Wine Regression

Joseph Oliveira

7/18/2020

```
library(xgboost)
```

```
## Warning: package 'xgboost' was built under R version 4.0.2
```

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
```

```
## v ggplot2 3.3.1     v purrr   0.3.4
## v tibble  3.0.1     v dplyr   0.8.5
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------------------------------------------ tidyv
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x dplyr::slice()  masks xgboost::slice()
```

```
library(tree)
```

```
## Registered S3 method overwritten by 'tree':
##   method     from
##   print.tree cli
```

```
library(leaps)
set.seed(5)
```

```
wine_train <- read_csv('../Data/Clean/wine_train.csv') %>%
  rename('fixed_acidity' = `fixed acidity`,
         'vol_acidity' = `volatile acidity`,
         'citric_acid' = `citric acid`,
         'resid_sugar' = `residual sugar`,
         'free_SO2' = `free sulfur dioxide`,
         'tot_SO2' = `total sulfur dioxide`)
```

```
## Parsed with column specification:
## cols(
##   `fixed acidity` = col_double(),
```

```
##   `volatile acidity` = col_double(),
##   `citric acid` = col_double(),
##   `residual sugar` = col_double(),
##   chlorides = col_double(),
##   `free sulfur dioxide` = col_double(),
##   `total sulfur dioxide` = col_double(),
##   density = col_double(),
##   pH = col_double(),
##   sulphates = col_double(),
##   alcohol = col_double(),
##   quality = col_double()
## )
```

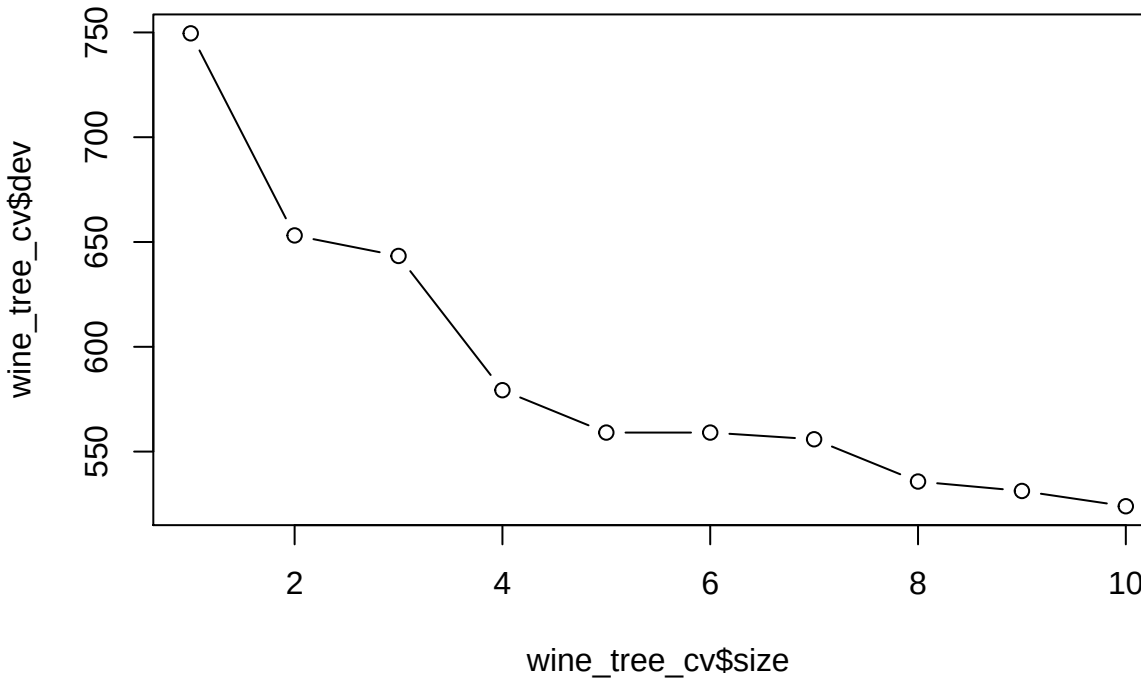
```
wine_test <- read_csv('../Data/Clean/wine_test.csv') %>%
  rename('fixed_acidity' = `fixed acidity`,
         'vol_acidity' = `volatile acidity`,
         'citric_acid' = `citric acid`,
         'resid_sugar' = `residual sugar`,
         'free_SO2' = `free sulfur dioxide`,
         'tot_SO2' = `total sulfur dioxide`)
```

