Module 4 CT Option 1

Shaun Stearns

8/10/2019

# Libraries and initial data exploration

library(gains)  
library(caret)

## Loading required package: lattice

## Loading required package: ggplot2

setwd("~/Desktop/CSU Global Data Analytics/MIS510/Module 4/")  
bank.df <- read.csv("banks.csv")  
dim(bank.df)

## [1] 20 5

t(t(names(bank.df)))

## [,1]   
## [1,] "Obs"   
## [2,] "Financial.Condition"  
## [3,] "TotCap.Assets"   
## [4,] "TotExp.Assets"   
## [5,] "TotLns.Lses.Assets"

head(bank.df)

## Obs Financial.Condition TotCap.Assets TotExp.Assets TotLns.Lses.Assets  
## 1 1 1 9.7 0.12 0.65  
## 2 2 1 1.0 0.11 0.62  
## 3 3 1 6.9 0.09 1.02  
## 4 4 1 5.8 0.10 0.67  
## 5 5 1 4.3 0.11 0.69  
## 6 6 1 9.1 0.13 0.74

bank.df <- bank.df[ , -c(1)] # Drop ID column.

# Check that ID column was removed

t(t(names(bank.df)))

## [,1]   
## [1,] "Financial.Condition"  
## [2,] "TotCap.Assets"   
## [3,] "TotExp.Assets"   
## [4,] "TotLns.Lses.Assets"

# Treat Financial.Condition as categorical (R will create dummy variables)

bank.df$Financial.Condition <- factor(bank.df$Financial.Condition, levels = c(1, 0),  
 labels = c("Strong", "Weak"))  
head(bank.df)

## Financial.Condition TotCap.Assets TotExp.Assets TotLns.Lses.Assets  
## 1 Strong 9.7 0.12 0.65  
## 2 Strong 1.0 0.11 0.62  
## 3 Strong 6.9 0.09 1.02  
## 4 Strong 5.8 0.10 0.67  
## 5 Strong 4.3 0.11 0.69  
## 6 Strong 9.1 0.13 0.74

# glm() (general linear model) with family = “binomial” to fit a logistic regression.

### Max iteration added for convergence and getting actual p-values

logit.reg <- glm(Financial.Condition ~ ., data = bank.df, family = "binomial",   
 control=glm.control(maxit = 1))

## Warning: glm.fit: algorithm did not converge

options(scipen=999)  
summary(logit.reg)

##   
## Call:  
## glm(formula = Financial.Condition ~ ., family = "binomial", data = bank.df,   
## control = glm.control(maxit = 1))  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.07839 -0.57755 0.00022 0.64741 1.26064   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 6.1551 3.2687 1.883 0.0597 .  
## TotCap.Assets 0.1972 0.1164 1.695 0.0901 .  
## TotExp.Assets -41.7091 22.0461 -1.892 0.0585 .  
## TotLns.Lses.Assets -5.7832 3.4328 -1.685 0.0920 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 27.726 on 19 degrees of freedom  
## Residual deviance: 10.216 on 16 degrees of freedom  
## AIC: 18.216  
##   
## Number of Fisher Scoring iterations: 1

# Multiple regression model

### Values do not match logistic regression. Multiple regression is not appropriate for binary variables.

bank.df <- read.csv("banks.csv")  
bank.df <- bank.df[ , -c(1)] # Drop ID column.  
Mult.Reg <- lm(Financial.Condition ~ ., data = bank.df)  
summary(Mult.Reg)

##   
## Call:  
## lm(formula = Financial.Condition ~ ., data = bank.df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.53980 -0.23271 0.09236 0.18962 0.45119   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.76546 0.44219 -1.731 0.1027   
## TotCap.Assets -0.04055 0.01574 -2.576 0.0203 \*  
## TotExp.Assets 8.57525 2.98243 2.875 0.0110 \*  
## TotLns.Lses.Assets 1.18901 0.46440 2.560 0.0210 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3124 on 16 degrees of freedom  
## Multiple R-squared: 0.6877, Adjusted R-squared: 0.6291   
## F-statistic: 11.74 on 3 and 16 DF, p-value: 0.0002569

# Lostic Regression Model with coefficients and odds

### Max iteration added for convergence and getting actual p-values

bank.df$Financial.Condition <- factor(bank.df$Financial.Condition, levels = c(1, 0),  
 labels = c("Strong", "Weak"))  
logit.reg <- glm(Financial.Condition ~ ., data = bank.df, family = "binomial",   
 control=glm.control(maxit=1))

## Warning: glm.fit: algorithm did not converge

data.frame(summary(logit.reg)$coefficients, odds = exp(coef(logit.reg)))

## Estimate Std..Error z.value Pr...z..  
## (Intercept) 6.1550797 3.2687019 1.883035 0.05969564  
## TotCap.Assets 0.1972402 0.1163806 1.694786 0.09011601  
## TotExp.Assets -41.7090601 22.0461160 -1.891901 0.05850423  
## TotLns.Lses.Assets -5.7832318 3.4328145 -1.684691 0.09204820  
## odds  
## (Intercept) 471.1043931425508048960182350  
## TotCap.Assets 1.2180365223953835140946467  
## TotExp.Assets 0.0000000000000000007691045  
## TotLns.Lses.Assets 0.0030787494948847928498414

round(data.frame(summary(logit.reg)$coefficients, odds = exp(coef(logit.reg))), 5)

## Estimate Std..Error z.value Pr...z.. odds  
## (Intercept) 6.15508 3.26870 1.88303 0.05970 471.10439  
## TotCap.Assets 0.19724 0.11638 1.69479 0.09012 1.21804  
## TotExp.Assets -41.70906 22.04612 -1.89190 0.05850 0.00000  
## TotLns.Lses.Assets -5.78323 3.43281 -1.68469 0.09205 0.00308

# Confusion Matrix

predict <- predict(logit.reg, type = 'response')  
table(bank.df$Financial.Condition, predict > .5)

##   
## FALSE TRUE  
## Strong 10 0  
## Weak 1 9

# Reload data set for additional analyses

bank.df <- read.csv("banks.csv")  
bank.df <- bank.df[ , -c(1)] # Drop ID column.  
t(t(names(bank.df)))

## [,1]   
## [1,] "Financial.Condition"  
## [2,] "TotCap.Assets"   
## [3,] "TotExp.Assets"   
## [4,] "TotLns.Lses.Assets"

# Lift Chart

pred <- predict(logit.reg, bank.df)  
gain <- gains(bank.df$Financial.Condition, pred, groups=2)  
plot(c(0,gain$cume.pct.of.total\*sum(bank.df$Financial.Condition))~  
 c(0,gain$cume.obs),   
 xlab="# cases", ylab="Cumulative", main="", type="l")  
lines(c(0,sum(bank.df$Financial.Condition))~c(0, dim(bank.df)[1]), lty=2)

