Statistics 652 - Final

Ken Vu

March 10, 2021

Final

For the *Ozone* data from the R package *mlbench* try the following machine learning prediction algorithm that is useful for feature selection.

Read the paper Feature Selection with the Boruta Package and implement the algorithm.

Which features are most important as determined by the Boruta RandomForest Algorithm?

Answers: Based on the results of the Boruta RandomForest algorithm, we find that the most important features are V1, V4, V5, V6, V7, V8, V9, V10, V11, V12, and V13.

Code and Comments (Final):

Get all of the libraries first

```
library(mlbench)
```

```
## Warning: package 'mlbench' was built under R version 4.0.4
```

```
library(Boruta)
```

```
## Warning: package 'Boruta' was built under R version 4.0.4
```

```
set.seed(300)
```

1) Load the data set

```
data(Ozone)
data_raw <- Ozone
```

2) Explore the data set

```
summary(data_raw)
```

```
##        V1            V2      V3           V4              V5
##  1      : 31   1      : 12   1:52   Min.   : 1.00   Min.   :5320
##  3      : 31   2      : 12   2:52   1st Qu.: 5.00   1st Qu.:5700
##  5      : 31   3      : 12   3:52   Median : 9.00   Median :5770
##  7      : 31   4      : 12   4:53   Mean   :11.53   Mean   :5753
##  8      : 31   5      : 12   5:53   3rd Qu.:16.00   3rd Qu.:5830
##  10     : 31   6      : 12   6:52   Max.   :38.00   Max.   :5950
##  (Other):180   (Other):294   7:52   NA's   :5       NA's   :12
##        V6              V7             V8              V9
```

```
##  Min.   : 0.000   Min.   :19.00   Min.   :25.00   Min.   :27.68
##  1st Qu.: 3.000   1st Qu.:49.00   1st Qu.:51.00   1st Qu.:49.73
##  Median : 5.000   Median :65.00   Median :62.00   Median :57.02
##  Mean   : 4.869   Mean   :58.48   Mean   :61.91   Mean   :56.85
##  3rd Qu.: 6.000   3rd Qu.:73.00   3rd Qu.:72.00   3rd Qu.:66.11
##  Max.   :11.000   Max.   :93.00   Max.   :93.00   Max.   :82.58
##                   NA's   :15      NA's   :2       NA's   :139
##       V10             V11             V12             V13
##  Min.   : 111   Min.   :-69.0   Min.   :27.50   Min.   :  0.0
##  1st Qu.: 890   1st Qu.:-10.0   1st Qu.:51.26   1st Qu.: 70.0
##  Median :2125   Median : 24.0   Median :62.24   Median :110.0
##  Mean   :2591   Mean   : 17.8   Mean   :60.93   Mean   :123.3
##  3rd Qu.:5000   3rd Qu.: 45.0   3rd Qu.:70.52   3rd Qu.:150.0
##  Max.   :5000   Max.   :107.0   Max.   :91.76   Max.   :500.0
##  NA's   :15     NA's   :1       NA's   :14
```

Investigate the missing values

```
missing_values <- as.data.frame(sapply(data_raw, function(x) sum(is.na(x))))
colnames(missing_values) <- c("Number of NAs")
missing_values$freq <- round(missing_values[,c("Number of NAs")]/length(data_raw[,1]),3)

# Get columns where 10% of data r less is NAs
colnames_lessmiss <- rownames(missing_values[missing_values$freq <= 0.1,])
missing_values[]
```

```
##     Number of NAs  freq
## V1              0 0.000
## V2              0 0.000
## V3              0 0.000
## V4              5 0.014
## V5             12 0.033
## V6              0 0.000
## V7             15 0.041
## V8              2 0.005
## V9            139 0.380
## V10            15 0.041
## V11             1 0.003
## V12            14 0.038
## V13             0 0.000
```

See if you can mean impute for some of the variables

```
# Mean impute for the columns with 10% or less data missing
NA2mean <- function(x) replace(x, is.na(x), mean(x, na.rm = TRUE))
data_clean <- data_raw
data_clean[,colnames_lessmiss] <- lapply(data_clean[,colnames_lessmiss], NA2mean)
```

```
## Warning in mean.default(x, na.rm = TRUE): argument is not numeric or logical:
## returning NA
```

```
## Warning in mean.default(x, na.rm = TRUE): argument is not numeric or logical:
## returning NA
```

```
## Warning in mean.default(x, na.rm = TRUE): argument is not numeric or logical:
## returning NA
```

```
# Check the proportion of missing data to total row count again
missing_values_imputed <- as.data.frame(sapply(data_clean, function(x) sum(is.na(x))))
colnames(missing_values_imputed) <- c("Number of NAs")
missing_values_imputed$freq <- round(missing_values_imputed[,c("Number of NAs")]/length(data_clean[,1])
missing_values_imputed
```

```
##     Number of NAs freq
## V1              0 0.00
## V2              0 0.00
## V3              0 0.00
## V4              0 0.00
## V5              0 0.00
## V6              0 0.00
## V7              0 0.00
## V8              0 0.00
## V9            139 0.38
## V10             0 0.00
## V11             0 0.00
## V12             0 0.00
## V13             0 0.00
```

Now, clean out the data of NAs

```
data_clean <- na.omit(data_clean)
```

3) Train Boruta onto the data set.

We limit max runs to 12 in case there are attributes with importance too close to MSZA that might make algorithm indecisive about which attributes to consider important.

```
Boruta.Short <- Boruta(V4 ~ ., data = data_clean, doTrace = 2, ntree = 500, maxRuns=12)
```

```
##  1. run of importance source...
##  2. run of importance source...
##  3. run of importance source...
##  4. run of importance source...
##  5. run of importance source...
##  6. run of importance source...
##  7. run of importance source...
##  8. run of importance source...
##  9. run of importance source...
##  10. run of importance source...
##  11. run of importance source...
## After 11 iterations, +1.8 secs:
##  confirmed 9 attributes: V1, V10, V11, V12, V13 and 4 more;
##  rejected 2 attributes: V3, V6;
```

```
##  still have 1 attribute left.
```

```
Boruta.Short
```

```
## Boruta performed 11 iterations in 1.829209 secs.
##  9 attributes confirmed important: V1, V10, V11, V12, V13 and 4 more;
##  2 attributes confirmed unimportant: V3, V6;
##  1 tentative attributes left: V2;
```

4) Now, try to do a tentative fix on the algorithm to sort out the tentative variables.

```
Boruta.TentFix<- TentativeRoughFix(Boruta.Short)
Boruta.TentFix
```

```
## Boruta performed 11 iterations in 1.829209 secs.
## Tentatives roughfixed over the last 11 iterations.
##  9 attributes confirmed important: V1, V10, V11, V12, V13 and 4 more;
##  3 attributes confirmed unimportant: V2, V3, V6;
```

5) Plot the results of Boruta

```
plot(Boruta.TentFix, las=2)
```