LogitModel.RMD

Bart Ciastkowski

12/2/2019

# The Code

### Adding libraries

library("plyr")  
library("conflicted")  
library("tidyverse")

## ── Attaching packages ──────────────────────────────────────────────────────────────────────────────────────── tidyverse 1.3.0 ──

## ✔ ggplot2 3.2.1 ✔ purrr 0.3.3  
## ✔ tibble 2.1.3 ✔ dplyr 0.8.3  
## ✔ tidyr 1.0.0 ✔ stringr 1.4.0  
## ✔ readr 1.3.1 ✔ forcats 0.4.0

library("here")

## here() starts at /Users/garretthawes/wine-project

library("ggthemes")  
library("stringr")  
library("stringi")  
library("readxl")  
library("ggExtra")  
library("PerformanceAnalytics")

## Loading required package: xts

## Loading required package: zoo

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

## Registered S3 method overwritten by 'xts':  
## method from  
## as.zoo.xts zoo

##   
## Attaching package: 'xts'

## The following objects are masked from 'package:dplyr':  
##   
## first, last

library("plotROC")  
library('sentimentr')

### Resolving library conflicts

conflict\_prefer("mutate", "dplyr")

## [conflicted] Will prefer dplyr::mutate over any other package

conflict\_prefer("margin","ggplot2")

## [conflicted] Will prefer ggplot2::margin over any other package

### Clear all objects including hidden objects.

rm(list = ls(all.names = TRUE))

### Load Model and Train Data

load(here::here("data","output","limited\_factors","wine\_train.RData"))  
load(here::here("data","output","limited\_factors","wine\_test.RData"))  
names(wine\_train)

## [1] "price" "points" "points.category"   
## [4] "country" "province" "winery"   
## [7] "color" "variety" "variety\_and\_color"   
## [10] "designation" "title.n\_words" "title.sentement"   
## [13] "title.n\_chars" "title.has\_accents" "taster.name"   
## [16] "taster.twitter\_handle" "taster.gender" "taster.avg\_points"   
## [19] "taster.n\_reviews" "taster.n\_tweets" "taster.n\_followers"

names(wine\_test)

## [1] "price" "points" "points.category"   
## [4] "country" "province" "winery"   
## [7] "color" "variety" "variety\_and\_color"   
## [10] "designation" "title.n\_words" "title.sentement"   
## [13] "title.n\_chars" "title.has\_accents" "taster.name"   
## [16] "taster.twitter\_handle" "taster.gender" "taster.avg\_points"   
## [19] "taster.n\_reviews" "taster.n\_tweets" "taster.n\_followers"

# Determining Good Value (i.e. well\_priced)

### Need to consider diminishing returns, as the rate of marginal increase in points rating decreases with respect to the increase in the price of a bottle of wine, hence using log(price of wine):

##### Well\_priced == whether we think wine is well priced

##### Well\_priced will be determined according to a median price to points ratio computed as follows using observable from the training data set

##### median\_price\_to\_points\_ratio= Dataset ratio numerator / Dataset ratio denominator

##### Dataset ratio numerator = Median (points awarded) - Min (points awarded)

##### Dataset ratio denominator = Median ( log ( price of wine)) - Min (log (expected(min price of drinkable wine))

##### Where expected(min price of drinkable wine) = $2.5 (i.e. educated guess)

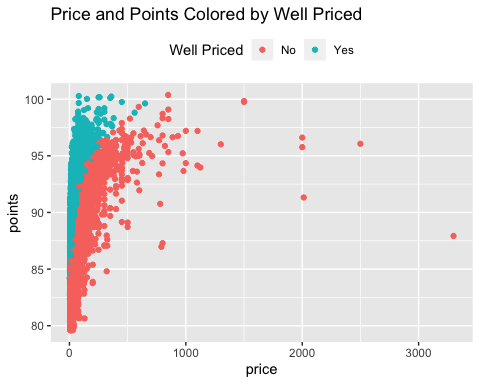
##### Where price of wine > expected (min price of drinkable wine)

