Vandals

Ethan Marcano

2022-03-16

```r
vandals <- read_csv("Wikipedia.csv")
```

```
## Rows: 3876 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (6): Vandal, Minor, LoggedIn, HTTP, NumWordsAdded, NumWordsRemoved
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
vandal_count <- vandals %>%
    select(everything()) %>%
    filter(Vandal == 1)
count(vandal_count)
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1  1815
```

- There are 1815 cases of vandalism.

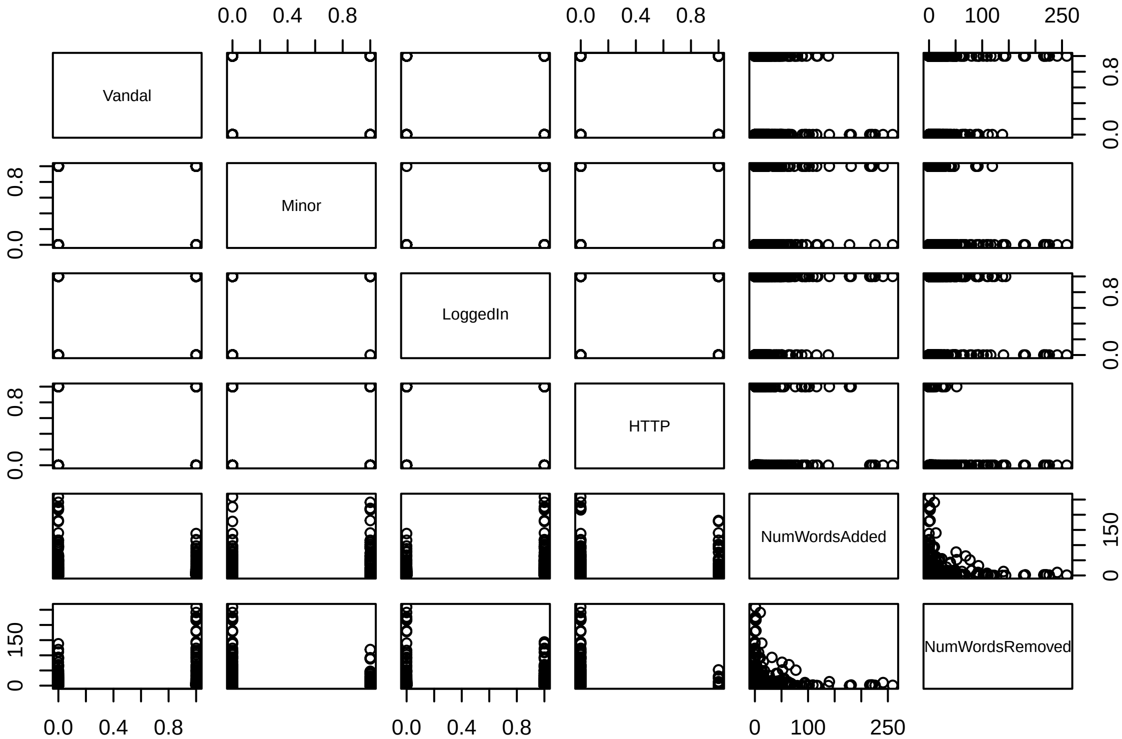
```r
mean(vandals$NumWordsAdded)
```

```
## [1] 4.050052
```

```r
mean(vandals$NumWordsRemoved)
```

```
## [1] 3.5129
```

```r
pairs(vandals)
```




```
cor(vandals)
```

```
##                        Vandal        Minor    LoggedIn        HTTP
## Vandal           1.0000000000 -0.213995217 -0.42925457  0.15155368
## Minor           -0.2139952169  1.000000000  0.44516561 -0.08429685
## LoggedIn        -0.4292545749  0.445165610  1.00000000 -0.11063301
## HTTP             0.1515536849 -0.084296852 -0.11063301  1.00000000
## NumWordsAdded   -0.0007289019 -0.007726385  0.02622296  0.11442149
## NumWordsRemoved  0.0363597359 -0.037629294 -0.03642207 -0.03986582
##                 NumWordsAdded NumWordsRemoved
## Vandal          -0.0007289019      0.03635974
## Minor           -0.0077263847     -0.03762929
## LoggedIn         0.0262229639     -0.03642207
## HTTP             0.1144214902     -0.03986582
## NumWordsAdded    1.0000000000      0.02523534
## NumWordsRemoved  0.0252353411      1.00000000
```

- Vandal is most negatively correlated to LoggedIn.

```
set.seed(100)
spl = sample.split(vandals$Vandal, SplitRatio = 0.7)
vandalstrain = subset(vandals, spl == TRUE)
vandalstest = subset(vandals, spl == FALSE)

nrow(vandalstrain)/nrow(vandals)
```

```
## [1] 0.6999484
```

```
nrow(vandalstest)/nrow(vandals)
```

```
## [1] 0.3000516
```

Question B)

```
simple_vandal <- table(vandalstest$Vandal)
simple_vandal
```

```
##
##   0   1
## 618 545
```

```
simple_vandal[2]/sum(simple_vandal)
```

```
##         1
## 0.4686156
```

- The baseline model is 47% accurate.

