PhishingAttack

Abhinav Ram Bhatta, Prajwal Prashanth, Anviksha Gupta

2022-09-28

```
library("rpart")
library("ggplot2")
library("tidyverse")
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ─
─
## ✔ tibble  3.1.8      ✔ dplyr   1.0.10
## ✔ tidyr   1.2.0      ✔ stringr 1.4.1
## ✔ readr   2.1.2      ✔ forcats 0.5.2
## ✔ purrr   0.3.4
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ─
─
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library("psych")
```

```
##
## Attaching package: 'psych'
##
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library("corrplot")
```

```
## corrplot 0.92 loaded
```

```
library("RColorBrewer")
```

Load Dataset

```
q5=read.csv("/Users/abhinavram/Documents/IDS572 Data Mining/Assignment 1/Q5.csv")
```

Summary of dataset

```
summary(q5)
```

```
##         id         having_IP_Address   URL_Length      Shortining_Service
##  Min.   :    1   Min.   :-1.0000   Min.   :-1.0000   Min.   :-1.0000
##  1st Qu.: 2764   1st Qu.:-1.0000   1st Qu.:-1.0000   1st Qu.: 1.0000
##  Median : 5528   Median : 1.0000   Median :-1.0000   Median : 1.0000
##  Mean   : 5528   Mean   : 0.3138   Mean   :-0.6332   Mean   : 0.7388
##  3rd Qu.: 8292   3rd Qu.: 1.0000   3rd Qu.:-1.0000   3rd Qu.: 1.0000
##  Max.   :11055   Max.   : 1.0000   Max.   : 1.0000   Max.   : 1.0000
##  having_At_Symbol  double_slash_redirecting Prefix_Suffix    having_Sub_Domain
##  Min.   :-1.0000   Min.   :-1.0000          Min.   :-1.000   Min.   :-1.00000
##  1st Qu.: 1.0000   1st Qu.: 1.0000          1st Qu.:-1.000   1st Qu.:-1.00000
##  Median : 1.0000   Median : 1.0000          Median :-1.000   Median : 0.00000
##  Mean   : 0.7006   Mean   : 0.7415          Mean   :-0.735   Mean   : 0.06395
##  3rd Qu.: 1.0000   3rd Qu.: 1.0000          3rd Qu.:-1.000   3rd Qu.: 1.00000
##  Max.   : 1.0000   Max.   : 1.0000          Max.   : 1.000   Max.   : 1.00000
##  SSLfinal_State    Domain_registeration_length    Favicon
##  Min.   :-1.0000   Min.   :-1.0000             Min.   :-1.0000
##  1st Qu.:-1.0000   1st Qu.:-1.0000             1st Qu.: 1.0000
##  Median : 1.0000   Median :-1.0000             Median : 1.0000
##  Mean   : 0.2509   Mean   :-0.3368             Mean   : 0.6286
##  3rd Qu.: 1.0000   3rd Qu.: 1.0000             3rd Qu.: 1.0000
##  Max.   : 1.0000   Max.   : 1.0000             Max.   : 1.0000
##       port          HTTPS_token       Request_URL      URL_of_Anchor
##  Min.   :-1.0000   Min.   :-1.0000   Min.   :-1.0000   Min.   :-1.00000
##  1st Qu.: 1.0000   1st Qu.: 1.0000   1st Qu.:-1.0000   1st Qu.:-1.00000
##  Median : 1.0000   Median : 1.0000   Median : 1.0000   Median : 0.00000
##  Mean   : 0.7283   Mean   : 0.6751   Mean   : 0.1868   Mean   :-0.07653
##  3rd Qu.: 1.0000   3rd Qu.: 1.0000   3rd Qu.: 1.0000   3rd Qu.: 0.00000
##  Max.   : 1.0000   Max.   : 1.0000   Max.   : 1.0000   Max.   : 1.00000
##  Links_in_tags          SFH          Submitting_to_email  Abnormal_URL
##  Min.   :-1.0000   Min.   :-1.0000   Min.   :-1.0000     Min.   :-1.0000
##  1st Qu.:-1.0000   1st Qu.:-1.0000   1st Qu.: 1.0000     1st Qu.: 1.0000
##  Median : 0.0000   Median :-1.0000   Median : 1.0000     Median : 1.0000
##  Mean   :-0.1181   Mean   :-0.5957   Mean   : 0.6356     Mean   : 0.7053
##  3rd Qu.: 0.0000   3rd Qu.:-1.0000   3rd Qu.: 1.0000     3rd Qu.: 1.0000
##  Max.   : 1.0000   Max.   : 1.0000   Max.   : 1.0000     Max.   : 1.0000
##     Redirect       on_mouseover       RightClick       popUpWidnow
##  Min.   :0.0000   Min.   :-1.0000   Min.   :-1.0000   Min.   :-1.0000
##  1st Qu.:0.0000   1st Qu.: 1.0000   1st Qu.: 1.0000   1st Qu.: 1.0000
##  Median :0.0000   Median : 1.0000   Median : 1.0000   Median : 1.0000
##  Mean   :0.1157   Mean   : 0.7621   Mean   : 0.9139   Mean   : 0.6134
##  3rd Qu.:0.0000   3rd Qu.: 1.0000   3rd Qu.: 1.0000   3rd Qu.: 1.0000
##  Max.   :1.0000   Max.   : 1.0000   Max.   : 1.0000   Max.   : 1.0000
##      Iframe        age_of_domain        DNSRecord        web_traffic
##  Min.   :-1.0000   Min.   :-1.00000   Min.   :-1.0000   Min.   :-1.0000
##  1st Qu.: 1.0000   1st Qu.:-1.00000   1st Qu.:-1.0000   1st Qu.: 0.0000
##  Median : 1.0000   Median : 1.00000   Median : 1.0000   Median : 1.0000
##  Mean   : 0.8169   Mean   : 0.06124   Mean   : 0.3771   Mean   : 0.2873
##  3rd Qu.: 1.0000   3rd Qu.: 1.00000   3rd Qu.: 1.0000   3rd Qu.: 1.0000
##  Max.   : 1.0000   Max.   : 1.00000   Max.   : 1.0000   Max.   : 1.0000
##    Page_Rank        Google_Index     Links_pointing_to_page Statistical_report
##  Min.   :-1.0000   Min.   :-1.0000   Min.   :-1.000         Min.   :-1.0000
##  1st Qu.:-1.0000   1st Qu.: 1.0000   1st Qu.: 0.000         1st Qu.: 1.0000
##  Median :-1.0000   Median : 1.0000   Median : 0.000         Median : 1.0000
##  Mean   :-0.4837   Mean   : 0.7216   Mean   : 0.344         Mean   : 0.7196
##  3rd Qu.: 1.0000   3rd Qu.: 1.0000   3rd Qu.: 1.000         3rd Qu.: 1.0000
##  Max.   : 1.0000   Max.   : 1.0000   Max.   : 1.000         Max.   : 1.0000
##      Result
##  Min.   :-1.0000
##  1st Qu.:-1.0000
##  Median : 1.0000
##  Mean   : 0.1139
##  3rd Qu.: 1.0000
##  Max.   : 1.0000
```