```
## Parsed with column specification:
## cols(
##   `fixed acidity` = col_double(),
##   `volatile acidity` = col_double(),
##   `citric acid` = col_double(),
##   `residual sugar` = col_double(),
##   chlorides = col_double(),
##   `free sulfur dioxide` = col_double(),
##   `total sulfur dioxide` = col_double(),
##   density = col_double(),
##   pH = col_double(),
##   sulphates = col_double(),
##   alcohol = col_double(),
##   quality = col_double()
## )
```

```


wine_tree_reg <- tree(quality  ~ ., data = wine_train)
wine_tree_cv <- cv.tree(wine_tree_reg, FUN = prune.tree)

plot(wine_tree_cv$size, wine_tree_cv$dev, type = 'b')
```



tree model approach

Going with a depth of 5 max for a boosted tree approach.

```
wine_train_mat <- wine_train %>%
  select(-quality) %>%
  as.matrix()

wine_test_mat <- wine_test %>%
  select(-quality) %>%
  as.matrix()

wine_tree_bst <- xgboost(data = wine_train_mat,
                         label = wine_train$quality,
                         max.depth = 5,
                         eta = 0.1,
                         nrounds = 100,
                         objective = "reg:squarederror")
```

```
## [1]  train-rmse:4.677372
## [2]  train-rmse:4.222025
## [3]  train-rmse:3.812560
## [4]  train-rmse:3.444427
## [5]  train-rmse:3.114186
## [6]  train-rmse:2.817509
## [7]  train-rmse:2.551465
## [8]  train-rmse:2.312803
## [9]  train-rmse:2.098889
```

```
## [10] train-rmse:1.907612
## [11] train-rmse:1.735866
## [12] train-rmse:1.581522
## [13] train-rmse:1.444632
## [14] train-rmse:1.321872
## [15] train-rmse:1.213145
## [16] train-rmse:1.116301
## [17] train-rmse:1.029949
## [18] train-rmse:0.954544
## [19] train-rmse:0.888183
## [20] train-rmse:0.828362
## [21] train-rmse:0.776820
## [22] train-rmse:0.730790
## [23] train-rmse:0.691188
## [24] train-rmse:0.656928
## [25] train-rmse:0.626859
## [26] train-rmse:0.600228
## [27] train-rmse:0.576959
## [28] train-rmse:0.557509
## [29] train-rmse:0.539683
## [30] train-rmse:0.524482
## [31] train-rmse:0.511067
## [32] train-rmse:0.499528
## [33] train-rmse:0.489971
## [34] train-rmse:0.479262
## [35] train-rmse:0.470409
## [36] train-rmse:0.461915
## [37] train-rmse:0.453186
## [38] train-rmse:0.448320
## [39] train-rmse:0.443716
## [40] train-rmse:0.438467
## [41] train-rmse:0.434411
## [42] train-rmse:0.431530
## [43] train-rmse:0.427217
## [44] train-rmse:0.425220
## [45] train-rmse:0.422713
## [46] train-rmse:0.420542
## [47] train-rmse:0.417894
## [48] train-rmse:0.413100
## [49] train-rmse:0.410825
## [50] train-rmse:0.408943
## [51] train-rmse:0.405777
## [52] train-rmse:0.404558
## [53] train-rmse:0.402211
## [54] train-rmse:0.401315
## [55] train-rmse:0.397159
## [56] train-rmse:0.394969
## [57] train-rmse:0.390818
## [58] train-rmse:0.389417
## [59] train-rmse:0.387821
## [60] train-rmse:0.386834
## [61] train-rmse:0.385838
## [62] train-rmse:0.385171
## [63] train-rmse:0.383362
```

