HW2

muwuxu

2022-09-28

Q1

```
require(ISLR2)
```

```
## Loading required package: ISLR2
```

```
require(caret)
```

```
## Loading required package: caret
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
require(tidyverse)
```

```
## Loading required package: tidyverse
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──
## ✔ tibble  3.1.8      ✔ dplyr   1.0.10
## ✔ tidyr   1.2.0      ✔ stringr 1.4.0
## ✔ readr   2.1.2      ✔ forcats 0.5.1
## ✔ purrr   0.3.4
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ✖ purrr::lift()   masks caret::lift()
```

```
# Structure of the Data
data('College')
str(College)
```

```
## 'data.frame':    777 obs. of  18 variables:
##  $ Private    : Factor w/ 2 levels "No","Yes": 2 2 2 2 2 2 2 2 2 ...
##  $ Apps       : num  1660 2186 1428 417 193 ...
##  $ Accept     : num  1232 1924 1097 349 146 ...
##  $ Enroll     : num  721 512 336 137 55 158 103 489 227 172 ...
##  $ Top10perc  : num  23 16 22 60 16 38 17 37 30 21 ...
##  $ Top25perc  : num  52 29 50 89 44 62 45 68 63 44 ...
##  $ F.Undergrad: num  2885 2683 1036 510 249 ...
##  $ P.Undergrad: num  537 1227 99 63 869 ...
##  $ Outstate   : num  7440 12280 11250 12960 7560 ...
##  $ Room.Board : num  3300 6450 3750 5450 4120 ...
##  $ Books      : num  450 750 400 450 800 500 500 450 300 660 ...
##  $ Personal   : num  2200 1500 1165 875 1500 ...
##  $ PhD        : num  70 29 53 92 76 67 90 89 79 40 ...
##  $ Terminal   : num  78 30 66 97 72 73 93 100 84 41 ...
##  $ S.F.Ratio  : num  18.1 12.2 12.9 7.7 11.9 9.4 11.5 13.7 11.3 11.5 ...
##  $ perc.alumni: num  12 16 30 37 2 11 26 37 23 15 ...
##  $ Expend     : num  7041 10527 8735 19016 10922 ...
##  $ Grad.Rate  : num  60 56 54 59 15 55 63 73 80 52 ...
```

```
set.seed(1)
```

```
dim(College) # 777 * 18
```

```
## [1] 777  18
```

```
summary(College)
```

```
##  Private        Apps           Accept          Enroll       Top10perc
##  No :212   Min.   :   81   Min.   :   72   Min.   :  35   Min.   : 1.00
##  Yes:565   1st Qu.:  776   1st Qu.:  604   1st Qu.: 242   1st Qu.:15.00
##            Median : 1558   Median : 1110   Median : 434   Median :23.00
##            Mean   : 3002   Mean   : 2019   Mean   : 780   Mean   :27.56
##            3rd Qu.: 3624   3rd Qu.: 2424   3rd Qu.: 902   3rd Qu.:35.00
##            Max.   :48094   Max.   :26330   Max.   :6392   Max.   :96.00
##    Top25perc      F.Undergrad     P.Undergrad         Outstate
##  Min.   :  9.0   Min.   :  139   Min.   :    1.0   Min.   : 2340
##  1st Qu.: 41.0   1st Qu.:  992   1st Qu.:   95.0   1st Qu.: 7320
##  Median : 54.0   Median : 1707   Median :  353.0   Median : 9990
##  Mean   : 55.8   Mean   : 3700   Mean   :  855.3   Mean   :10441
##  3rd Qu.: 69.0   3rd Qu.: 4005   3rd Qu.:  967.0   3rd Qu.:12925
##  Max.   :100.0   Max.   :31643   Max.   :21836.0   Max.   :21700
##    Room.Board       Books           Personal         PhD
##  Min.   :1780   Min.   :  96.0   Min.   : 250   Min.   :  8.00
##  1st Qu.:3597   1st Qu.: 470.0   1st Qu.: 850   1st Qu.: 62.00
##  Median :4200   Median : 500.0   Median :1200   Median : 75.00
##  Mean   :4358   Mean   : 549.4   Mean   :1341   Mean   : 72.66
##  3rd Qu.:5050   3rd Qu.: 600.0   3rd Qu.:1700   3rd Qu.: 85.00
##  Max.   :8124   Max.   :2340.0   Max.   :6800   Max.   :103.00
##     Terminal       S.F.Ratio      perc.alumni        Expend
##  Min.   : 24.0   Min.   : 2.50   Min.   : 0.00   Min.   : 3186
##  1st Qu.: 71.0   1st Qu.:11.50   1st Qu.:13.00   1st Qu.: 6751
##  Median : 82.0   Median :13.60   Median :21.00   Median : 8377
##  Mean   : 79.7   Mean   :14.09   Mean   :22.74   Mean   : 9660
##  3rd Qu.: 92.0   3rd Qu.:16.50   3rd Qu.:31.00   3rd Qu.:10830
##  Max.   :100.0   Max.   :39.80   Max.   :64.00   Max.   :56233
##    Grad.Rate
##  Min.   : 10.00
##  1st Qu.: 53.00
##  Median : 65.00
##  Mean   : 65.46
##  3rd Qu.: 78.00
##  Max.   :118.00
```