Checking for missing values

```
sapply(q5, function(x) sum(is.na(x)))
```

```
##                          id           having_IP_Address
##                           0                           0
##                  URL_Length          Shortining_Service
##                           0                           0
##            having_At_Symbol    double_slash_redirecting
##                           0                           0
##               Prefix_Suffix           having_Sub_Domain
##                           0                           0
##              SSLfinal_State Domain_registeration_length
##                           0                           0
##                     Favicon                        port
##                           0                           0
##                 HTTPS_token                 Request_URL
##                           0                           0
##               URL_of_Anchor               Links_in_tags
##                           0                           0
##                         SFH         Submitting_to_email
##                           0                           0
##                Abnormal_URL                    Redirect
##                           0                           0
##                on_mouseover                  RightClick
##                           0                           0
##                 popUpWidnow                      Iframe
##                           0                           0
##               age_of_domain                   DNSRecord
##                           0                           0
##                 web_traffic                   Page_Rank
##                           0                           0
##                Google_Index      Links_pointing_to_page
##                           0                           0
##          Statistical_report                      Result
##                           0                           0
```

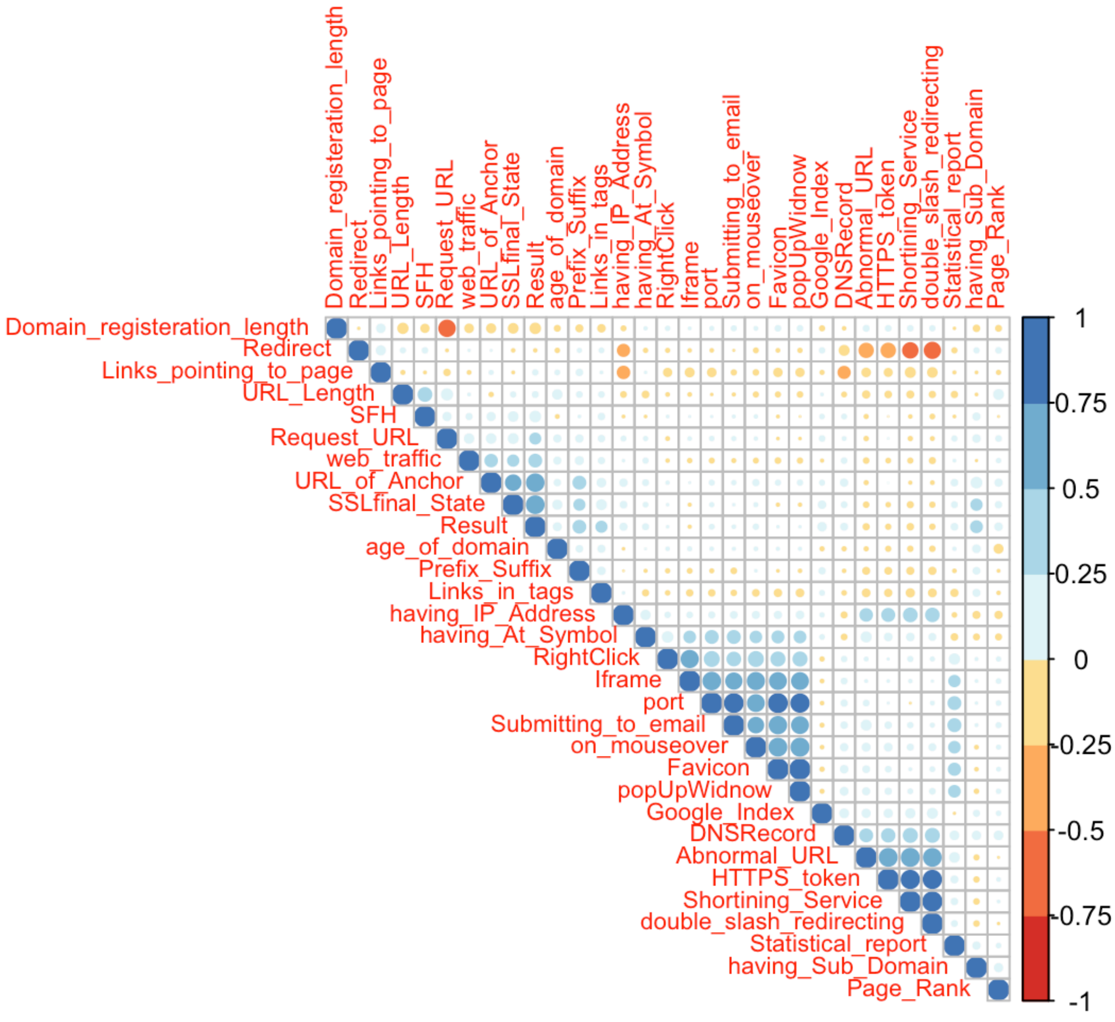
Split data into training and test

```
set.seed(1)
sample <- sample(c(TRUE, FALSE), nrow(q5), replace=TRUE, prob=c(0.7,0.3))
q5_train  <- q5[sample, ]
q5_test   <- q5[!sample, ]

q5_train = select(q5_train, -id)
q5_test = select(q5_test, -id)
```

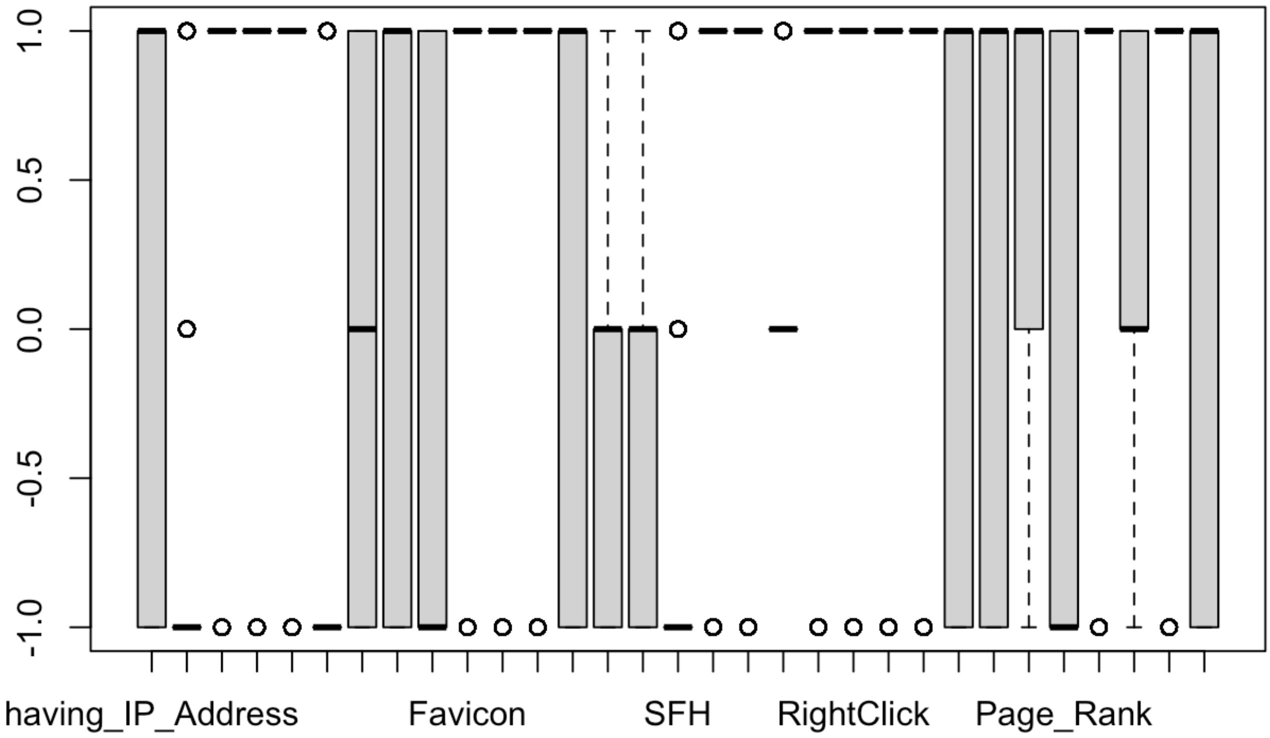
Correlation Plot

```
cor_graph_phishing <- cor(q5_train)
corrplot(cor_graph_phishing, type="upper",order= "hclust", tl.cex = 0.7,col=brewer
.pal(n=8, name="RdYlBu"))
```