Question C)

```
set.seed(100)
spl <- sample.split(vandalstrain$Vandal, SplitRatio = 0.5)
vandals_validate_train <- subset(vandalstrain, spl == TRUE)
vandals_validate_test <- subset(vandalstrain, spl == FALSE)

vandal_CART1 <- rpart(Vandal ~ Minor + LoggedIn + HTTP + NumWordsAdded +
    NumWordsRemoved, method = "class", data = vandals_validate_train,
    minbucket = 5)
vandal_CART2 <- rpart(Vandal ~ Minor + LoggedIn + HTTP + NumWordsAdded +
    NumWordsRemoved, method = "class", data = vandals_validate_train,
    minbucket = 15)
vandal_CART3 <- rpart(Vandal ~ Minor + LoggedIn + HTTP + NumWordsAdded +
    NumWordsRemoved, method = "class", data = vandals_validate_train,
    minbucket = 25)

vandal_predict1 <- predict(vandal_CART1, newdata = vandals_validate_test,
    type = "class")
vandal_predict2 <- predict(vandal_CART2, newdata = vandals_validate_test,
    type = "class")
vandal_predict3 <- predict(vandal_CART3, newdata = vandals_validate_test,
    type = "class")

vandal_predict_table1 <- table(vandals_validate_test$Vandal,
    vandal_predict1)
vandal_predict_table2 <- table(vandals_validate_test$Vandal,
    vandal_predict2)
vandal_predict_table3 <- table(vandals_validate_test$Vandal,
    vandal_predict3)

vandal_predict_table1
```

```
##    vandal_predict1
##       0   1
##   0 606 115
##   1 284 351
```

```
sum(diag(vandal_predict_table1))/sum(vandal_predict_table1)
```

```
## [1] 0.7057522
```

```r
vandal_predict_table2
```

```
##    vandal_predict2
##       0   1
##   0 606 115
##   1 284 351
```

```r
sum(diag(vandal_predict_table2))/sum(vandal_predict_table2)
```

```
## [1] 0.7057522
```

```r
vandal_predict_table3
```

```
##    vandal_predict3
##       0   1
##   0 604 117
##   1 281 354
```

```r
sum(diag(vandal_predict_table3))/sum(vandal_predict_table3)
```

