Predicting Citrus Hill vs Minute Maid Orange Juice Purchases Based on Sales Information

Data Analysis Assignment Lesson 12

David Chen

April 21st, 2020

Introduction

For our analysis, we will classify/predict purchases of Citrus Hill (CH) vs Minute Maid (MM) brands of orange juice based on pricing and store characteristics contained within the `OJ` dataset. We will fit a classification tree, and then use cross validation to select the optimal sized tree based on classification error rate. While our pruned tree reduced the size by 1, the training and testing error rates, 0.161 and 0.163 respectively, both did not change. Thus, we conclude with our classification error rate of 0.163.

Data

Taken from Data Analysis Assignment Lesson 10

The `OJ` dataset is sourced from the ISLR package, originally published in 1998. This dataset contains 1070 observations, where customers purchased either Citrus Hill or Minute Maid orange juice. The variables predominately consist of store/pricing information. The following variable descriptions are copied from the CRAN documentation:

- `Purchase` - A factor with levels CH and MM indicating whether the customer purchased Citrus Hill or Minute Maid Orange Juice
- `WeekofPurchase` - Week of purchase
- `StoreID` - Store ID
- `PriceCH` - Price charged for CH
- `PriceMM` - Price charged for MM
- `DiscCH` - Discount offered for CH
- `DiscMM` - Discount offered for MM
- `SpecialCH` - Indicator of special on CH
- `SpecialMM` - Indicator of special on MM
- `LoyalCH` - Customer brand loyalty for CH
- `SalePriceMM` - Sale price for MM
- `SalePriceCH` - Sale price for CH
- `PriceDiff` - Sale price of MM less sale price of CH
- `Store7` - A factor with levels No and Yes indicating whether the sale is at Store 7
- `PctDiscMM` - Percentage discount for MM
- `PctDiscCH` - Percentage discount for CH
- `ListPriceDiff` - List price of MM less list price of CH
- `STORE` - Which of 5 possible stores the sale occurred at

Exploratory Analysis

While in Data Analysis Assignment 10 we removed multiple variables due to overlapping behaviors, we will not do that for this assignment. This is because we are using classifications trees, and inherently splits would not occur for variables that explain the same exact information (i.e. if there would have been a split for store 7, `STORE` and `STORE7` will not both be represented).

For the sake of understanding the variables, we will explore some of their behaviors. Looking at the `StoreID` variable, we note that all the purchases seem evenly distributed within a store besides Store 7. Store 7 seems to purchase CH branded OJ at higher proportions than the remaining stores.

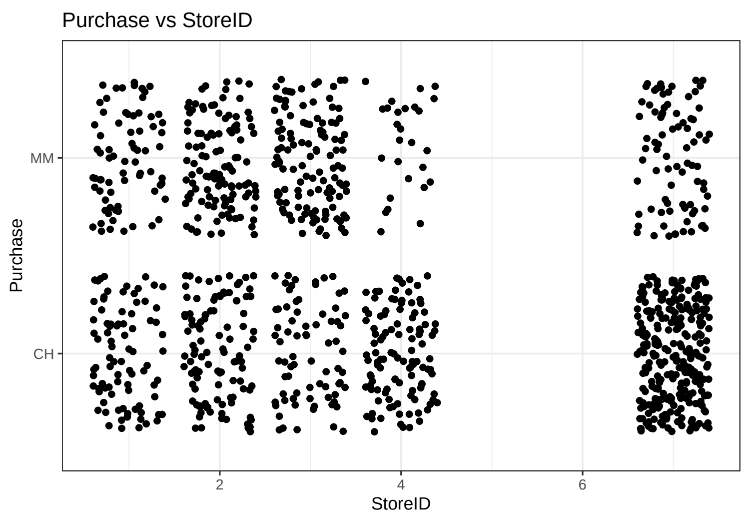


Figure 1: Purchase vs StoreID

Next, we examine the relationship between `WeekofPurchase` and `Purchase`. While Figure 2 seems to suggest that as the weeks go on, the purchases of CH increases, there is a great deal of uncertainty as to why this trend exists. Since we are unclear about external factors and whether or not there is a cyclical relationship (i.e. if we go further into time the purchases may decrease), we will keep a close eye on how the variable may be used in the future.

