NTU Exercise1

Yu Tian

2022-08-011

Package

```r
#package
library(dplyr)

#set seed
set.seed(0623)
```

Q1.

Use the data set called "Q1data.csv". This data set describes one consumer's purchase history of buying a certain good. It contains three variables.

- choice = 0 means no purchase; choice = 1 means buy.
- price ($)
- inventory

```r
# Read (and import) the full exercise data set into R using read.csv()
data1 <- read.csv(file = 'Q1data.csv')

# view the data example in R
data1
```

```
##    choice price inventory
## 1       1  11.0        20
## 2       0  25.0        30
## 3       0  12.0        23
## 4       1  25.0         3
## 5       0  26.0        15
## 6       1  10.0        23
## 7       1  12.0        40
## 8       0  24.0        15
## 9       0  26.0        13
## 10      0  28.0        18
## 11      0  11.0        60
## 12      1  12.0        17
## 13      1  11.5         3
## 14      1  10.0        25
## 15      0  26.0        40
## 16      1  28.0         5
## 17      1  11.0        35
## 18      1  25.0        10
```

```r
dim(data1)
```

```
## [1] 18  3
```

Use this data set to estimate the logit model.

Use choice as the dependent variable. price and inventory as the indenpendent variable.

Report the estimation results.

```r
lr_data1 <- glm(choice ~ price + inventory, data = data1, family = 'binomial')
summary(lr_data1)
```

```
##
## Call:
## glm(formula = choice ~ price + inventory, family = "binomial",
##     data = data1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2395  -0.4524   0.1740   0.6240   1.2530
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  9.86049    4.77572   2.065   0.0390 *
## price       -0.32684    0.15939  -2.051   0.0403 *
## inventory   -0.15286    0.08406  -1.819   0.0690 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 24.731  on 17  degrees of freedom
## Residual deviance: 13.940  on 15  degrees of freedom
## AIC: 19.94
##
## Number of Fisher Scoring iterations: 6
```

Q2

Based on your parameter estimates, compute the choice probability of choosing to buy when price = 20 and inventory equals mean inventory. [Note that we do not observe price = 20 in the data.

The estimated model allows us to predict what will happen if we set price at some values that we have not tried before.

```r
# calculate mean inventory
mean_inventory = mean(data1$inventory)

# specific prediction with price = 20 and mean inventory
spec_data <- with(data1, data.frame(price = 20, inventory = mean_inventory))
data1_pred = predict(lr_data1,spec_data)
data1_pred
```

```
##           1
## -0.03064686
```

```
prob = exp(data1_pred)/(1+exp(data1_pred))
prob
```

```
##         1
## 0.4923389
```

```
# or
data2_pred = predict(lr_data1,spec_data,type='response')
data2_pred
```

```
##         1
## 0.4923389
```

Q3

Use the train.csv data set to train a decision tree. The dependent variable is default and the independent variables are as what I give you in the lecture code.