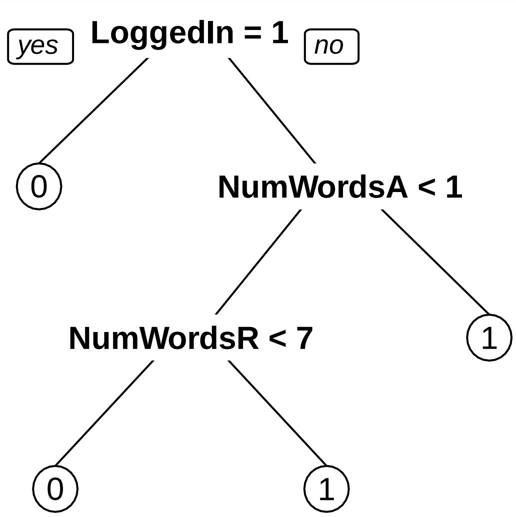
```
## [1] 0.7064897
```

```r
vandal_CART <- rpart(Vandal ~ Minor + LoggedIn + HTTP + NumWordsAdded +
    NumWordsRemoved, method = "class", data = vandalstrain, minbucket = 25)
summary(vandal_CART)
```

```
## Call:
## rpart(formula = Vandal ~ Minor + LoggedIn + HTTP + NumWordsAdded +
##     NumWordsRemoved, data = vandalstrain, method = "class", minbucket = 25)
##   n= 2713
##
##           CP nsplit rel error    xerror       xstd
## 1 0.36614173      0 1.0000000 1.0000000 0.02046475
## 2 0.01456693      1 0.6338583 0.6338583 0.01873521
## 3 0.01000000      3 0.6047244 0.6188976 0.01860474
##
## Variable importance
##        LoggedIn   NumWordsAdded NumWordsRemoved            HTTP
##              72              24               3               1
##
## Node number 1: 2713 observations,    complexity param=0.3661417
##   predicted class=0  expected loss=0.4681165  P(node) =1
##     class counts:  1443  1270
##    probabilities: 0.532 0.468
##   left son=2 (1810 obs) right son=3 (903 obs)
##   Primary splits:
##       LoggedIn        < 0.5  to the right, improve=226.65320, (0 missing)
##       NumWordsAdded   < 0.5  to the left,  improve= 95.19137, (0 missing)
##       Minor           < 0.5  to the right, improve= 56.02468, (0 missing)
##       NumWordsRemoved < 0.5  to the right, improve= 28.98684, (0 missing)
##       HTTP            < 0.5  to the left,  improve= 27.98224, (0 missing)
##   Surrogate splits:
##       NumWordsRemoved < 139  to the left,  agree=0.67, adj=0.010, (0 split)
##       HTTP            < 0.5  to the left,  agree=0.67, adj=0.009, (0 split)
##
## Node number 2: 1810 observations
```

```
##   predicted class=0  expected loss=0.3237569  P(node) =0.6671581
##     class counts:  1224   586
##    probabilities: 0.676 0.324
##
## Node number 3: 903 observations,    complexity param=0.01456693
##   predicted class=1  expected loss=0.2425249  P(node) =0.3328419
##     class counts:   219   684
##    probabilities: 0.243 0.757
##   left son=6 (302 obs) right son=7 (601 obs)
##   Primary splits:
##       NumWordsAdded   < 0.5  to the left,  improve=74.894540, (0 missing)
##       HTTP            < 0.5  to the left,  improve= 6.332982, (0 missing)
##       NumWordsRemoved < 15.5 to the left,  improve= 3.469435, (0 missing)
##
## Node number 6: 302 observations,    complexity param=0.01456693
##   predicted class=0  expected loss=0.4701987  P(node) =0.1113159
##     class counts:   160   142
##    probabilities: 0.530 0.470
##   left son=12 (267 obs) right son=13 (35 obs)
##   Primary splits:
##       NumWordsRemoved < 6.5  to the left,  improve=7.184389, (0 missing)
##
## Node number 7: 601 observations
##   predicted class=1  expected loss=0.09816972  P(node) =0.221526
##     class counts:    59   542
##    probabilities: 0.098 0.902
##
## Node number 12: 267 observations
##   predicted class=0  expected loss=0.4307116  P(node) =0.09841504
##     class counts:   152   115
##    probabilities: 0.569 0.431
##
## Node number 13: 35 observations
##   predicted class=1  expected loss=0.2285714  P(node) =0.01290085
##     class counts:     8    27
##    probabilities: 0.229 0.771
```

```
prp(vandal_CART)
```



- The relevant independent variables are LoggedIn, NumWordsAdded, and NumWordsRemoved. LoggedIn is the most relevant, with NumWordsRemoved branching off of NumWordsAdded.

```
vandal_CART_predict <- predict(vandal_CART, newdata = vandalstest,
    type = "class")
vandal_predict_CART_final <- table(vandalstest$Vandal, vandal_CART_predict)

vandal_predict_CART_final
```

ii)

```
##    vandal_CART_predict
##       0   1
##   0 589  29
##   1 272 273
```

```
sum(diag(vandal_predict_CART_final))/sum(vandal_predict_CART_final)
```

```
## [1] 0.7411866
```

- The model is 74.1% accurate.

Question D)

```
vandalstrain$Vandal <- as.factor(vandalstrain$Vandal)
vandalstest$Vandal <- as.factor(vandalstest$Vandal)

vandals_forest <- randomForest(Vandal ~ Minor + LoggedIn + HTTP +
    NumWordsAdded + NumWordsRemoved, data = vandalstrain, ntree = 200,
    nodesize = 15)
vandals_predict_forest <- predict(vandals_forest, newdata = vandalstest)
vandals_predict_final <- table(vandalstest$Vandal, vandals_predict_forest)

vandals_predict_final
```

```
##    vandals_predict_forest
##       0   1
##   0 564  54
##   1 237 308
```

```
sum(diag(vandals_predict_final))/sum(vandals_predict_final)
```

```
## [1] 0.749785
```

- The model is slightly more accurate at ~75%.

Question E)

i)

- It can potentially be useful but for now it just gives hints as to who is committing vandalism, and the accuracy could be better with the introduction of more independent variables.

ii)

- I would want to collect data on whether it was under an umbrella category (sports, politics, biology) or if the article in question was something that happened recently. Current events see more foot traffic and are heavily edited.

iii)

- This model may not extend easily due to other pages often containing more contentious material, and thus subject to more vandalism as a whole.