```
# there is no NAs.
```

```
# split the data
inTrain <- createDataPartition(College$Apps, p = 0.80, list = FALSE)

training <- College[inTrain,]
testing <- College[-inTrain,]

head(training)
```

```
##                              Private Apps Accept Enroll Top10perc Top25perc
## Abilene Christian University     Yes 1660   1232    721        23        52
## Adelphi University               Yes 2186   1924    512        16        29
## Adrian College                   Yes 1428   1097    336        22        50
## Agnes Scott College              Yes  417    349    137        60        89
## Alaska Pacific University        Yes  193    146     55        16        44
## Albion College                   Yes 1899   1720    489        37        68
##                              F.Undergrad P.Undergrad Outstate Room.Board Books
## Abilene Christian University        2885         537     7440       3300   450
## Adelphi University                  2683        1227    12280       6450   750
## Adrian College                      1036          99    11250       3750   400
## Agnes Scott College                  510          63    12960       5450   450
## Alaska Pacific University            249         869     7560       4120   800
## Albion College                      1594          32    13868       4826   450
##                              Personal PhD Terminal S.F.Ratio perc.alumni Expend
## Abilene Christian University     2200  70       78      18.1          12   7041
## Adelphi University               1500  29       30      12.2          16  10527
## Adrian College                   1165  53       66      12.9          30   8735
## Agnes Scott College               875  92       97       7.7          37  19016
## Alaska Pacific University        1500  76       72      11.9           2  10922
## Albion College                    850  89      100      13.7          37  11487
##                              Grad.Rate
## Abilene Christian University        60
## Adelphi University                  56
## Adrian College                      54
## Agnes Scott College                 59
## Alaska Pacific University           15
## Albion College                      73
```

```
nrow(training)
```

```
## [1] 624
```

```
head(testing)
```

```
##                                Private Apps Accept Enroll Top10perc Top25perc
## Albertson College                  Yes  587    479    158         38         62
## Albertus Magnus College            Yes  353    340    103         17         45
## Alderson-Broaddus College          Yes  582    498    172         21         44
## Alverno College                    Yes  494    313    157         23         46
## American International College     Yes 1420   1093    220          9         22
## Andrews University                 Yes 1130    704    322         14         23
##                                F.Undergrad P.Undergrad Outstate Room.Board
## Albertson College                      678          41    13500       3335
## Albertus Magnus College                416         230    13290       5720
## Alderson-Broaddus College              799          78    10468       3380
## Alverno College                       1317        1235     8352       3640
## American International College        1018         287     8700       4780
## Andrews University                    1586         326     9996       3090
##                                Books Personal PhD Terminal S.F.Ratio
## Albertson College                500      675  67       73       9.4
## Albertus Magnus College          500     1500  90       93      11.5
## Alderson-Broaddus College        660     1800  40       41      11.5
## Alverno College                  