Blogpost

Group 15: Ruiying Yang, Lukas Zeiz

28 Juni 2021

Description of the Dataset:

- Our topic is spam email data and our task is to predict whether an email is a spam or not. The data consist of 4601 email items, of which 1813 items were identified as spam. There are 6 prediators, which are the following:
  - The total length of words in capitals (crl.tot)
  - The number of occurrences of the $ symbol (dollar)
  - The number of occurrences of the ! symbol (bang)
  - The number of occurrences of the word 'money' (money)
  - The number of occurrences of the string '000' (n000)
  - The number of occurrences of the word 'make' (make)
  - 1 target variable, which is a factor with levels n not spam, y spam (yesno).

Problems in the Dataset:

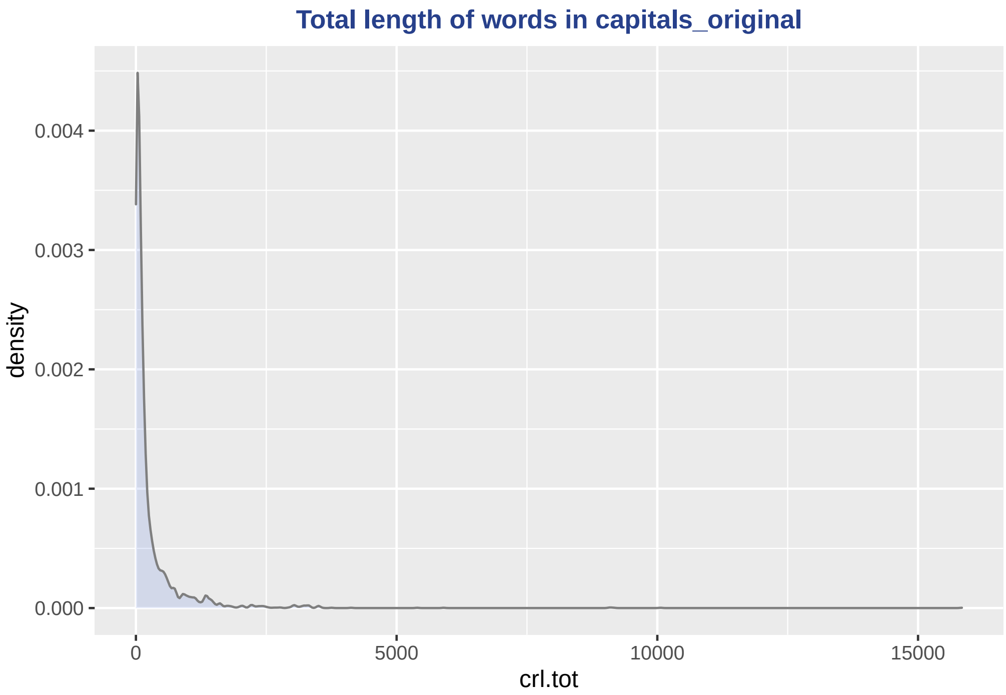
0 values

First of all, we noticed that there are many 0 values in the dataset, we have considered, if we treat them as missing values or real values. Because there are just too much 0s, for example in the column "dollar", there are more than 2 times 0 values than other values. Considering the fact that, in many emails there is just no dollar symbol, we decide to take them as real values and do not handle them.

Missing values, outliers, skewed distribution and new features (continuous variables)