Figure 2: Purchase vs Week of Purchase

EDA

Lastly, we review a summary of all the `OJ` variables. We first note that there are more purchases of CH than of MM (653 vs 417). Next, we observe that the price of MM is a little bit higher, but not substantially. We can also note that it seems that `DiscCH` is a fixed value (the max of 0.5), while `DiscMM` has more of a range. Lastly, Store 7 appears 356 times, while the other 4 stores combined appear 714 times.

```
##  Purchase WeekofPurchase    StoreID        PriceCH        PriceMM
##  CH:653   Min.   :227    Min.   :1.00   Min.   :1.69   Min.   :1.69
##  MM:417   1st Qu.:240    1st Qu.:2.00   1st Qu.:1.79   1st Qu.:1.99
##           Median :257    Median :3.00   Median :1.86   Median :2.09
##           Mean   :254    Mean   :3.96   Mean   :1.87   Mean   :2.08
##           3rd Qu.:268    3rd Qu.:7.00   3rd Qu.:1.99   3rd Qu.:2.18
##           Max.   :278    Max.   :7.00   Max.   :2.09   Max.   :2.29
##      DiscCH          DiscMM        SpecialCH       SpecialMM
##  Min.   :0.000   Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.:0.000   1st Qu.:0.000   1st Qu.:0.000   1st Qu.:0.000
##  Median :0.000   Median :0.000   Median :0.000   Median :0.000
##  Mean   :0.052   Mean   :0.123   Mean   :0.148   Mean   :0.162
##  3rd Qu.:0.000   3rd Qu.:0.230   3rd Qu.:0.000   3rd Qu.:0.000
##  Max.   :0.500   Max.   :0.800   Max.   :1.000   Max.   :1.000
##     LoyalCH       SalePriceMM    SalePriceCH     PriceDiff      Store7
##  Min.   :0.000   Min.   :1.19   Min.   :1.39   Min.   :-0.670   No :714
##  1st Qu.:0.325   1st Qu.:1.69   1st Qu.:1.75   1st Qu.: 0.000   Yes:356
##  Median :0.600   Median :2.09   Median :1.86   Median : 0.230
##  Mean   :0.566   Mean   :1.96   Mean   :1.82   Mean   : 0.146
##  3rd Qu.:0.851   3rd Qu.:2.13   3rd Qu.:1.89   3rd Qu.: 0.320
##  Max.   :1.000   Max.   :2.29   Max.   :2.09   Max.   : 0.640
##    PctDiscMM       PctDiscCH       ListPriceDiff       STORE
##  Min.   :0.000   Min.   :0.0000   Min.   :0.000   Min.   :0.00
##  1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:0.140   1st Qu.:0.00
##  Median :0.000   Median :0.0000   Median :0.240   Median :2.00
##  Mean   :0.059   Mean   :0.0273   Mean   :0.218   Mean   :1.63
##  3rd Qu.:0.113   3rd Qu.:0.0000   3rd Qu.:0.300   3rd Qu.:3.00
##  Max.   :0.402   Max.   :0.2527   Max.   :0.440   Max.   :4.00
```

Analysis

For this analysis, we will split the data into training and testing sets. From the original sample of 1070 customer purchases, we will randomly split 800 (74.8%) observations into the training set, while the remaining 270 observations become the testing set.

Classification Tree

First, we fit a classification tree on the training data using `Purchase` as the response, while all the other variables serve as predictors.