650     2449  36       69      11.1
## American International College   450     1400  78       84      14.7
## Andrews University               900     1320  62       66      11.5
##                                perc.alumni Expend Grad.Rate
## Albertson College                       11   9727        55
## Albertus Magnus College                 26   8861        63
## Alderson-Broaddus College               15   8991        52
## Alverno College                         26   8127        55
## American International College          19   7355        69
## Andrews University                      18  10908        46
```

```
nrow(testing)
```

```
## [1] 153
```

```
# do the transfomation
preObj <- preProcess(training, method = c('center', 'scale'))

training <- predict(preObj, training)
testing <- predict(preObj, testing)

head(training)
```

```
##                               Private       Apps      Accept      Enroll
## Abilene Christian University     Yes -0.3360340 -0.31740340 -0.06705396
## Adelphi University               Yes -0.2005673 -0.03779614 -0.28761208
## Adrian College                   Yes -0.3957836 -0.37195106 -0.47334524
## Agnes Scott College              Yes -0.6561578 -0.67418550 -0.68335034
## Alaska Pacific University        Yes -0.7138471 -0.75620902 -0.76988511
## Albion College                   Yes -0.2744816 -0.12022371 -0.31188403
##                                Top10perc  Top25perc F.Undergrad  P.Undergrad
## Abilene Christian University -0.2306658 -0.1599444  -0.1664647 -0.223692098
## Adelphi University           -0.6351412 -1.3323257  -0.2081655  0.220852776
## Adrian College               -0.2884480 -0.2618906  -0.5481715 -0.505881453
## Agnes Scott College           1.9072756  1.7260602  -0.6567587 -0.529075098
## Alaska Pacific University    -0.6351412 -0.5677292  -0.7106394 -0.009795144
## Albion College                0.5782850  0.6556251  -0.4329782 -0.549047404
##                                Outstate Room.Board      Books   Personal
## Abilene Christian University -0.7166224 -0.9461378 -0.6117445  1.2355947
## Adelphi University            0.4936784  1.9289647  1.3600280  0.2206753
## Adrian College                0.2361144 -0.5354089 -0.9403733 -0.2650362
## Agnes Scott College           0.6637206  1.0162338 -0.6117445 -0.6855028
## Alaska Pacific University    -0.6866149 -0.1976984  1.6886568  0.2206753
## Albion College                0.8907770  0.4466896 -0.6117445 -0.7217499
##                                     PhD    Terminal  S.F.Ratio perc.alumni
## Abilene Christian University -0.1608567 -0.09475071  0.9903695  -0.8382715
## Adelphi University           -2.7704677 -3.39471968 -0.4971848  -0.5159289
## Adrian College               -1.2428905 -0.91974295 -0.3206953   0.6122701
## Agnes Scott College           1.2394224  1.21148700 -1.6317601   1.1763696
## Alaska Pacific University     0.2210376 -0.50724683 -0.5728231  -1.6441279
## Albion College                1.0484753  1.41773506 -0.1189930   1.1763696
##                                  Expend  Grad.Rate
## Abilene Christian University -0.4776280 -0.2972397
## Adelphi University            0.2309198 -0.5299593
## Adrian College               -0.1333136 -0.6463191
## Agnes Scott College           1.9563534 -0.3554196
## Alaska Pacific University     0.3112056 -2.9153357
## Albion College                0.4260448  0.4590992
```

```
head(testing)
```

```