Box Plot

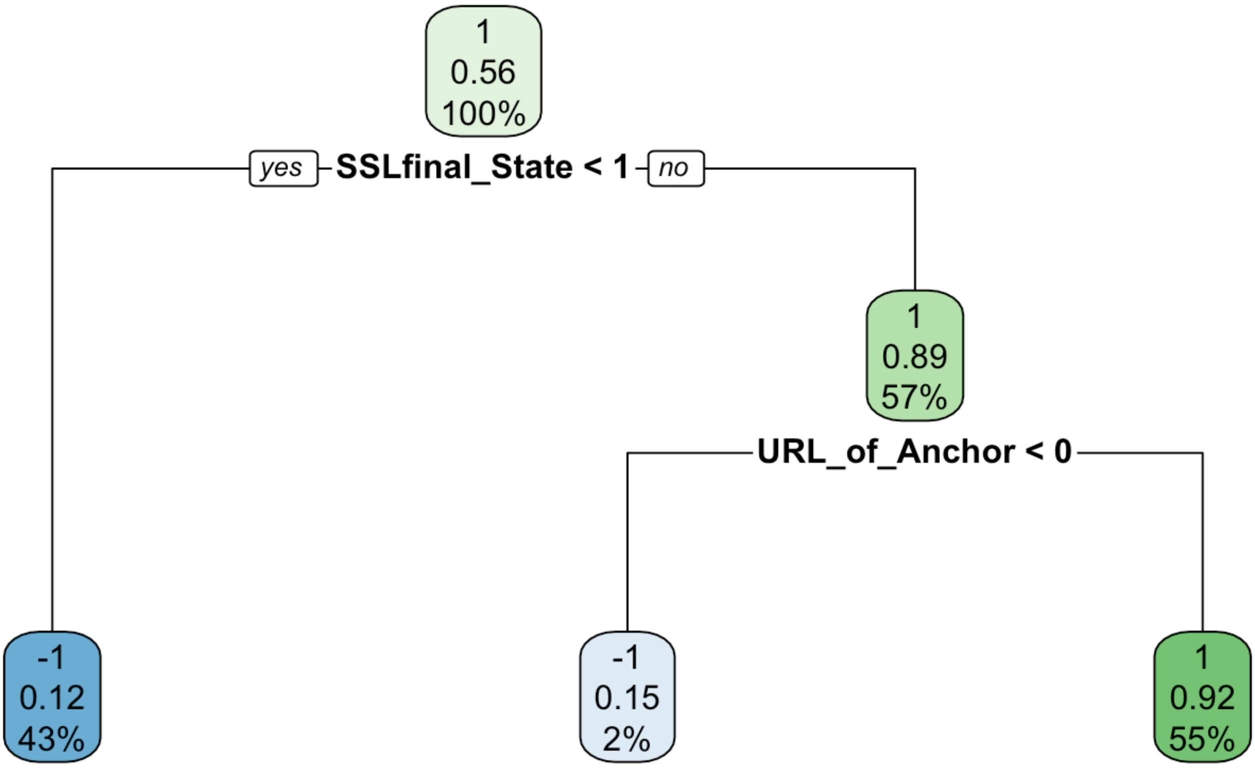
```
boxplot(q5_train)
```



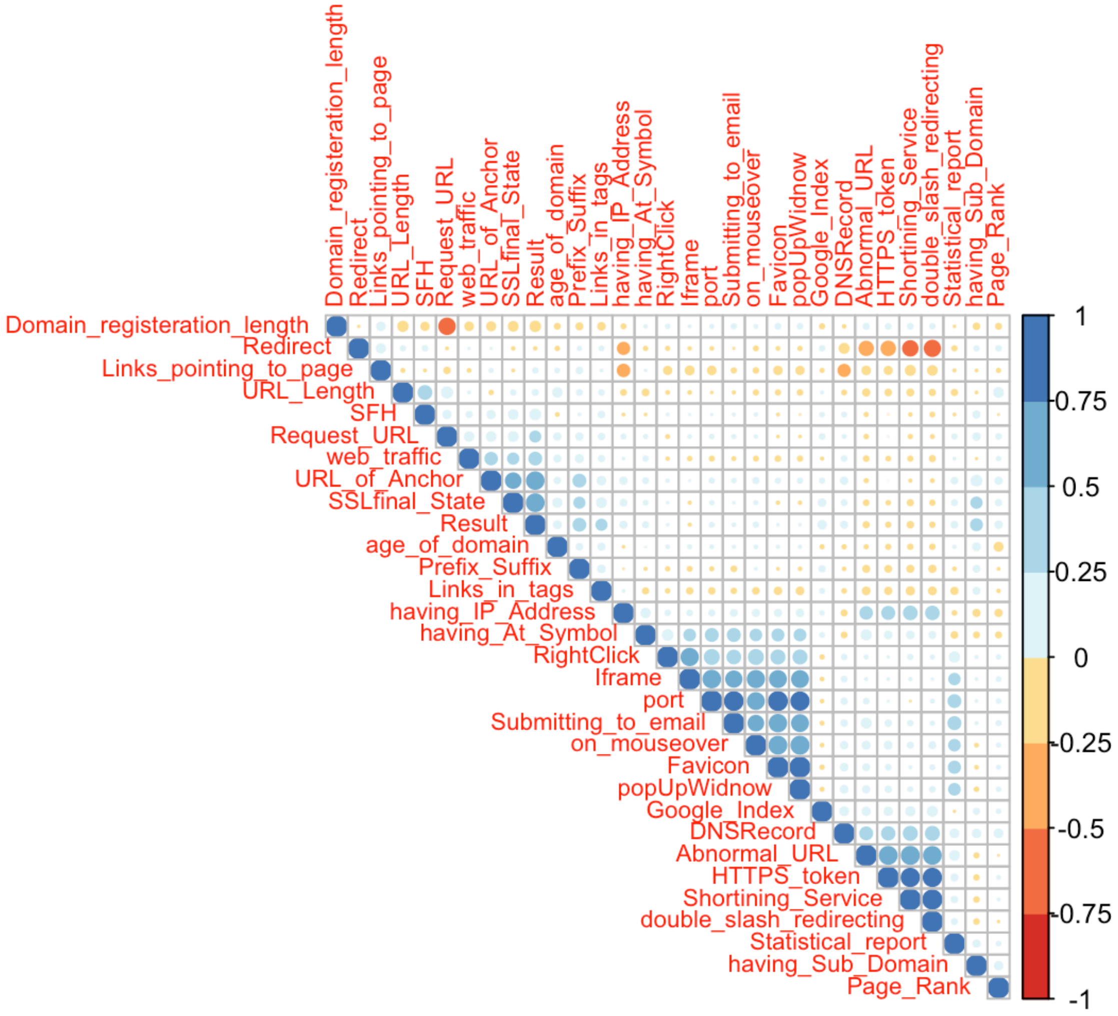
No outliers in this data

Decision Tree

```
library(rpart.plot)
fit_phishing=rpart(Result~.,data=q5_train, parms = list(split="information"), meth
od = 'class')
rpart.plot(fit_phishing, extra = 106)
```



```
q5_train$Result <- as.numeric(q5_train$Result)
cor_graph_phishing <- cor(q5_train)
corrplot(cor_graph_phishing, type="upper",order= "hclust",tl.cex = 0.7,col=brewer.
pal(n=8, name="RdYlBu"))
```



Confusion Matrix for combined and accuracy

```
t_pred_phishing=predict(fit_phishing, q5_train, type='class')
table(q5_train$Result, t_pred_phishing)
```

```
##     t_pred_phishing
##         -1    1
##    -1 3070  339
##    1   410 3884
```

```
confusion_mat_phishing_dt = table(q5_train$Result, t_pred_phishing)
acc_phishing_dt = sum(diag(confusion_mat_phishing_dt))/sum(confusion_mat_phishing_
dt)
print(confusion_mat_phishing_dt)
```