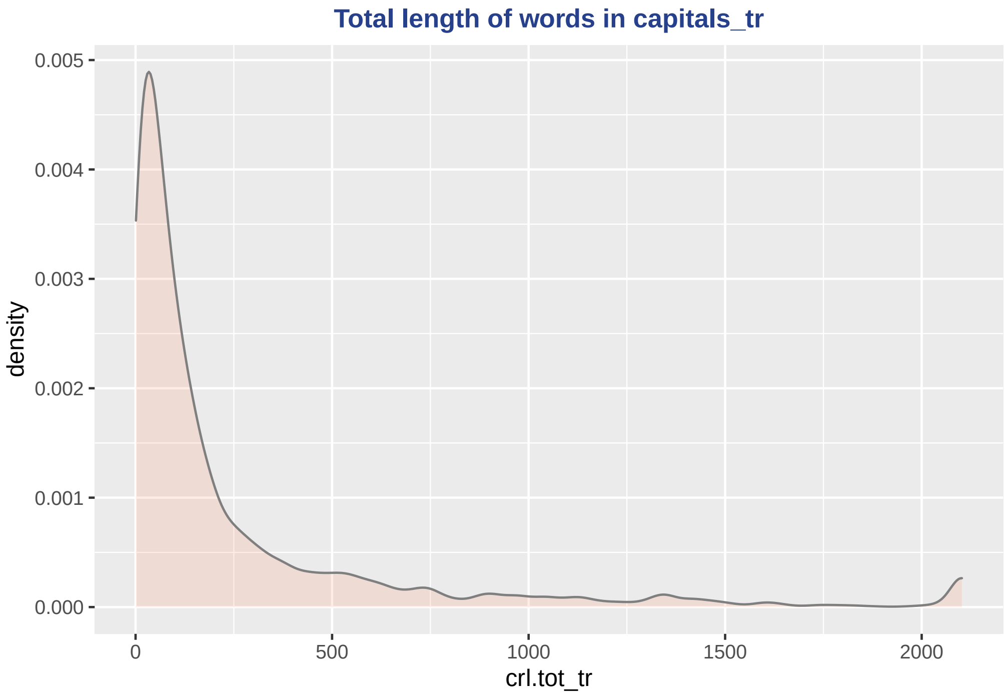
Missing values and outliers

There is no missing value in our dataset, either explicit nor implicit. There are many outliers. Take the column "crl.tot" as an example, we can see from the follwing plot that, the plot has a really long tail and most of the values are between 0-500.



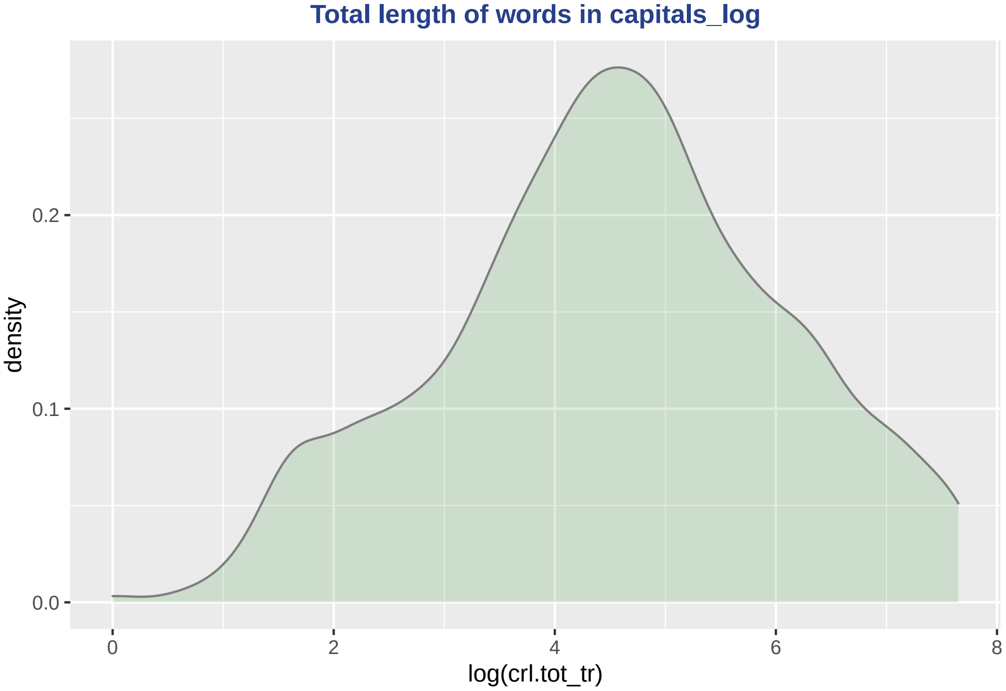
Treating outliers with Zscore

so we used Zscore and created new column "crl.tot_tr" so that we can better deal with the realtionship between different predicators and the target variable. After getting rid of the outliers, we got the follwing plot:



Handling with skewed distribution

From the plot above, we know that outliers are removed and we also kept the intactness the original data. To better finish the modelling tasks, we changed the skeness and then we got the follwing plot:



Treating other continuous variables

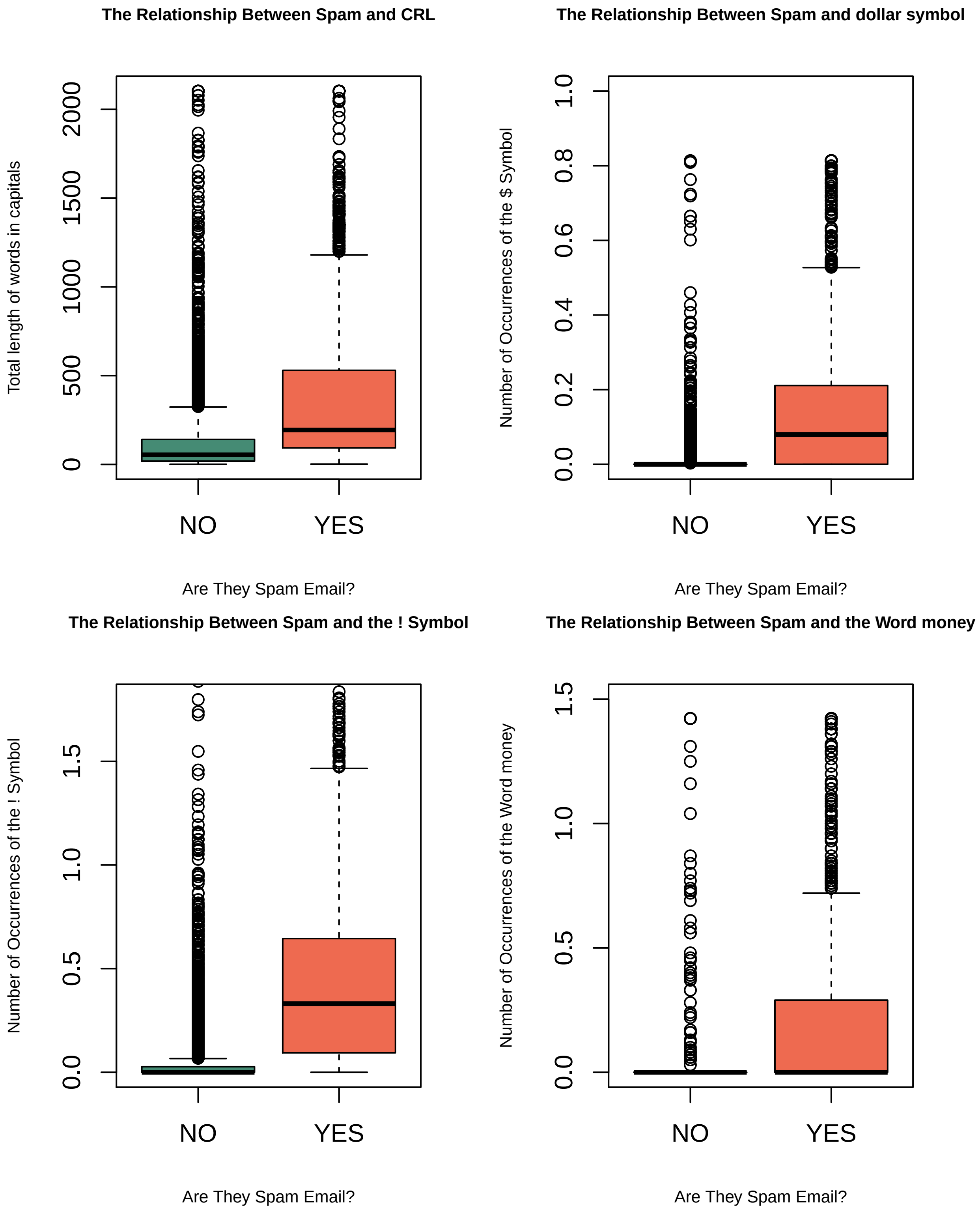
We used the same methods for other variables and created for each variable a truncated variable and a log variable.

Treating categoriacal variable

There is only one categorical varaibale in our dataset which is yesno. There is no sparse classes in this variable, so we did not treat it.

The most valuable insights from EDA

After plotting the relationships between each predicator and the target, it can be easily notice that, generally, the less symbols there are, the more likely that the email is not a spam email. For example, from the plot "The relationship betweeen spam and crl", we can see that, if the eamil is not a spam email, the total length of words in capitals is approximately in the range of 0-400, the average length is about 100. Compared to nonspam eamils, the range of the total length of capitals words in spam emails varies between 0-750. The avarage value is almost 250. It is the same for the reltionships between other predicators and the target.