From the summary output, we observe that only 4 of the original 17 possible predictors were chosen: `LoyalCH`, `PriceDiff`, `ListPriceDiff`, and `DiscMM`. This tree also has 8 terminal nodes, with a training error rate of 0.161.

```
##
## Classification tree:
## tree(formula = Purchase ~ ., data = training)
## Variables actually used in tree construction:
## [1] "LoyalCH"       "PriceDiff"     "ListPriceDiff" "DiscMM"
## Number of terminal nodes:  8
## Residual mean deviance:  0.763 = 604 / 792
## Misclassification error rate: 0.161 = 129 / 800
```

From the tree, we can pull insights from all of the individual notes. As an example, we select node 8:

"LoyalCH < 0.0356415 49 0 MM (0 1) *"

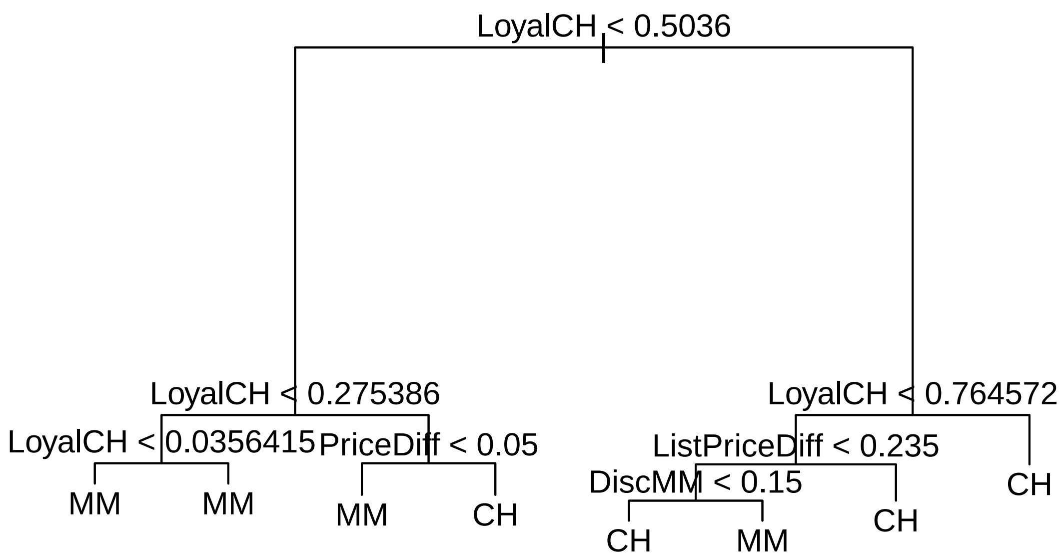
This output says that for those with LoyalCH < 0.0356, 49 out of 49 of the training observations that fit that criteria all purchased MM branded OJ. Thus, with 100% certainty, those with the specified LoyalCH values purchased MM.

Post-submission note: There seems to be an error with the output as it indicates that for the root there is a probability of 1 of being left, and 0 for right. This is obviously not true. Currently it is unclear where the source of this error comes from.