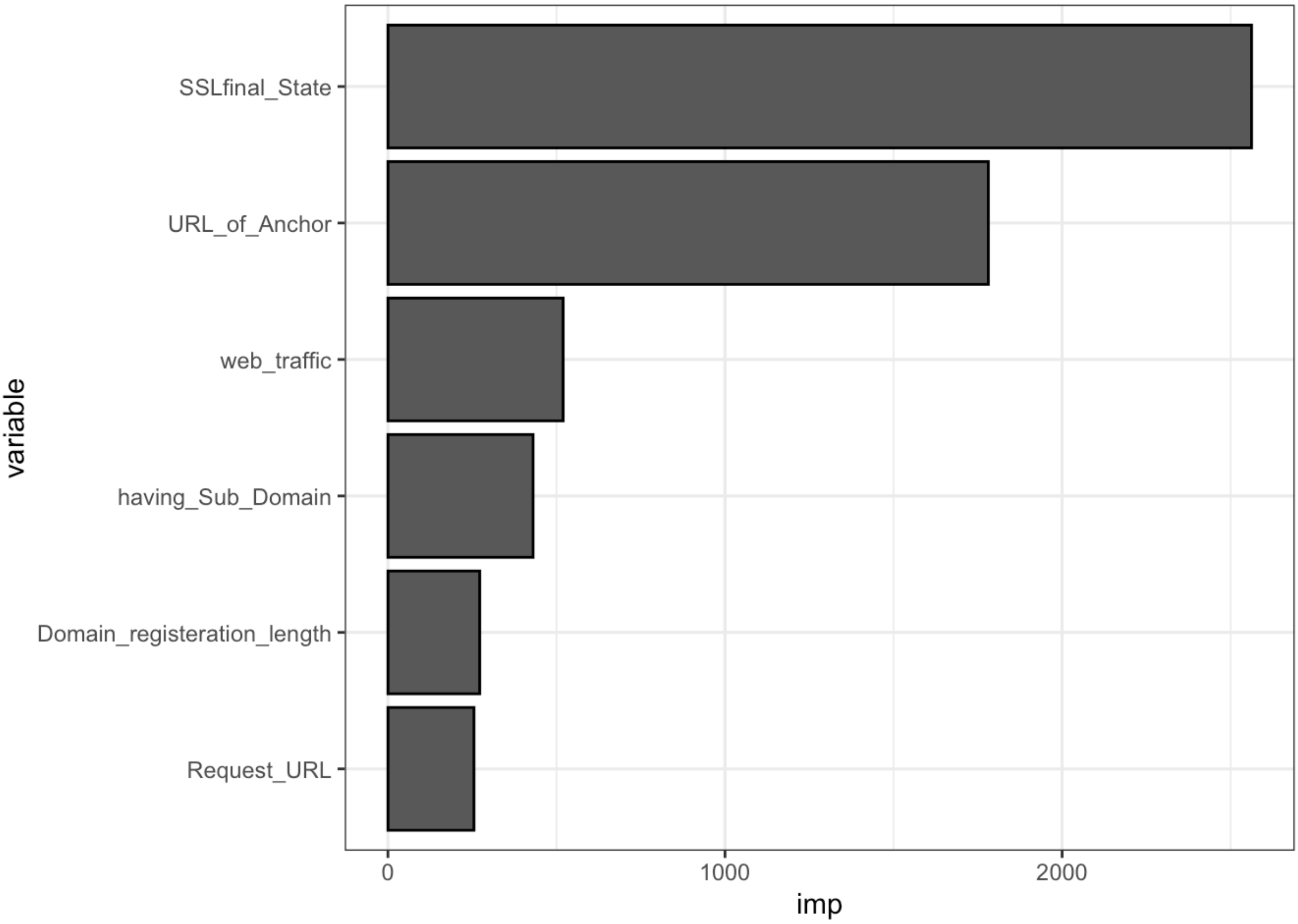
```
##     t_pred_phishing
##         -1    1
##    -1 3070  339
##    1   410 3884
```

```
print(acc_phishing_dt)
```

```
## [1] 0.9027652
```

ggplot

```
df_phishing=data.frame(imp=fit_phishing$variable.importance)
df2_phishing=df_phishing %>%
  tibble::rownames_to_column()%>%
  dplyr::rename("variable"= rowname)%>%
  dplyr::arrange(imp)%>%
  dplyr::mutate(variable = forcats::fct_inorder(variable))
ggplot2::ggplot(df2_phishing) +
  geom_col(aes(x=variable, y=imp), col="black", show.legend = F) +
  coord_flip() +
  scale_fill_grey() +
  theme_bw()
```



Confusion Matrix for test

```
t_pred_phishing_test=predict(fit_phishing, q5_test, type='class')

table(q5_test$Result, t_pred_phishing_test)
```

```
##     t_pred_phishing_test
##        -1    1
##   -1 1355  134
##   1   153 1710
```

```
confusion_mat_test_phishing_dt = table(q5_test$Result, t_pred_phishing_test)
accTest_phishing_dt = sum(diag(confusion_mat_test_phishing_dt))/sum(confusion_mat_
test_phishing_dt)
print(confusion_mat_phishing_dt)
```

```
##     t_pred_phishing
##        -1    1
##   -1 3070  339
##   1   410 3884
```

```
print(accTest_phishing_dt)
```

```
## [1] 0.9143795
```

Random Forest

```
library(party)
```

```
## Loading required package: grid
```

```
## Loading required package: mvtnorm
```

```
## Loading required package: modeltools
```

```
## Loading required package: stats4
```

```
## Loading required package: strucchange
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: sandwich
```

```
##
## Attaching package: 'strucchange'
```

```
## The following object is masked from 'package:stringr':
##
##     boundary
```

```r
library(randomForest)
```

```
## randomForest 4.7-1.1
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:psych':
##
##     outlier
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```r
library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
q5_train$Result <- as.character(q5_train$Result)
q5_train$Result <- as.factor(q5_train$Result)
set.seed(1234)
fit_rf=randomForest(Result~ SSLfinal_State + URL_of_Anchor + web_traffic , data=q5
_train, ntree=500,
                 importance=TRUE, proximity=TRUE)
```