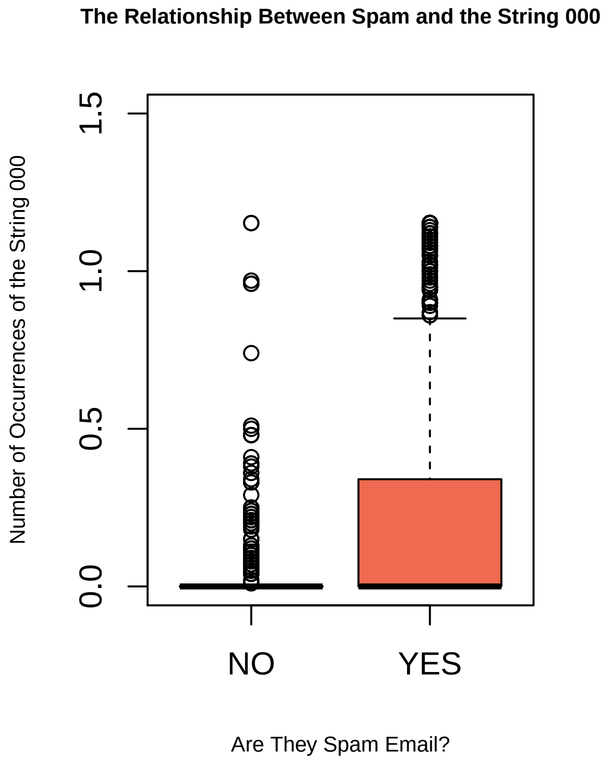


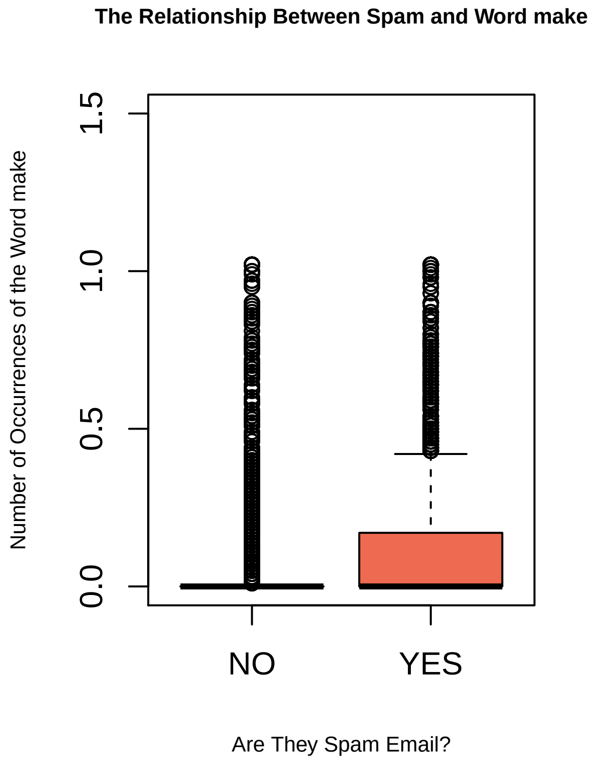
The Relationship Between Spam and CRL

The Relationship Between Spam and dollar symbol

The Relationship Between Spam and the ! Symbol

The Relationship Between Spam and the Word money





Based on this observation, we can make the conclusion that, if there exist a lot of symbols like dollar, bang, 000 or the word ‘money’, ‘make’ in an email or there are a lot of capital words in an email, then this email is very likely a spam email.

Training Models

Method

Because our models have to classify whether an email is spam or not Logistic Regression and Decision Trees are appropriate methods. In the first step we thought about the right predictor sets and decided on four different. For all we used the log variables as predictors.

Predictor Set 1 = {money, make, crl.tot} - A set based on occurrence of the different letter sequences.

Predictor Set 2 = {dollar, bang} - A set based on the occurrence of the symbols $ and !.

Predictor Set 3 = {dollar, money, n000} - A set with variables that are directly related to the topic of money.

Predictor Set 4 = All variables - A set with all variables because all together are high indicators of spam in our opinion.

Before we trained the models we saved 30% of the data as a test set to avoid overfitting.

We trained four different Logistic Regression Models with one predictor set each. Also, we trained two Ridge Regression Models with predictor set 1 and 2 and two LASSO Regression Models with Predictor Set 3 and 4. Last but not least we created a Decision Tree with all variables as predictors.

Best Model

The model with the highest Accuracy and the lowest Brier Score is the LASSO Regression Model with all variables as predictors. The R Code for this looks like this:

```
set.seed(777)
#set feature set
features.all_features <- c("crl.tot_ln" , "dollar_ln" ,
                           "bang_ln" , "money_ln" , "n000_ln" , "make_ln")

#set train and test data
X.train.all_features <- model.matrix( ~ . -1,
                                data = spam.train[,features.all_features])
X.test.all_features <- model.matrix( ~ . -1,
                                data = spam.test[, features.all_features])

#fit LASSO Regression Model
all_features.lasso <- glmnet(X.train.all_features,
                             y.train, alpha= 1,
                             family = "binomial")

#selecting the optimal lambda with the lowest misclassification error
all_features.lasso_cv <- cv.glmnet(X.train.all_features,
                                   y.train, alpha = 1,
                                   type.measure = "class",
                       lambda = 10^seq(-5, 1, length.out = 100),
                       family="binomial", nfolds = 10)

#make predictions
pred.all_features.lasso <- as.vector(predict(all_features.lasso,
                                             newx = X.test.all_features,
                                             type = "response",
                                        s = all_features.lasso_cv$lambda.min))
```

As the best lambda with the lowest misclassification error we discovered 0.0027.

After it was trained on the training data the model reached an Accuracy of 0.867, a classification Error of 0.133 and a Brier Score of 0.332

Evaluation of prediction task

An Accuracy of 0.867 is good but not outstanding. We still have 186 wrong classifications and declare more emails as spam than the other way around. On the one hand, this can be good because more dangerous spam mails can be intercepted. On the other hand, important e-mails could also be mistakenly viewed as spam and then overlooked.