```
## [64] train-rmse:0.379286
## [65] train-rmse:0.375219
## [66] train-rmse:0.374376
## [67] train-rmse:0.373843
## [68] train-rmse:0.370702
## [69] train-rmse:0.367589
## [70] train-rmse:0.366630
## [71] train-rmse:0.363127
## [72] train-rmse:0.359647
## [73] train-rmse:0.355487
## [74] train-rmse:0.354878
## [75] train-rmse:0.353526
## [76] train-rmse:0.352192
## [77] train-rmse:0.351169
## [78] train-rmse:0.349534
## [79] train-rmse:0.348337
## [80] train-rmse:0.344451
## [81] train-rmse:0.341894
## [82] train-rmse:0.340472
## [83] train-rmse:0.336778
## [84] train-rmse:0.336143
## [85] train-rmse:0.333529
## [86] train-rmse:0.329601
## [87] train-rmse:0.328695
## [88] train-rmse:0.326914
## [89] train-rmse:0.324224
## [90] train-rmse:0.323312
## [91] train-rmse:0.322194
## [92] train-rmse:0.320728
## [93] train-rmse:0.319350
## [94] train-rmse:0.316764
## [95] train-rmse:0.315686
## [96] train-rmse:0.312247
## [97] train-rmse:0.311190
## [98] train-rmse:0.309766
## [99] train-rmse:0.307191
## [100]    train-rmse:0.306019
```

```
pred_bst <- predict(wine_tree_bst, wine_test_mat)

mse_bst <- mean((pred_bst - wine_test$quality)^2)
misclas_bst <- 1 - mean(round(pred_bst) == wine_test$quality)
```

Boosted model test MSE: 0.3746187 Rounding the predicted scores, the boosted misclassification rate: 0.3333333

```
wine_lm <- lm(quality ~ ., data = wine_train)
pred_lm <- predict(wine_lm, select(wine_test, -quality))
mse_lm  <- mean((pred_lm - wine_test$quality)^2)
misclas_lm <- 1 - mean(round(pred_lm) == wine_test$quality)
```

linear regression LM model test MSE: 0.3968936 Rounding the predicted scores, the linear model misclassification rate: 0.4083333

```
wine_lm_full <- regsubsets(quality ~ ., data = wine_train, nvmax = 10)
summary(wine_lm_full)
```

```
## Subset selection object
## Call: regsubsets.formula(quality ~ ., data = wine_train, nvmax = 10)
## 11 Variables  (and intercept)
##               Forced in Forced out
## fixed_acidity     FALSE      FALSE
## vol_acidity       FALSE      FALSE
## citric_acid       FALSE      FALSE
## resid_sugar       FALSE      FALSE
## chlorides         FALSE      FALSE
## free_SO2          FALSE      FALSE
## tot_SO2           FALSE      FALSE
## density           FALSE      FALSE
## pH                FALSE      FALSE
## sulphates         FALSE      FALSE
## alcohol           FALSE      FALSE
## 1 subsets of each size up to 10
## Selection Algorithm: exhaustive
##           fixed_acidity vol_acidity citric_acid resid_sugar chlorides free_SO2
## 1  ( 1 )  " "           " "          " "         " "         " "       " "
## 2  ( 1 )  " "           "*"          " "         " "         " "       " "
## 3  ( 1 )  " "           "*"          " "         " "         " "       " "
## 4  ( 1 )  " "           "*"          " "         " "         " "       " "
## 5  ( 1 )  " "           "*"          " "         " "         "*"       " "
## 6  ( 1 )  " "           "*"          " "         " "         "*"       " "
## 7  ( 1 )  " "           "*"          " "         " "         "*"       "*"
## 8  ( 1 )  " "           "*"          " "         " "         "*"       "*"
## 9  ( 1 )  " "           "*"          "*"         " "         "*"       "*"
## 10  ( 1 ) "*"           "*"          "*"         " "         "*"       "*"
##           tot_SO2 density pH  sulphates alcohol
## 1  ( 1 )  " "     " "     " " " "       "*"
## 2  ( 1 )  " "     " "     " " " "       "*"
## 3  ( 1 )  " "     " "     " " "*"       "*"
## 4  ( 1 )  "*"     " "     " " "*"       "*"
## 5  ( 1 )  "*"     " "     " " "*"       "*"
## 6  ( 1 )  "*"     " "     "*" "*"       "*"
## 7  ( 1 )  "*"     " "     "*" "*"       "*"
## 8  ( 1 )  "*"     "*"     "*" "*"       "*"
## 9  ( 1 )  "*"     "*"     "*" "*"       "*"
## 10  ( 1 ) "*"     "*"     "*" "*"       "*"
```

Among the variables, `alcohol`, `vol_acidity`, `sulphates`, `total sulphur dioxide`, and `chlorides` are the top predictors in the selection process.