View the forest results

```
print(fit_rf)
```

```
##
## Call:
##  randomForest(formula = Result ~ SSLfinal_State + URL_of_Anchor +      web_traf
fic, data = q5_train, ntree = 500, importance = TRUE,      proximity = TRUE)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 1
##
##         OOB estimate of  error rate: 8.72%
## Confusion matrix:
##      -1    1 class.error
## -1 3034  375  0.11000293
## 1   297 3997  0.06916628
```

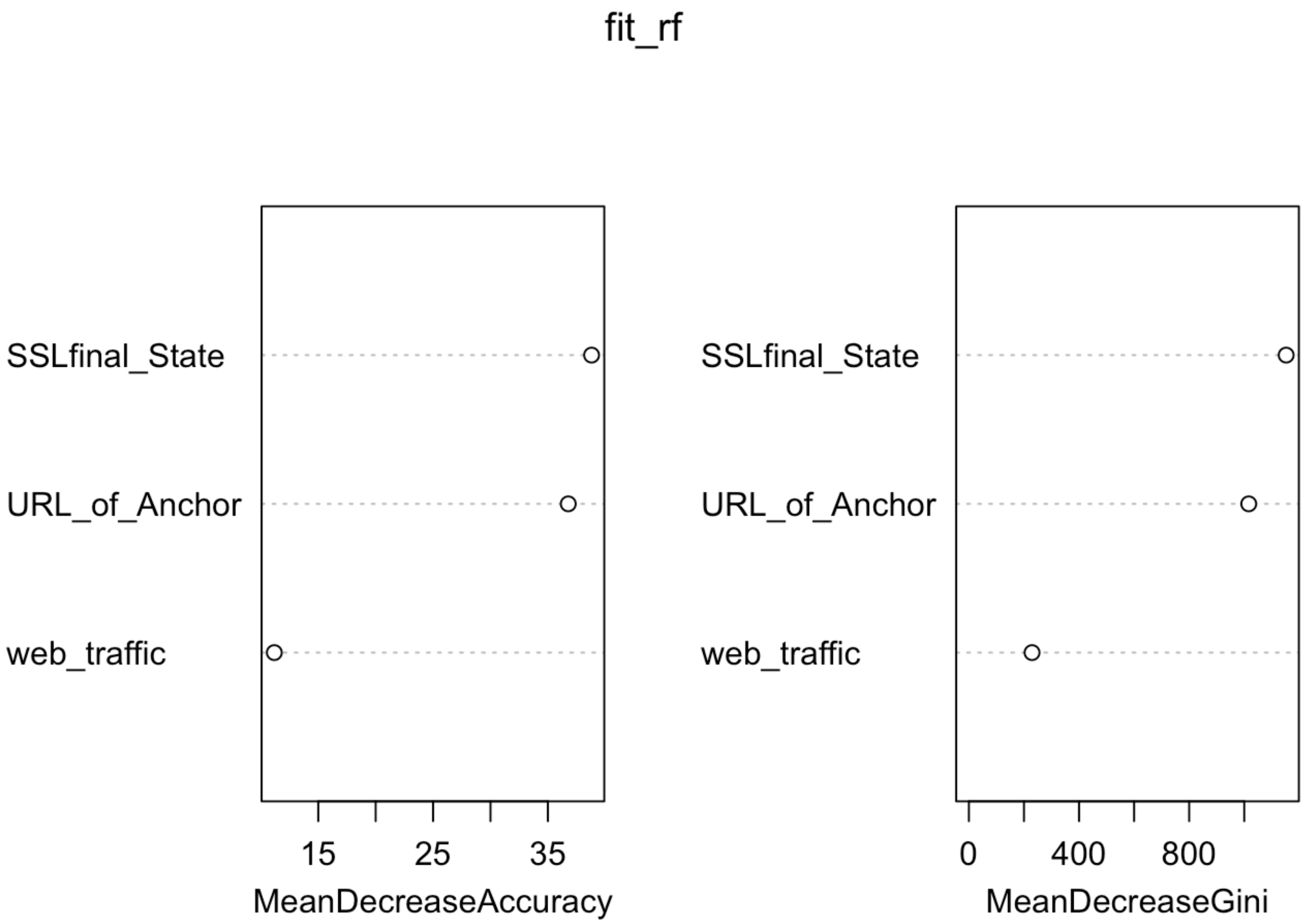
Importance of each predictor.

```
out.importance <- round(importance(fit_rf), 2)
print(out.importance )
```

