Project Modeling with Classification Trees

Joshua Ambrose

9/19/2020

Summary

Summary of the data used for this project. PRODUCT is removed as a predictor for this project since it records an event that takes place after the outcome. ANSWERED, FEMALE, JOB, RENT, OWN_RES, NEW_CAR, MOBILE were converted to factors.

```
##  answered     income       female        age       job      num_dependents
##  0:2285   Min.   :  2760   0:4729   Min.   :19.0   0: 108   Min.   :1.000
##  1:2715   1st Qu.: 13520   1: 271   1st Qu.:26.0   1: 956   1st Qu.:1.000
##           Median : 23370            Median :32.0   2:3151   Median :1.000
##           Mean   : 33908            Mean   :34.8   3: 785   Mean   :1.147
##           3rd Qu.: 42490            3rd Qu.:40.0            3rd Qu.:1.000
##           Max.   :159450            Max.   :74.0            Max.   :2.000
##  rent     own_res     new_car          chk_acct       sav_acct
##  0:3931   0:1586   Min.   :0.0000   Min.   :0.00   Min.   :0.0000
##  1:1069   1:3414   1st Qu.:0.0000   1st Qu.:0.00   1st Qu.:0.0000
##                    Median :0.0000   Median :1.00   Median :0.0000
##                    Mean   :0.2384   Mean   :1.47   Mean   :0.9824
##                    3rd Qu.:0.0000   3rd Qu.:3.00   3rd Qu.:2.0000
##                    Max.   :1.0000   Max.   :3.00   Max.   :4.0000
##    num_accts     mobile
##  Min.   :0.000   0:4528
##  1st Qu.:2.000   1: 472
##  Median :2.000
##  Mean   :2.384
##  3rd Qu.:3.000
##  Max.   :4.000
```

Proportion of Answered Calls

This is the proportion of answered calls for question one. Since the data is binary where 0 indicates unanswered and 1 indicates answered, the mean is the average of answered calls.

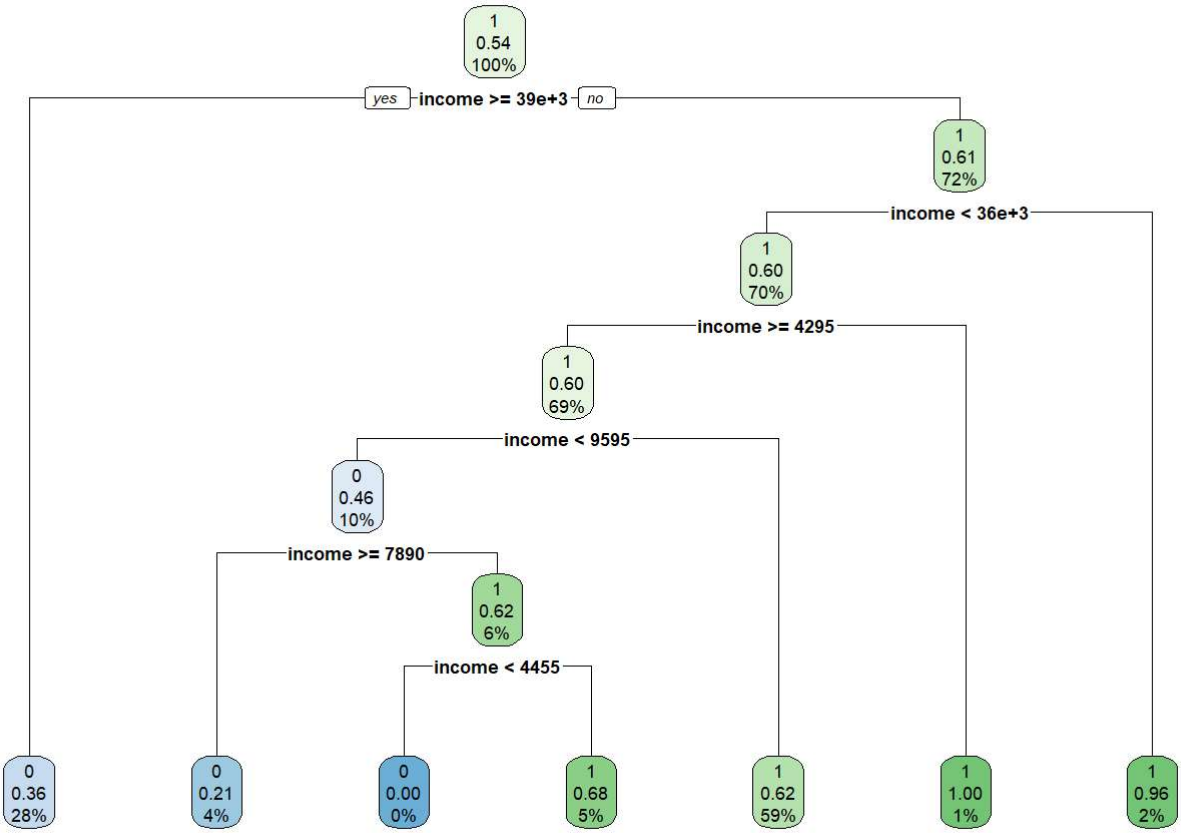
```
summary(data_unclean$answered)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   1.000   0.543   1.000   1.000
```

Income Model

The below defined income model has an accuracy of 0.648.

```
income_model <- rpart(answered ~ income, data = data)
rpart.plot(income_model)
```




```
(predict(income_model, type = "class") == data$answered) %>%
  mean
```

```
## [1] 0.648
```