```
wine_lm_fwd <- regsubsets(quality ~ ., data = wine_train, nvmax = 5, method = "forward")
summary(wine_lm_fwd)
```

```
## Subset selection object
```

```
## Call: regsubsets.formula(quality ~ ., data = wine_train, nvmax = 5,
##     method = "forward")
## 11 Variables  (and intercept)
##               Forced in Forced out
## fixed_acidity     FALSE      FALSE
## vol_acidity       FALSE      FALSE
## citric_acid       FALSE      FALSE
## resid_sugar       FALSE      FALSE
## chlorides         FALSE      FALSE
## free_SO2          FALSE      FALSE
## tot_SO2           FALSE      FALSE
## density           FALSE      FALSE
## pH                FALSE      FALSE
## sulphates         FALSE      FALSE
## alcohol           FALSE      FALSE
## 1 subsets of each size up to 5
## Selection Algorithm: forward
##          fixed_acidity vol_acidity citric_acid resid_sugar chlorides free_SO2
## 1  ( 1 ) " "           " "         " "         " "         " "       " "
## 2  ( 1 ) " "           "*"         " "         " "         " "       " "
## 3  ( 1 ) " "           "*"         " "         " "         " "       " "
## 4  ( 1 ) " "           "*"         " "         " "         " "       " "
## 5  ( 1 ) " "           "*"         " "         " "         "*"       " "
##          tot_SO2 density pH  sulphates alcohol
## 1  ( 1 ) " "     " "     " " " "       "*"
## 2  ( 1 ) " "     " "     " " " "       "*"
## 3  ( 1 ) " "     " "     " " "*"       "*"
## 4  ( 1 ) "*"     " "     " " "*"       "*"
## 5  ( 1 ) "*"     " "     " " "*"       "*"
```

The same 5 variables are selected in forward selection and backward selection.

```
wine_lm_bwd <- regsubsets(quality ~ ., data = wine_train, nvmax = 5, method = "backward")
summary(wine_lm_bwd)
```

```
## Subset selection object
## Call: regsubsets.formula(quality ~ ., data = wine_train, nvmax = 5,
##     method = "backward")
## 11 Variables  (and intercept)
##               Forced in Forced out
## fixed_acidity     FALSE      FALSE
## vol_acidity       FALSE      FALSE
## citric_acid       FALSE      FALSE
## resid_sugar       FALSE      FALSE
## chlorides         FALSE      FALSE
## free_SO2          FALSE      FALSE
## tot_SO2           FALSE      FALSE
## density           FALSE      FALSE
## pH                FALSE      FALSE
## sulphates         FALSE      FALSE
## alcohol           FALSE      FALSE
## 1 subsets of each size up to 5
## Selection Algorithm: backward
```

```
##          fixed_acidity vol_acidity citric_acid resid_sugar chlorides free_SO2
## 1  ( 1 ) " "           " "          " "         " "         " "       " "
## 2  ( 1 ) " "           "*"          " "         " "         " "       " "
## 3  ( 1 ) " "           "*"          " "         " "         " "       " "
## 4  ( 1 ) " "           "*"          " "         " "         " "       " "
## 5  ( 1 ) " "           "*"          " "         " "         "*"       " "
##          tot_SO2 density pH  sulphates alcohol
## 1  ( 1 ) " "     " "     " " " "       "*"
## 2  ( 1 ) " "     " "     " " " "       "*"
## 3  ( 1 ) " "     " "     " " "*"       "*"
## 4  ( 1 ) "*"     " "     " " "*"       "*"
## 5  ( 1 ) "*"     " "     " " "*"       "*"
```

```
wine_lm_five <- lm(quality ~ alcohol + vol_acidity +
                sulphates + tot_SO2 + chlorides, data = wine_train)
pred_lm_five <- predict(wine_lm_five, select(wine_test, -quality))
mse_lm_five  <- mean((pred_lm_five - wine_test$quality)^2)
misclas_lm_five <- 1 - mean(round(pred_lm_five) == wine_test$quality)
```

Boosted model test MSE: 0.4021208 Rounding the predicted scores, the boosted misclassification rate: 0.4166667