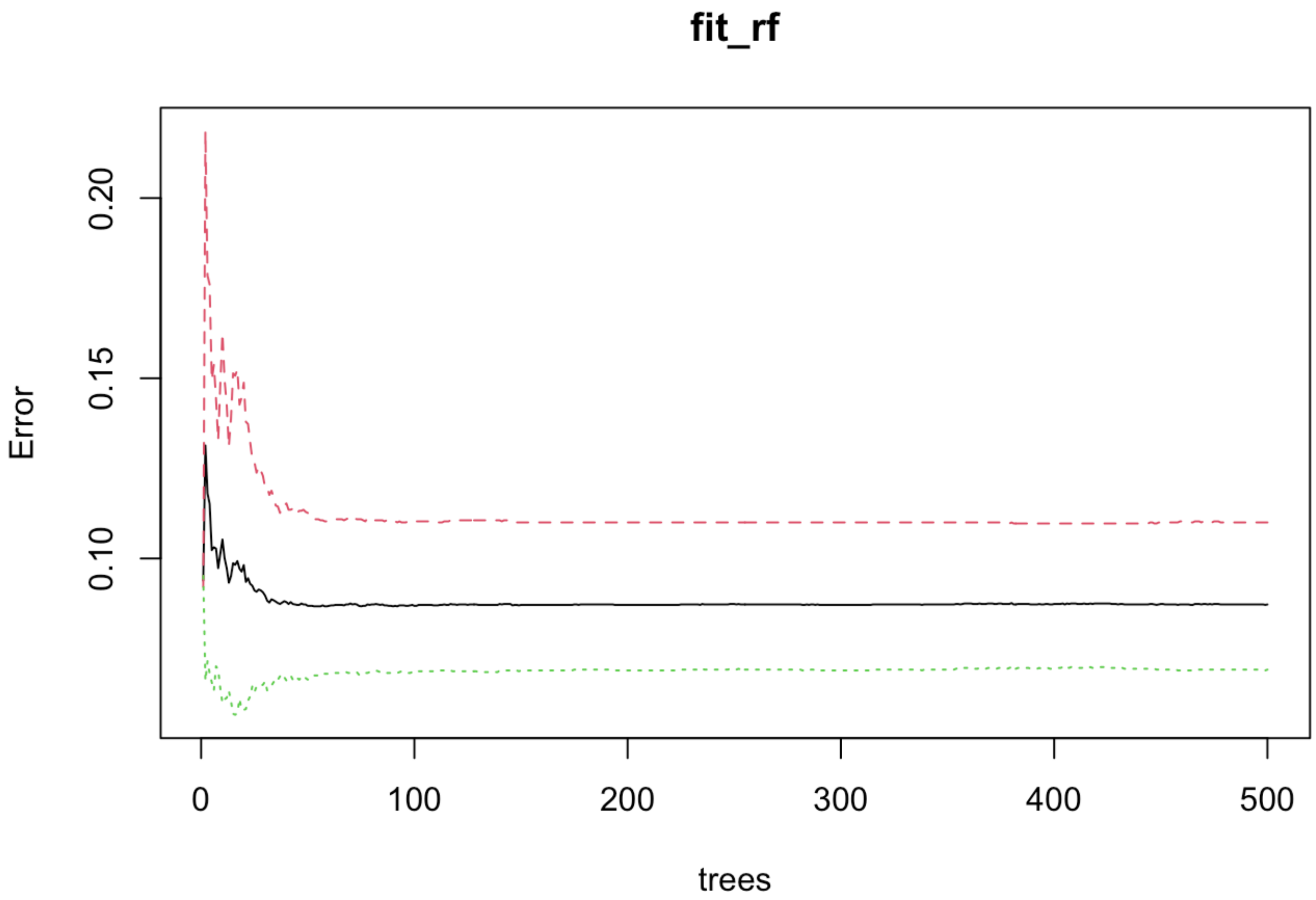
```
##                   -1     1 MeanDecreaseAccuracy MeanDecreaseGini
## SSLfinal_State 23.36 43.68                38.80          1152.25
## URL_of_Anchor  20.94 36.98                36.77          1016.31
## web_traffic     9.73  8.75                11.17           229.32
```

Graph of RF Model

```
varImpPlot(fit_rf)
```


fit_rf
SSLfinal_State
URL_of_Anchor
web_traffic
15
25
35
MeanDecreaseAccuracy
SSLfinal_State
URL_of_Anchor
web_traffic
0
400
800
MeanDecreaseGini

```
plot(fit_rf)
```



As we increase the number of trees, the error decreases exponentially. This is why we prefer Random Forest over Decision Tree. Moreover, for large quantities of data, Random Forest is more effective and quicker than Decision Tree.

From the above data we can observe %IncMSE and IncNodePurity where they define Mean Decrease Accuracy and Mean Decrease Gini respecively. With the former, it shows how much our accuracy will get affected without that particular variable and the latter shows how important a variable is (higher is better).

Accuracy of Decision Tree = 90.27% Accuracy of Random Forest = 91.57%