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 800 1000 CH ( 1 0 )
##    2) LoyalCH < 0.5036 341  400 MM ( 0 1 )
##      4) LoyalCH < 0.275386 157  100 MM ( 0 1 )
##        8) LoyalCH < 0.0356415 49    0 MM ( 0 1 ) *
##        9) LoyalCH > 0.0356415 108  100 MM ( 0 1 ) *
##      5) LoyalCH > 0.275386 184  300 MM ( 0 1 )
##       10) PriceDiff < 0.05 76   80 MM ( 0 1 ) *
##       11) PriceDiff > 0.05 108  100 CH ( 1 0 ) *
##    3) LoyalCH > 0.5036 459  400 CH ( 1 0 )
##      6) LoyalCH < 0.764572 193  200 CH ( 1 0 )
##       12) ListPriceDiff < 0.235 77  100 CH ( 1 0 )
##         24) DiscMM < 0.15 43   50 CH ( 1 0 ) *
##         25) DiscMM > 0.15 34   40 MM ( 0 1 ) *
##       13) ListPriceDiff > 0.235 116   80 CH ( 1 0 ) *
##      7) LoyalCH > 0.764572 266  100 CH ( 1 0 ) *
```

Next, we observe the generated plot of the tree. We observe that the first split is based off `LoyalCH`, and then both branches split on `LoyalCH` again. It seems that price difference only really matters when the buyer has a `LoyalCH` value between 0.275 and 0.764. Additionally, `DiscMM` only plays a role under the specific conditions of `LoyalCH` and `ListPriceDiff` values.

From this plot, we would conclude that `LoyalCH` is likely the most influential variable in this classification tree.



Applying the classification tree to the testing set, we observe the following predictions compared to the truth. In total, the testing error rate is 0.163, only slightly worse than the training error rate of 0.161.

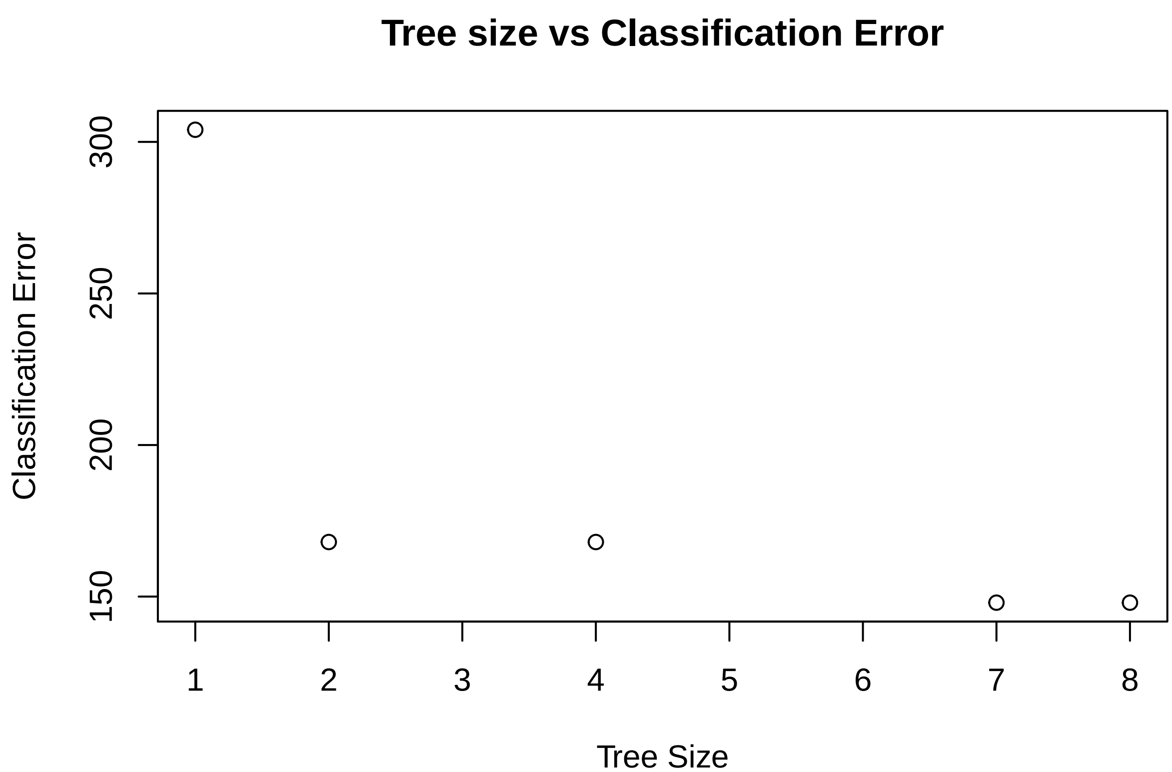
	True CH	True MM
CH	140	27
MM	17	86

With Cross Validation

Applying 10-fold cross validation, and pruning based on the classification error rate, we observe that trees of the size 8 (original) and 7 are the lowest. Since sizes 8 and 7 are equal, we proceed forward with the 7.