##                                Private       Apps     Accept     Enroll
## Albertson College                  Yes -0.6123758 -0.6216581 -0.6611890
## Albertus Magnus College            Yes -0.6726405 -0.6778220 -0.7192306
## Alderson-Broaddus College          Yes -0.6136635 -0.6139810 -0.6464148
## Alverno College                    Yes -0.6363271 -0.6887315 -0.6622443
## American International College     Yes -0.3978439 -0.3735673 -0.5957603
## Andrews University                 Yes -0.4725309 -0.5307454 -0.4881195
##                                 Top10perc  Top25perc F.Undergrad P.Undergrad
## Albertson College               0.6360672  0.3497866  -0.6220769  -0.5432490
## Albertus Magnus College        -0.5773590 -0.5167561  -0.6761640  -0.4214824
## Alderson-Broaddus College      -0.3462302 -0.5677292  -0.5970977  -0.5194111
## Alverno College                -0.2306658 -0.4657830  -0.4901620   0.2260069
## American International College -1.0396166 -1.6891374  -0.5518874  -0.3847591
## Andrews University             -0.7507056 -1.6381643  -0.4346297  -0.3596326
##                                   Outstate Room.Board      Books    Personal
## Albertson College               0.79875417 -0.9141922 -0.2831158 -0.97547977
## Albertus Magnus College         0.74624112  1.2626711 -0.2831158  0.22067527
## Alderson-Broaddus College       0.04056577 -0.8731193  0.7684962  0.65564074
## Alverno College                -0.48856571 -0.6358093  0.7027705  1.59661603
## American International College -0.40154409  0.4047040 -0.6117445  0.07568678
## Andrews University             -0.07746356 -1.1378113  2.3459143 -0.04030401
##                                       PhD   Terminal  S.F.Ratio perc.alumni
## Albertson College              -0.3518038 -0.4384975 -1.2031428  -0.9188571
## Albertus Magnus College         1.1121243  0.9364896 -0.6736743   0.2899275
## Alderson-Broaddus College      -2.0703281 -2.6384768 -0.6736743  -0.5965145
## Alverno College                -2.3249243 -0.7134949 -0.7745254   0.2899275
## American International College  0.3483357  0.3177454  0.1331348  -0.2741720
## Andrews University             -0.6700491 -0.9197430 -0.6736743  -0.3547576
##                                    Expend  Grad.Rate
## Albertson College               0.06831558 -0.5881392
## Albertus Magnus College        -0.10770345 -0.1226999
## Alderson-Broaddus College      -0.08128027 -0.7626790
## Alverno College                -0.25689279 -0.5881392
## American International College -0.41380583  0.2263795
## Andrews University              0.30836001 -1.1117584
```

```
#
y_train <- training$Apps
y_test <- testing$Apps

# change the categorical to dummy
one_hot_encoding <- dummyVars(Apps ~ ., data = training)
x_train <- predict(one_hot_encoding, training)
x_test <- predict(one_hot_encoding, testing)

dim(x_train)
```

```
## [1] 624  18
```

```
colnames(College)
```

```
##  [1] "Private"     "Apps"        "Accept"      "Enroll"      "Top10perc"
##  [6] "Top25perc"   "F.Undergrad" "P.Undergrad" "Outstate"    "Room.Board"
## [11] "Books"       "Personal"    "PhD"         "Terminal"    "S.F.Ratio"
## [16] "perc.alumni" "Expend"      "Grad.Rate"
```

```
colnames(x_train)
```

```
##  [1] "Private.No"  "Private.Yes" "Accept"      "Enroll"      "Top10perc"
##  [6] "Top25perc"   "F.Undergrad" "P.Undergrad" "Outstate"    "Room.Board"
## [11] "Books"       "Personal"    "PhD"         "Terminal"    "S.F.Ratio"
## [16] "perc.alumni" "Expend"      "Grad.Rate"
```

```
head(x_train)
```

```
##                               Private.No Private.Yes      Accept      Enroll
## Abilene Christian University          0           1 -0.31740340 -0.06705396
## Adelphi University                    0           1 -0.03779614 -0.28761208
## Adrian College                        0           1 -0.37195106 -0.47334524
## Agnes Scott College                   0           1 -0.67418550 -0.68335034
## Alaska Pacific University             0           1 -0.75620902 -0.76988511
## Albion College                        0           1 -0.12022371 -0.31188403
##                                Top10perc  Top25perc F.Undergrad  P.Undergrad
## Abilene Christian University -0.2306658 -0.1599444  -0.1664647 -0.223692098
## Adelphi University           -0.6351412 -1.3323257  -0.2081655  0.220852776
## Adrian College               -0.2884480 -0.2618906  -0.5481715 -0.505881453
## Agnes Scott College           1.9072756  1.7260602  -0.6567587 -0.529075098
## Alaska Pacific University    -0.6351412 -0.5677292  -0.7106394 -0.009795144
## Albion College                0.5782850  0.6556251  -0.4329782 -0.549047404
##                                 Outstate Room.Board      Books   Personal
## Abilene Christian University -0.7166224 -0.9461378 -0.6117445  1.2355947
## Adelphi University            0.4936784  1.9289647  1.3600280  0.2206753
## Adrian College                0.2361144 -0.5354089 -0.9403733 -0.2650362
## Agnes Scott College           0.6637206  1.0162338 -0.6117445 -0.6855028
## Alaska Pacific University    -0.6866149 -0.1976984  1.6886568  0.2206753
## Albion College                0.8907770  0.4466896 -0.6117445 -0.7217499
##                                      PhD    Terminal  S.F.Ratio perc.alumni
## Abilene Christian University -0.1608567 -0.09475071  0.9903695  -0.8382715
## Adelphi University           -2.7704677 -3.39471968 -0.4971848  -0.5159289
## Adrian College               -1.2428905 -0.91974295 -0.3206953   0.6122701
## Agnes Scott College           1.2394224  1.21148700 -1.6317601   1.1763696
## Alaska Pacific University     0.2210376 -0.50724683 -0.5728231  -1.6441279
## Albion College                1.0484753  1.41773506 -0.1189930   1.1763696
##                                  Expend  Grad.Rate
## Abilene Christian University -0.4776280 -0.2972397
## Adelphi University            0.2309198 -0.5299593
## Adrian College               -0.1333136 -0.6463191
## Agnes Scott College           1.9563534 -0.3554196
## Alaska Pacific University     0.3112056 -2.9153357
## Albion College                0.4260448  0.4590992
```

```
lin_mod <- lm(Apps ~ ., data = training)

pred <- predict(lin_mod, testing)

lin_info <- postResample(pred, testing$Apps)
lin_info
```

```
##      RMSE  Rsquared       MAE
## 0.3449780 0.8780602 0.1845678
```

```
ridge_mod <- train(x = x_train, y = y_train,
                   method = 'glmnet',
                   trControl = trainControl(method = 'cv', number = 10),
                   tuneGrid = expand.grid(alpha = 0,
                                          lambda = seq(0, 10e3, length.out = 20)))
```

```
## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info = trainInfo, :
## There were missing values in resampled performance measures.
```

```
ridge_info <- postResample(predict(ridge_mod, x_test), y_test)
ridge_info
```

```
##      RMSE  Rsquared       MAE
## 0.3232724 0.8936783 0.1749656
```

```
coef(ridge_mod$finalModel, ridge_mod$bestTune$lambda)
```

```
## 19 x 1 sparse Matrix of class "dgCMatrix"
##                         s1
## (Intercept)  3.446877e-02
## Private.No   7.741433e-02
## Private.Yes -7.807576e-02
## Accept       6.814488e-01
## Enroll       8.710131e-02
## Top10perc    9.917812e-02
## Top25perc    5.384254e-03
## F.Undergrad  6.318160e-02
## P.Undergrad  1.704241e-02
## Outstate    -2.770280e-02
## Room.Board   5.307513e-02
## Books        9.415904e-03
## Personal    -1.251208e-02
## PhD         -1.493455e-02
## Terminal    -9.985495e-03
## S.F.Ratio    8.011033e-05
## perc.alumni -3.317318e-02
## Expend       7.881035e-02
## Grad.Rate    5.333449e-02
```

```
lasso_mod <- train(x = x_train, y = y_train,
                   method = 'glmnet',
                   trControl = trainControl(method = 'cv', number = 10),
                   tuneGrid = expand.grid(alpha = 1,
                                          lambda = seq(0.0001, 1, length.out = 50)))
```

```
## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info = trainInfo, :
