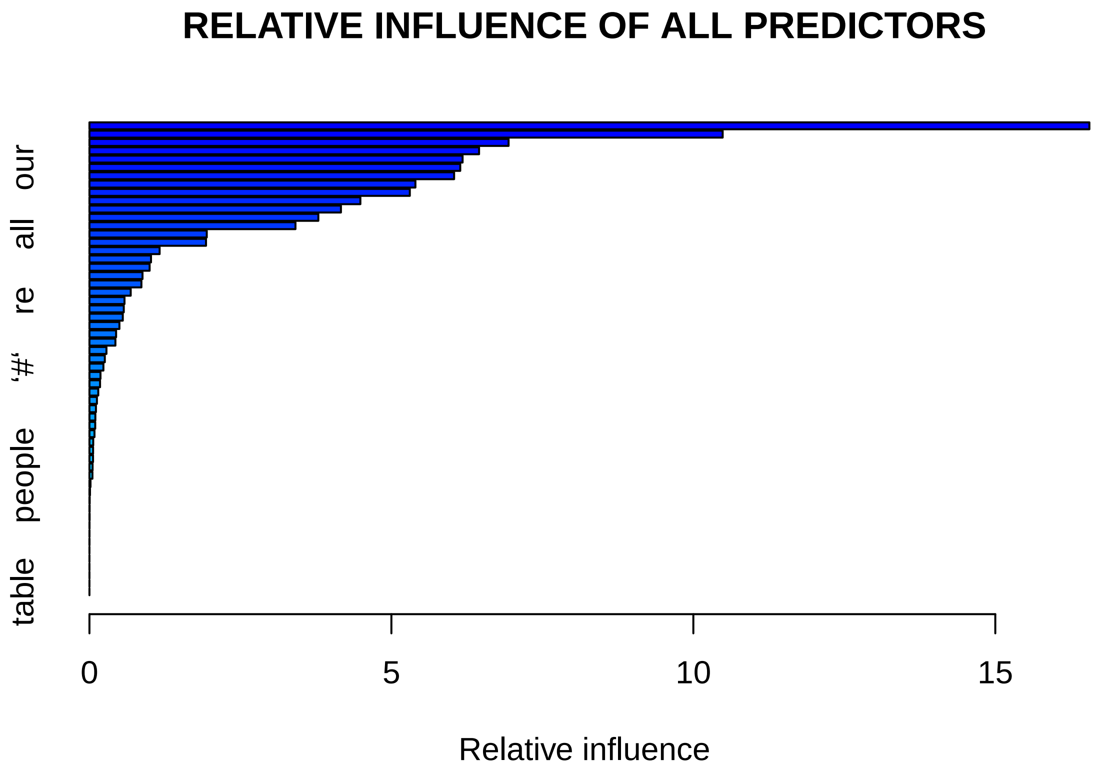
Part b)

We want to lower the non-spam misclassification rate to be less than 0.3%. Since the gbm package and function do not allow us to modify the cost matrix directly, we try a few combinations of different threshold values (other than 0.5) for classification, as well as different weights for the spam and non-spam emails to achieve the required rate. If we simply modify the threshold value, it was observed that having a threshold of 0.988 was the smallest threshold value that gave us a non-spam misclassification rate of less than 0.3%. However, with this approach, we got the overall misclassification rate to be 15.42%. The exact misclassification rate for non-spam email was 0.22%, and the misclassification rate for spam emails was 37.37%. So then we wrote a function to try a different threshold value but also a series of different weight values for spam and non-spam emails to reach the required non-spam misclassification rate while trying to keep the overall misclassification rate low.

i) With the smallest value of the weight of 400 for non-spam and 10 for spam, and a threshold value of 0.75, we got a non-spam misclassification rate of 0.22%, a spam misclassification rate of 14.47%, and an overall misclassification rate of 35.60%. We went ahead with this model.

ii) With regards to the important variables for discriminating good emails from spam emails, the five variables with the highest relative influence values are "`$`", "`!`", "remove", "hp", and "free", which make intuitive sense.