Information Gain

IG(parent,children)=entropy(parent)−[p(c1)entropy(c1)+p(c2)entropy(c2)+…]

The calculations used to get IG, and IG itself, are shown below:

```
income_model
```

```
## n= 5000
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
##  1) root 5000 2285 1 (0.4570000 0.5430000)
##    2) income>=39135 1385  495 0 (0.6425993 0.3574007) *
##    3) income< 39135 3615 1395 1 (0.3858921 0.6141079)
##      6) income< 36355 3490 1390 1 (0.3982808 0.6017192)
##       12) income>=4295 3450 1390 1 (0.4028986 0.5971014)
##         24) income< 9595 480  223 0 (0.5354167 0.4645833)
##           48) income>=7890 183   39 0 (0.7868852 0.2131148) *
##           49) income< 7890 297  113 1 (0.3804714 0.6195286)
##             98) income< 4455 25    0 0 (1.0000000 0.0000000) *
##             99) income>=4455 272   88 1 (0.3235294 0.6764706) *
##         25) income>=9595 2970 1133 1 (0.3814815 0.6185185) *
##       13) income< 4295 40    0 1 (0.0000000 1.0000000) *
##      7) income>=36355 125    5 1 (0.0400000 0.9600000) *
```

```
entropy_parent <- -0.4570000 * log2(0.4570000) - 0.5430000 * log2(0.5430000)
print(entropy_parent)
```

```
## [1] 0.9946583
```

```
pc1 <- 1385/5000
print(pc1)
```

```
## [1] 0.277
```

```
pc2 = 1 - pc1
print(pc2)
```

```
## [1] 0.723
```

```
entropyc1 = -0.3574007 * log2(0.3574007) - 0.6425993 * log2(0.6425993)
print(entropyc1)
```

```
## [1] 0.9405044
```

```
entropyc2 = -0.3858921  * log2(0.3858921 ) - 0.6141079 * log2(0.6141079)
print(entropyc2)
```

```
## [1] 0.9620973
```

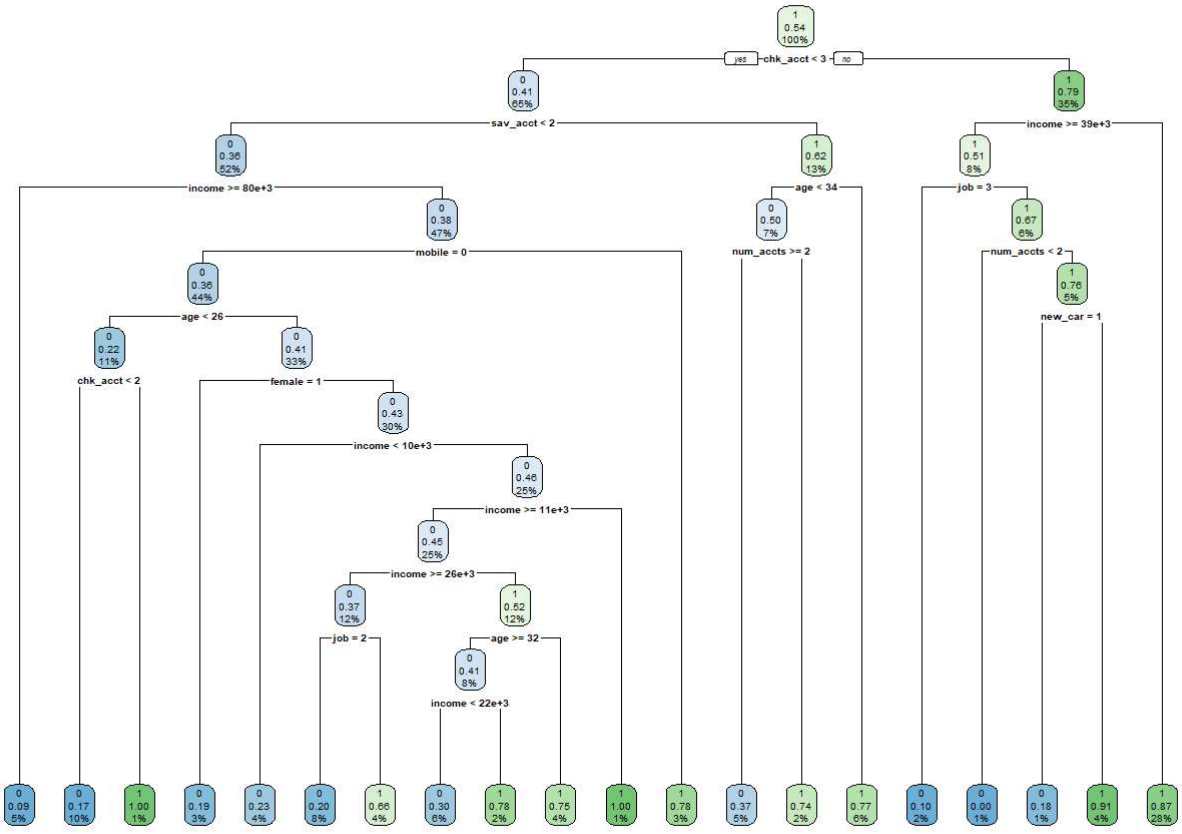
```
IG <- entropy_parent - (pc1 * entropyc1 + pc2 * entropyc2)
print(IG)
```

```
## [1] 0.03854222
```

Tree Model

The below defined tree model has an accuracy of 0.8104.

```
tree_model <- rpart(answered ~ ., data = data)
rpart.plot(tree_model)
```




```
(predict(tree_model, type = "class") == data$answered) %>%
  mean
```

```
## [1] 0.8104
```