## There were missing values in resampled performance measures.
```

```
lasso_info <- postResample(predict(lasso_mod, x_test), y_test)
lasso_info
```

```
##      RMSE  Rsquared       MAE
## 0.3449963 0.8778473 0.1832701
```

```
as_data_frame(rbind(lin_info,
      ridge_info,
      lasso_info)) %>%
    mutate(model = c('Linear', 'Ridge', 'Lasso' ))
```

```
## Warning: `as_data_frame()` was deprecated in tibble 2.0.0.
## Please use `as_tibble()` instead.
## The signature and semantics have changed, see `?as_tibble`.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```

```
## # A tibble: 3 × 4
##    RMSE Rsquared   MAE model
##   <dbl>    <dbl> <dbl> <chr>
## 1 0.345    0.878 0.185 Linear
## 2 0.323    0.894 0.175 Ridge
## 3 0.345    0.878 0.183 Lasso
```

```
## The models all perform similarly. R2 around 87% for them all and RMSE≤0.35. When we c
ompare the RMSE scores with the mean and standard deviation of the response variable we
 see that the models all have great accuracy. So there are not much differences between
 these three approaches.
```

Q2

```
library(ggplot2)
library(caret)
library(glmnet)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Loaded glmnet 4.1-4
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(leaps)
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

Can you predict who will be interested in buying a caravan insurance policy and give an explanation why? Yes. Becaues we can use the features to fit a model and then predict the probability so to priedict who will be interested in buying the insurance.

```
train_data <- read.delim("./ticdata2000.txt", header = FALSE, sep = "\t", dec = ".")
test_data <-  read.delim("./ticeval2000.txt", header = FALSE, sep = "\t", dec = ".")
targets <-    read.delim("./tictgts2000.txt", header = FALSE, sep = "\t", dec = ".")
names(targets) = "V86"
test_data = cbind(test_data,targets)
head(test_data)
```

```
##   V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21
## 1 33  1  4  2  8  0  6  0  3   5   0   4   1   1   8   2   2   6   0   0   1
## 2  6  1  3  2  2  0  5  0  4   5   2   2   1   4   5   5   4   0   5   0   0
## 3 39  1  3  3  9  1  4  2  3   5   2   3   2   3   6   2   4   4   2   1   1
## 4  9  1  2  3  3  2  3  2  4   5   4   1   2   4   4   2   4   4   2   1   1
## 5 31  1  2  4  7  0  2  0  7   9   0   0   0   6   3   0   0   9   0   0   0
## 6 30  1  2  4  7  1  4  2  3   5   0   4   4   3   2   1   2   6   1   0   1
##   V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34 V35 V36 V37 V38 V39 V40
## 1   2   6   1   0   2   1   5   3   1   8   8   1   1   8   1   3   3   3   0
## 2   4   0   0   4   3   0   2   1   3   6   9   0   0   7   2   1   1   5   4
## 3   3   2   2   1   1   5   2   1   1   8   6   2   2   6   3   2   4   3   1
## 4   5   1   2   3   1   3   2   2   3   6   7   2   1   7   2   2   5   3   1
## 5   2   4   4   0   0   0   7   2   9   0   7   2   0   9   0   5   4   0   0
## 6   3   3   3   1   1   2   5   1   5   4   5   1   4   9   0   2   5   2   1
##   V41 V42 V43 V44 V45 V46 V47 V48 V49 V50 V51 V52 V53 V54 V55 V56 V57 V58 V59