```
## $size
## [1] 8 7 4 2 1
##
## $dev
## [1] 148 148 168 168 304
##
## $k
## [1]   -Inf   0.00   4.67   8.00 145.00
##
## $method
## [1] "misclass"
##
## attr(,"class")
## [1] "prune"         "tree.sequence"
```

From the plot, we observe that the training error starts to level off, before becoming equal at sizes 7 and 8.



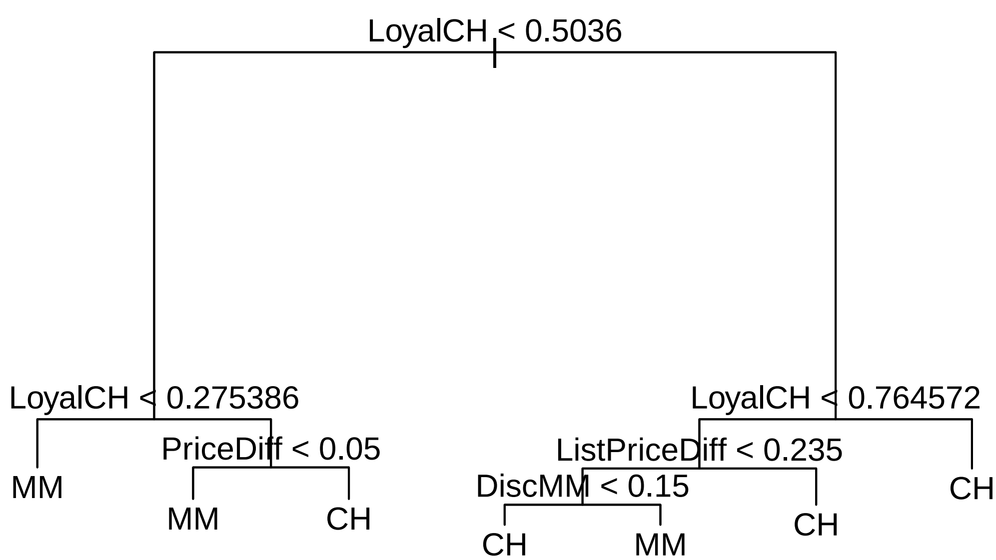
Since the optimal tree size was determined to be 7, we proceed to prune the tree with 7 as the upper bound. While the optimal tree size obtained using cross-validation is smaller, the training error is equal to the unpruned tree (0.161).

```
##
## Classification tree:
## snip.tree(tree = OJ_tree, nodes = 4L)
## Variables actually used in tree construction:
## [1] "LoyalCH"       "PriceDiff"     "ListPriceDiff" "DiscMM"
## Number of terminal nodes:  7
## Residual mean deviance:  0.783 = 621 / 793
## Misclassification error rate: 0.161 = 129 / 800
```

Examining the test error rates, the two trees result in identical results (0.163).

```
##
## OJ_prune_pred  CH  MM
##            CH 140  27
##            MM  17  86
```

To understand why these results were identical despite reducing the number of terminal nodes, we examine the tree plot. Previously, the left child of `LoyalCH < 0.275386` split further, but both still lead to the same prediction (MM). Previously, this preserved the purity of the nodes, enabling the 100% accuracy of one of the nodes, as described previously. Thus, even after reducing the total number of terminal nodes, our error rates have not changed at all.



Conclusion

In conclusion, our classification tree of Citrus Hill vs Minute Maid Orange Juice purchases had a 0.163 testing classification error rate. Using the pruned tree, there were a total of 7 terminal nodes, including 4 variables (`LoyalCH`, `PriceDiff`, `ListPriceDiff`, and `DiscMM`) out of the original 17. `LoyalCH` dominated the splits, and seems to be the most important variable for prediction.