### In Summary: the above formula takes into consideration diminishing returns i.e. marginal increase in points is accompanied by a higher and higher increase in price and assumes the lowest price for acceptable bottle of wine to be $2.5 (other numbers could be used in the future)

wine\_train.pt.min <- min(wine\_train$points)   
price.min <- 2.5  
ratio.numerator <- median(wine\_train$points)-wine\_train.pt.min  
ratio.denominator <- ifelse(wine\_train$price<price.min, price.min + 0.1, wine\_train$price)  
ratio.denominator <- log(ratio.denominator)  
ratio.denominator <- median(ratio.denominator)  
ratio.denominator <- ratio.denominator - log(price.min)  
median\_price\_to\_points\_ratio <- ratio.numerator/ratio.denominator  
.is\_well\_priced <- function(df){  
 test.LHS.numerator <- df$points-wine\_train.pt.min  
 test.LHS.denominator <- ifelse(df$price<price.min, price.min + 0.1,df$price)  
 test.LHS.denominator <- log(test.LHS.denominator)  
 test.LHS.denominator <- test.LHS.denominator-log(price.min)  
 test.LHS <- test.LHS.numerator/test.LHS.denominator  
 test.RHS <- median\_price\_to\_points\_ratio  
 well\_priced <- factor(test.LHS > test.RHS, labels=c("No", "Yes"))  
 return(well\_priced)  
}  
# Compute well\_priced for train ---- same formula as for median\_price\_to\_points\_ratio, except for an individual price point combination  
wine\_train\_logit <- wine\_train %>%   
 dplyr::mutate ( well\_priced = .is\_well\_priced(.) )

## The following chart describes that relationship:

ggplot(wine\_train\_logit , aes(x = price, y = points, color = well\_priced)) +  
 geom\_jitter() +  
 theme(legend.position = "top") +   
 labs(title="Price and Points Colored by Well Priced",   
 color = "Well Priced")



head(wine\_train)

## # A tibble: 6 x 21  
## price points points.category country province winery color variety  
## <dbl> <int> <fct> <fct> <fct> <fct> <fct> <fct>   
## 1 9 87 Very good Spain Norther… Other Red Other   
## 2 11 86 Very good US Washing… Other White Other   
## 3 14 90 Outstanding Other Other Other White Chardo…  
## 4 17 85 Very good Italy Northea… Other White Other   
## 5 45 94 Outstanding US Washing… Other Red Bordea…  
## 6 15 83 Good France Other Other White Chardo…  
## # … with 13 more variables: variety\_and\_color <fct>, designation <fct>,  
## # title.n\_words <int>, title.sentement <dbl>, title.n\_chars <int>,  
## # title.has\_accents <dbl>, taster.name <fct>, taster.twitter\_handle <fct>,  
## # taster.gender <fct>, taster.avg\_points <dbl>, taster.n\_reviews <int>,  
## # taster.n\_tweets <int>, taster.n\_followers <int>

### “Good Value” is about price, so we don’t want it to be part of the model + removing points\_category as using points

wine\_train\_logit <- wine\_train\_logit %>%   
 dplyr::select(-price,-points.category)  
# Compute well\_priced for test dataset and remove price and points  
wine\_test\_logit <- wine\_test %>%   
 dplyr::mutate ( well\_priced = .is\_well\_priced(.) ) %>%   
 select (-price,-points.category)

# Let’s create the model —-

### Since well\_priced is a binomial estimate of price, price is removed from the dataset (train and test) before model is created

### taster.twitter\_handle and variety\_and\_color are also removed as they provide no added value (i.e. NA values)

### Otherwise same variables are used for the model so that it can be compared against other models

logit\_mod <- glm( well\_priced ~ .,  
 data = wine\_train\_logit %>%   
 select (  
 -taster.twitter\_handle,  
 -variety\_and\_color,   
 ),  
 family = binomial) #our varaible can be 0 or 1, a binomial  
# summary of model ----  
summary(logit\_mod)