try to train the decision tree using different cp values and report the prediction accuracy for the validation set.

```
# Read (and import) the full exercise data set into R using read.csv()
train_data <- read.csv(file = 'train.csv')
valid_data <- read.csv(file = 'validation.csv')

# view the data example in R
train_data %>% head()
```

```
##   MonthlyLoanPayment mgRate TEDRATE Adj.Close AmountRequested BorrowerRate
## 1           5.685075 0.0388    0.16  7.134572        9.104980       0.1095
## 2           4.840479 0.0389    0.16  7.136913        8.006368       0.2950
## 3           4.483793 0.0388    0.16  7.137946        7.649693       0.2950
## 4           5.557716 0.0339    0.20  7.203257        8.987197       0.1029
## 5           5.742202 0.0388    0.16  7.137946        9.210340       0.0765
## 6           4.389126 0.0399    0.22  7.200746        7.600902       0.2599
##   EstimatedLoss category1 category2 category3 category6 category7
## 1         0.035         1         0         0         0         0
## 2         0.108         0         0         1         0         0
## 3         0.108         0         0         0         1         0
## 4         0.026         1         0         0         0         0
## 5         0.013         0         0         0         0         1
## 6         0.108         0         0         0         1         0
##   IsBorrowerHomeowner bankcard_utilization credit_lines_last7_years
## 1                   0                 0.13                       15
## 2                   1                 0.99                       19
## 3                   0                 0.76                       43
## 4                   1                 0.22                       14
## 5                   1                 0.51                       18
## 6                   0                 0.00                       40
##   current_credit_lines current_delinquencies delinquencies_last7_years
## 1                    3                     0                         0
## 2                    8                     0                         0
## 3                   10                     0                         2
```

```
## 4                   3                   0                         0
## 5                   8                   0                         0
## 6                   9                   0                         0
##   delinquencies_over30_days delinquencies_over60_days
## 1                         3                         0
## 2                         0                         0
## 3                         3                         3
## 4                         0                         0
## 5                         0                         0
## 6                         0                         0
##   prior_prosper_loans_active income_range income_verifiable total_inquiries
## 1                          0            3                 1               1
## 2                          0            3                 1               3
## 3                          0            4                 1               0
## 4                          1            3                 1               3
## 5                          0            6                 1               1
## 6                          0            4                 1               1
##   inquiries_last6_months prior_prosper_loans revolving_available_percent
## 1                      1                   0                          83
## 2                      1                   0                           8
## 3                      0                   0                          36
## 4                      1                   1                          81
## 5                      0                   1                          53
## 6                      1                   0                          80
##   total_open_revolving_accounts revolving_balance monthly_debt
## 1                             3          8.176392    5.1416636
## 2                             5         10.229513    5.4424177
## 3                             9          9.608311    6.3750248
## 4                             3          8.408940    5.6204009
## 5                             5         10.879518    0.6931472
## 6                             3          8.146130    4.6249728
##   real_estate_balance group1 scorex620 scorex650 scorex665 scorex690 scorex702
## 1            0.000000      0         0         0         0         0         0
## 2           12.385256      0         0         0         0         0         1
## 3            0.000000      0         1         0         0         0         0
## 4            9.923878      0         0         0         0         0         0
## 5           13.039067      0         0         0         0         0         0
## 6            0.000000      0         0         0         0         0         0
##   scorex724 scorex748 scorex778 dti1 dti2 dti3 dti4 default
## 1         0         1         0    0    1    0    0       0
## 2         0         0         0    0    1    0    0       0
## 3         0         0         0    0    1    0    0       0
## 4         0         1         0    0    0    1    0       0
## 5         0         0         1    1    0    0    0       0
## 6         0         1         0    1    0    0    0       0
```

```
dim(train_data)
```

```
## [1] 2159   45
```

```
valid_data %>% head()
```

```
##   MonthlyLoanPayment mgRate TEDRATE Adj.Close AmountRequested BorrowerRate
## 1           4.237434 0.0388    0.16  7.136300        7.600902       0.1490
## 2           5.568306 0.0388    0.16  7.136300        8.699515       0.3220
```

```