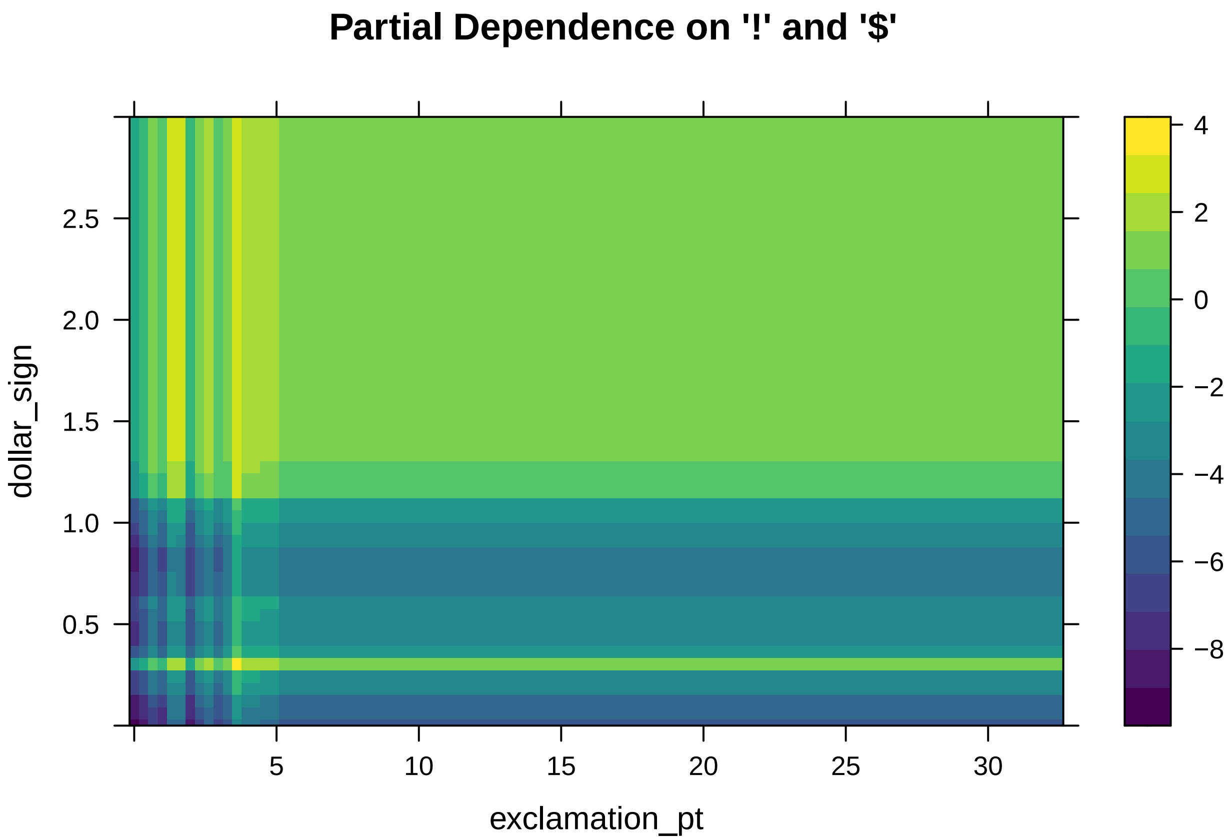
iii) To see the dependence of the response on the two most important variables "`$`" and "`!`", we created a partial dependence plot and see that there is indeed a strong interaction between "$" and "!", i.e. individually, the two variables "$" and "!" are not strong signals for a mail being spam, however, their appearance together is a strong signal for the mail being spam.




```
##                                 var      rel.inf
## remove                       remove 1.655768e+01
## exclamation_pt exclamation_pt 1.048552e+01
## open_paren               open_paren 6.940595e+00
## CAPTOT                       CAPTOT 6.451911e+00
## money                         money 6.178874e+00
## our                             our 6.139935e+00
## dollar_sign             dollar_sign 6.038729e+00
## `000`                         `000` 5.397620e+00
## free                           free 5.304909e+00
## you                             you 4.486176e+00
## CAPAVE                       CAPAVE 4.164579e+00
## your                           your 3.790491e+00
## CAPMAX                       CAPMAX 3.411465e+00
## all                             all 1.942905e+00
## credit                       credit 1.930695e+00
## `3d`                           `3d` 1.162171e+00
## email                         email 1.019393e+00
## over                           over 9.964828e-01
## mail                           mail 8.787833e-01
## business                   business 8.608026e-01
## will                           will 6.836075e-01
## re                               re 5.806721e-01
## order                         order 5.687120e-01
## font                           font 5.523432e-01
## address                     address 4.963701e-01
## receive                     receive 4.417097e-01
## internet                   internet 4.300614e-01
## semicolon                 semicolon 2.816286e-01
```

```
## make                         make 2.546599e-01
## `#`                           `#` 2.315809e-01
## technology             technology 1.829831e-01
## `1999`                     `1999` 1.756778e-01
## hp                             hp 1.463456e-01
## data                         data 1.242775e-01
## `857`                       `857` 1.076205e-01
## original                 original 9.806206e-02
## `650`                       `650` 9.587688e-02
## `415`                       `415` 8.344039e-02
## open_bracket         open_bracket 6.257926e-02
## report                     report 6.047018e-02
## edu                           edu 5.956262e-02
## project                   project 5.196318e-02
## people                     people 4.963753e-02
## george                     george 1.774393e-02
## direct                     direct 9.056100e-03
## addresses               addresses 4.599240e-03
## labs                         labs 3.303156e-03
## pm                             pm 2.945471e-03
## hpl                           hpl 1.192751e-03
## meeting                   meeting 9.274366e-04
## parts                       parts 3.540095e-04
## lab                           lab 2.059426e-04
## conference             conference 1.159315e-04
## telnet                     telnet 0.000000e+00
## `85`                         `85` 0.000000e+00
## cs                             cs 0.000000e+00
## table                       table 0.000000e+00
```

```
## gbm(formula = type ~ ., distribution = "bernoulli", data = spam_train_df,
##     weights = weights, n.trees = 2500, interaction.depth = 4,
##     shrinkage = 0.05, bag.fraction = 0.5, train.fraction = 0.8,
##     cv.folds = 5, verbose = F)
## A gradient boosted model with bernoulli loss function.
## 2500 iterations were performed.
## The best cross-validation iteration was 1960.
## The best test-set iteration was 306.
## There were 57 predictors of which 37 had non-zero influence.
```


Partial Dependence on '!' and '$'
dollar_sign
exclamation_pt
2.5
2.0
1.5
1.0
0.5
5
10
15
20
25
30
4
2
0
−2
−4
−6
−8