##   
## Call:  
## glm(formula = well\_priced ~ ., family = binomial, data = wine\_train\_logit %>%   
## select(-taster.twitter\_handle, -variety\_and\_color, ))  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.4473 -0.6983 0.2644 0.7110 3.1439   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.957e+01 1.692e+01 -1.156 0.247569   
## points 6.732e-01 5.510e-03 122.174 < 2e-16 \*\*\*  
## countryAustralia 5.740e-01 2.823e-01 2.033 0.042017 \*   
## countryAustria -2.395e-01 2.710e-01 -0.884 0.376897   
## countryCanada -2.078e+00 3.391e-01 -6.127 8.98e-10 \*\*\*  
## countryChile 6.025e-01 1.405e-01 4.287 1.81e-05 \*\*\*  
## countryFrance 2.118e-01 2.559e-01 0.828 0.407875   
## countryGermany -1.410e+00 3.096e-01 -4.553 5.29e-06 \*\*\*  
## countryGreece -3.843e-01 3.075e-01 -1.250 0.211420   
## countryIsrael -1.627e+00 2.852e-01 -5.706 1.16e-08 \*\*\*  
## countryItaly 3.338e-01 2.882e-01 1.158 0.246886   
## countryNew Zealand 3.570e-01 2.744e-01 1.301 0.193145   
## countryPortugal 4.829e-01 2.636e-01 1.832 0.066998 .   
## countrySouth Africa 1.803e-01 2.679e-01 0.673 0.500896   
## countrySpain -4.776e-02 1.451e-01 -0.329 0.742091   
## countryUS -7.373e-01 2.900e-01 -2.543 0.010999 \*   
## countryOther -1.044e+00 2.267e-01 -4.603 4.16e-06 \*\*\*  
## provinceBordeaux 1.115e+00 1.223e-01 9.110 < 2e-16 \*\*\*  
## provinceBurgundy -1.552e+00 1.236e-01 -12.560 < 2e-16 \*\*\*  
## provinceCalifornia 1.564e+00 3.113e-01 5.024 5.07e-07 \*\*\*  
## provinceChampagne -2.494e+00 1.462e-01 -17.060 < 2e-16 \*\*\*  
## provinceLoire Valley 5.688e-01 1.288e-01 4.418 9.98e-06 \*\*\*  
## provinceMendoza Province 1.051e+00 1.860e-01 5.648 1.62e-08 \*\*\*  
## provinceNortheastern Italy 6.114e-01 1.520e-01 4.021 5.80e-05 \*\*\*  
## provinceNorthern Spain 6.421e-01 1.445e-01 4.444 8.82e-06 \*\*\*  
## provinceOregon -9.089e-02 2.072e-01 -0.439 0.660986   
## provincePiedmont 9.095e-01 1.726e-01 5.270 1.36e-07 \*\*\*  
## provinceSouth Australia 5.452e-01 1.842e-01 2.960 0.003075 \*\*   
## provinceSouthwest France 1.748e+00 1.361e-01 12.846 < 2e-16 \*\*\*  
## provinceTuscany 8.295e-01 1.445e-01 5.739 9.53e-09 \*\*\*  
## provinceWashington 4.814e-01 2.030e-01 2.372 0.017701 \*   
## provinceOther 1.000e+00 1.195e-01 8.372 < 2e-16 \*\*\*  
## wineryChehalem -1.068e+00 3.671e-01 -2.910 0.003612 \*\*   
## wineryColumbia Crest 2.412e-01 3.245e-01 0.743 0.457239   
## wineryConcha y Toro -1.501e+00 3.325e-01 -4.515 6.34e-06 \*\*\*  
## wineryDFJ Vinhos 5.417e-01 3.331e-01 1.626 0.103918   
## wineryGeorges Duboeuf -3.521e-01 3.690e-01 -0.954 0.340029   
## wineryJean-Luc and Paul Aegerter -2.500e+00 4.494e-01 -5.563 2.65e-08 \*\*\*  
## wineryLouis Latour -3.407e+00 3.855e-01 -8.838 < 2e-16 \*\*\*  
## wineryMaryhill -1.639e+00 4.059e-01 -4.037 5.41e-05 \*\*\*  
## wineryMontes -1.541e+00 3.594e-01 -4.289 1.79e-05 \*\*\*  
## winerySanta Ema -1.322e+00 3.812e-01 -3.468 0.000525 \*\*\*  
## wineryTestarossa -2.050e+00 3.591e-01 -5.710 1.13e-08 \*\*\*  
## wineryTrapiche -5.488e-01 3.828e-01 -1.434 0.151699   
## wineryUndurraga -1.012e+00 3.868e-01 -2.616 0.008897 \*\*   
## wineryWines & Winemakers -7.428e-01 3.253e-01 -2.284 0.022386 \*   
## wineryOther -9.826e-01 2.390e-01 -4.111 3.94e-05 \*\*\*  
## colorWhite 8.806e-01 3.872e-02 22.742 < 2e-16 \*\*\*  
## colorOther 2.264e-01 7.326e-02 3.090 0.002000 \*\*   
## varietyPinot Noir -1.742e+00 1.063e-01 -16.394 < 2e-16 \*\*\*  
## varietyCabernet Sauvignon -1.780e+00 1.084e-01 -16.426 < 2e-16 \*\*\*  
## varietyChardonnay -1.546e+00 1.159e-01 -13.340 < 2e-16 \*\*\*  
## varietyMalbec -1.542e+00 1.219e-01 -12.647 < 2e-16 \*\*\*  
## varietyMerlot -1.205e+00 1.197e-01 -10.062 < 2e-16 \*\*\*  
## varietyRed Blend -1.521e+00 1.059e-01 -14.361 < 2e-16 \*\*\*  
## varietySangiovese -2.300e+00 1.332e-01 -17.262 < 2e-16 \*\*\*  
## varietySauvignon Blanc -1.186e+00 1.207e-01 -9.831 < 2e-16 \*\*\*  
## varietyBordeaux-style Red Blend -2.085e+00 1.164e-01 -17.903 < 2e-16 \*\*\*  
## varietyRosé -7.854e-01 1.345e-01 -5.839 5.24e-09 \*\*\*  
## varietySyrah -1.580e+00 1.143e-01 -13.825 < 2e-16 \*\*\*  
## varietyRiesling -8.517e-01 1.344e-01 -6.339 2.31e-10 \*\*\*  
## varietyNebbiolo -3.094e+00 1.625e-01 -19.042 < 2e-16 \*\*\*  
## varietyTempranillo -1.227e+00 1.281e-01 -9.581 < 2e-16 \*\*\*  
## varietyOther -1.240e+00 1.006e-01 -12.318 < 2e-16 \*\*\*  
## designationBrut -2.320e-01 1.105e-01 -2.100 0.035702 \*   
## designationClassic Vintage -1.943e-01 1.245e-01 -1.560 0.118736   
## designationDry -4.957e-01 1.527e-01 -3.247 0.001166 \*\*   
## designationEstate -4.113e-01 8.456e-02 -4.864 1.15e-06 \*\*\*  
## designationNo Designation 1.256e-01 7.450e-02 1.686 0.091747 .   
## designationOld Vine 2.096e-01 1.246e-01 1.683 0.092442 .   
## designationPremium -3.965e-01 1.414e-01 -2.803 0.005060 \*\*   
## designationRed 1.258e-01 1.208e-01 1.041 0.297858   
## designationReserve -3.391e-01 7.823e-02 -4.334 1.46e-05 \*\*\*  
## designationRose 3.262e-01 1.269e-01 2.570 0.010167 \*   
## designationSignature -1.013e-01 1.008e-01 -1.005 0.315047   
## designationSingle Vineyard -1.245e+00 1.448e-01 -8.599 < 2e-16 \*\*\*  
## designationSome Vineyard -5.295e-01 7.988e-02 -6.629 3.38e-11 \*\*\*  
## designationWhite -3.864e-01 1.234e-01 -3.131 0.001743 \*\*   
## designationOther -2.905e-01 7.240e-02 -4.012 6.02e-05 \*\*\*  
## title.n\_words 1.758e-02 1.310e-02 1.342 0.179732   
## title.sentement -1.394e-01 9.532e-02 -1.462 0.143706   
## title.n\_chars -6.997e-03 2.053e-03 -3.408 0.000655 \*\*\*  
## title.has\_accents 5.562e-02 3.015e-02 1.845 0.065063 .   
## taster.namePaul Gregutt 3.195e+01 1.011e+01 3.160 0.001575 \*\*   
## taster.nameMichael Schachner 1.201e+01 3.749e+00 3.202 0.001363 \*\*   
## taster.nameKerin O’Keefe 4.105e+01 1.389e+01 2.955 0.003124 \*\*   
## taster.nameVirginie Boone 3.455e+01 1.102e+01 3.135 0.001716 \*\*   
## taster.nameMatt Kettmann 3.806e+01 1.233e+01 3.087 0.002022 \*\*   
## taster.nameSean P. Sullivan 3.900e+01 1.183e+01 3.296 0.000980 \*\*\*  
## taster.nameJim Gordon 3.761e+01 1.141e+01 3.295 0.000983 \*\*\*  
## taster.nameJoe Czerwinski 4.287e+01 1.430e+01 2.997 0.002724 \*\*   
## taster.nameAnne Krebiehl MW 7.270e+01 2.491e+01 2.918 0.003520 \*\*   
## taster.nameOther 5.269e+01 1.625e+01 3.243 0.001182 \*\*   
## taster.genderM 6.389e+00 1.599e+00 3.995 6.48e-05 \*\*\*  
## taster.avg\_points -1.037e+00 3.377e-01 -3.070 0.002142 \*\*   
## taster.n\_reviews 2.439e-03 7.686e-04 3.174 0.001505 \*\*   
## taster.n\_tweets 1.197e-03 6.149e-04 1.947 0.051497 .   
## taster.n\_followers -2.453e-03 1.158e-03 -2.119 0.034110 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 93660 on 68624 degrees of freedom  
## Residual deviance: 61632 on 68527 degrees of freedom  
## AIC: 61828  
##   
## Number of Fisher Scoring iterations: 5