## 3           5.386008 0.0389    0.17  7.138692        8.517193       0.3220
## 4           4.840479 0.0395    0.15  7.149799        8.006368       0.2950
## 5           6.147677 0.0389    0.15  7.150294        9.615805       0.0765
## 6           5.330300 0.0381    0.14  7.156114        8.699515       0.1449
##   EstimatedLoss category1 category2 category3 category6 category7
## 1        0.0595         0         0         0         0         1
## 2        0.1420         0         0         0         0         1
## 3        0.1420         0         0         1         0         0
## 4        0.1080         0         0         0         1         0
## 5        0.0130         1         0         0         0         0
## 6        0.0595         1         0         0         0         0
##   IsBorrowerHomeowner bankcard_utilization credit_lines_last7_years
## 1                   0                 0.47                       28
## 2                   1                 0.86                       22
## 3                   1                 0.93                       21
## 4                   1                 0.88                       20
## 5                   1                 0.89                       40
## 6                   0                 0.40                       16
##   current_credit_lines current_delinquencies delinquencies_last7_years
## 1                   11                     0                         0
## 2                    7                     0                         0
## 3                    8                     0                         0
## 4                   11                     0                         0
## 5                   15                     0                         0
## 6                    8                     0                         0
##   delinquencies_over30_days delinquencies_over60_days
## 1                         2                         0
## 2                         0                         0
## 3                         0                         0
## 4                         0                         0
## 5                         0                         0
## 6                         1                         0
##   prior_prosper_loans_active income_range income_verifiable total_inquiries
## 1                          0            4                 1               4
## 2                          0            7                 0               2
## 3                          0            3                 1               4
## 4                          0            3                 1               7
## 5                          0            6                 1               0
## 6                          0            3                 1               0
##   inquiries_last6_months prior_prosper_loans revolving_available_percent
## 1                      2                   0                          59
## 2                      0                   0                          13
## 3                      2                   0                          20
## 4                      0                   0                          26
## 5                      0                   1                          10
## 6                      0                   0                          63
##   total_open_revolving_accounts revolving_balance monthly_debt
## 1                             7          9.444938     6.040255
## 2                             4          9.625096     6.944087
## 3                             8         10.364198     6.073045
## 4                             5          9.861206     6.161207
## 5                            11         13.093112     6.712956
## 6                             5          8.606851     6.733402
##   real_estate_balance group1 scorex620 scorex650 scorex665 scorex690 scorex702
```

```
## 1             0.00000      0          0          0          0          0          0
## 2            12.26367      0          0          0          0          0          1
## 3            11.04129      0          0          0          0          1          0
## 4            11.78037      0          0          0          1          0          0
## 5            13.59022      1          0          0          0          0          0
## 6             0.00000      0          0          0          0          0          0
##   scorex724 scorex748 scorex778 dti1 dti2 dti3 dti4 default
## 1         0         1         0    0    1    0    0       0
## 2         0         0         0    0    0    0    0       0
## 3         0         0         0    0    1    0    0       0
## 4         0         0         0    0    1    0    0       0
## 5         0         1         0    1    0    0    0       0
## 6         1         0         0    0    0    1    0       0
```

```r
dim(valid_data)
```

```
## [1] 539  45
```

```r
# package
library(rpart)

# Train a decision tree with cp value=0.05
tree_data <- rpart(default~BorrowerRate+AmountRequested+IsBorrowerHomeowner+bankcard_utilization+credit_

# report the prediction accuracy for the validation set with cp=0.005
valid_data1=valid_data[,c('BorrowerRate','AmountRequested','IsBorrowerHomeowner',
                        'bankcard_utilization','credit_lines_last7_years',
                        'delinquencies_last7_years','prior_prosper_loans_active',
                        'income_range')]
prediction = predict(tree_data, valid_data1, type = 'class')
accuracy = sum(prediction == valid_data$default)/dim(valid_data)[1]
accuracy
```

```
## [1] 0.6827458
```

```r
# using different cp value with 0.01
tree_data1 <- rpart(default~BorrowerRate+AmountRequested+IsBorrowerHomeowner+bankcard_utilization+credit

# report the prediction accuracy for the validation set with cp=0.001
prediction1 = predict(tree_data1, valid_data1, type = 'class')
accuracy1 = sum(prediction1 == valid_data$default)/dim(valid_data)[1]
accuracy1
```

```
## [1] 0.6864564
```

```r
#  using different cp value with 0.001
tree_data2 <- rpart(default~BorrowerRate+AmountRequested+IsBorrowerHomeowner+bankcard_utilization+credit

# report the prediction accuracy for the validation set with cp=0.001
prediction2 = predict(tree_data2, valid_data1, type = 'class')
accuracy2 = sum(prediction2 == valid_data$default)/dim(valid_data)[1]
accuracy2
```

```
## [1] 0.6474954
```

Q4 C

Choose the decision tree with the best prediction performance and plot the decision tree using rpart.plot

```r
# choose the best decision tree
x=c(accuracy, accuracy1, accuracy2)
y=c("tree_data", "tree_data1", "tree_data2")
z=c(0.005, 0.01, 0.001)
treemodel = data.frame(x,y,z)
treemodel[order(treemodel$x, decreasing = TRUE),]
```

```
##           x          y     z
## 2 0.6864564 tree_data1 0.010
## 1 0.6827458  tree_data 0.005
## 3 0.6474954 tree_data2 0.001
```

```r
# From the table above, we can find best tree decision with cp = 0.01

rpart.plot::rpart.plot(tree_data1, box.palette="RdBu", shadow.col="gray", nn=TRUE)
```