## Computing predictions

### Train test set

preds.train<- data.frame (  
 pred = predict(logit\_mod, type="response"),  
 wine\_train\_logit  
 #actual = wine\_train\_logit$well\_priced  
)   
#head(preds.train)

### Test set

preds.test<- data.frame (  
 pred = predict(logit\_mod,   
 newdata=wine\_test\_logit,   
 type="response"),  
 wine\_test\_logit  
 #actual = wine\_test\_logit$well\_priced  
)   
#head(preds.test)

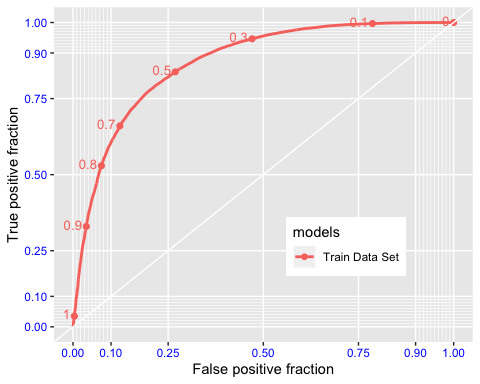
## ROC Curve

TrainDF <- data.frame(default = c(preds.train$well\_priced),  
 scores = c(preds.train$pred),  
 models = c(rep("Train Data Set",length(preds.train$pred))))  
#summary(TrainDF)  
TestDF <- data.frame(default = c(preds.test$well\_priced),  
 scores = c(preds.test$pred),  
 models = c(rep("Test Data Set",length(preds.test$pred))))

### ROC Curve train

TrainROC <- ggplot(TrainDF, aes(m = scores, d = default, color = models)) +   
 geom\_roc(show.legend = TRUE, labelsize = 3.5, cutoffs.at = c(.99,.9,.8,.7,.5,.3,.1,0))  
TrainROC <- TrainROC + style\_roc(theme = theme\_grey) +  
 theme(axis.text = element\_text(colour = "blue")) +  
 theme(legend.justification = c(1, 0),   
 legend.position = c(1, 0),  
 legend.box.margin=margin(c(50,50,50,50)))  
plot(TrainROC)

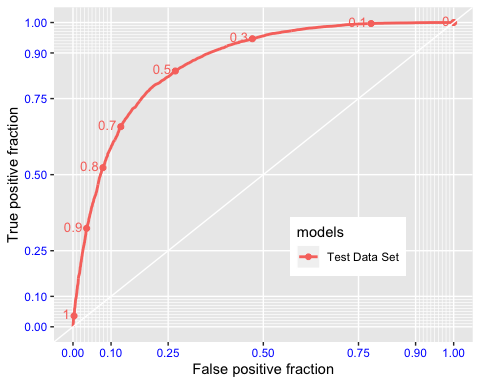
## Warning in verify\_d(data$d): D not labeled 0/1, assuming 1 = 0 and 2 = 1!



### ROC Curve test

TestROC <- ggplot(TestDF, aes(m = scores, d = default, color = models)) +   
 geom\_roc(show.legend = TRUE, labelsize = 3.5, cutoffs.at = c(.99,.9,.8,.7,.5,.3,.1,0))  
TestROC <- TestROC + style\_roc(theme = theme\_grey) +  
 theme(axis.text = element\_text(colour = "blue")) +  
 theme(legend.justification = c(1, 0),   
 legend.position = c(1, 0),  
 legend.box.margin=margin(c(50,50,50,50)))  
plot(TestROC)

## Warning in verify\_d(data$d): D not labeled 0/1, assuming 1 = 0 and 2 = 1!



### Area under the curve

AUC\_results <- data.frame (  
 TrainAUC = calc\_auc(TrainROC),  
 TestAUC =calc\_auc(TestROC)  
)

## Warning in verify\_d(data$d): D not labeled 0/1, assuming 1 = 0 and 2 = 1!  
  
## Warning in verify\_d(data$d): D not labeled 0/1, assuming 1 = 0 and 2 = 1!

AUC\_results

## TrainAUC.PANEL TrainAUC.group TrainAUC.AUC TestAUC.PANEL TestAUC.group  
## 1 1 1 0.8682864 1 1  
## TestAUC.AUC  
## 1 0.8665138

# Logit Model Summary

## Both train and test set curves are above the diagonal chance-only line

## This means that determining whether wine is a “good value” is better than chance

## AUC values are high, above 80% or about 70% higher than chance

## AUC for train and test are nearly identical (0.868 vs 0.867) hence model neither over- or underfit

## Points, Specific Province, Specific Variety, Specific Winery, followed by Specific Taster explain the model best whereas Title informat