## 1   0   3   3   1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   4
## 2   0   6   8   2   0   0   6   0   4   0   0   0   0   0   3   0   0   0   4
## 3   0   3   5   2   0   0   6   0   0   0   0   0   0   0   4   0   0   0   4
## 4   0   4   4   2   0   0   5   0   0   0   0   0   0   0   0   0   0   0   3
## 5   0   3   1   2   0   0   0   0   0   0   0   0   0   0   0   0   0   0   1
## 6   0   4   2   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   4
##   V60 V61 V62 V63 V64 V65 V66 V67 V68 V69 V70 V71 V72 V73 V74 V75 V76 V77 V78
## 1   0   0   0   0   0   1   0   0   0   0   0   0   0   0   0   0   0   0   0
## 2   0   0   0   0   0   1   0   0   1   0   1   0   0   0   0   0   2   0   0
## 3   0   0   0   0   0   1   0   0   1   0   0   0   0   0   0   0   1   0   0
## 4   0   0   0   0   0   1   0   0   1   0   0   0   0   0   0   0   0   0   0
## 5   0   0   0   0   0   1   0   0   0   0   0   0   0   0   0   0   0   0   0
## 6   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
##   V79 V80 V81 V82 V83 V84 V85 V86
## 1   0   1   0   0   0   0   0   0
## 2   0   1   0   0   0   0   0   1
## 3   0   1   0   0   0   0   0   0
## 4   0   1   0   0   0   0   0   0
## 5   0   1   0   0   0   0   0   0
## 6   0   2   0   0   0   0   0   0
```

Explore data

```
dim(train_data)
```

```
## [1] 5822   86
```

```
head(train_data)
```

```
##   V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21
## 1 33  1  3  2  8  0  5  1  3   7   0   2   1   2   6   1   2   7   1   0   1
## 2 37  1  2  2  8  1  4  1  4   6   2   2   0   4   5   0   5   4   0   0   0
## 3 37  1  2  2  8  0  4  2  4   3   2   4   4   4   2   0   5   4   0   0   0
## 4  9  1  3  3  3  2  3  2  4   5   2   2   2   3   4   3   4   2   4   0   0
## 5 40  1  4  2 10  1  4  1  4   7   1   2   2   4   4   5   4   0   0   5   4
## 6 23  1  2  1  5  0  5  0  5   0   6   3   3   5   2   0   5   4   2   0   0
##   V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34 V35 V36 V37 V38 V39 V40
## 1   2   5   2   1   1   2   6   1   1   8   8   0   1   8   1   0   4   5   0
## 2   5   0   4   0   2   3   5   0   2   7   7   1   2   6   3   2   0   5   2
## 3   7   0   2   0   5   0   4   0   7   2   7   0   2   9   0   4   5   0   0
## 4   3   1   2   3   2   1   4   0   5   4   9   0   0   7   2   1   5   3   0
## 5   0   0   0   9   0   0   0   0   4   5   6   2   1   5   4   0   0   9   0
## 6   4   2   2   2   2   2   4   2   9   0   5   3   3   9   0   5   2   3   0
##   V41 V42 V43 V44 V45 V46 V47 V48 V49 V50 V51 V52 V53 V54 V55 V56 V57 V58 V59
## 1   0   4   3   0   0   0   6   0   0   0   0   0   0   0   0   0   0   0   5
## 2   0   5   4   2   0   0   0   0   0   0   0   0   0   0   0   0   0   0   2
## 3   0   3   4   2   0   0   6   0   0   0   0   0   0   0   0   0   0   0   2
## 4   0   4   4   0   0   0   6   0   0   0   0   0   0   0   0   0   0   0   2
## 5   0   6   3   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   6
## 6   0   3   3   0   0   0   6   0   0   0   0   0   0   0   0   0   0   0   0
##   V60 V61 V62 V63 V64 V65 V66 V67 V68 V69 V70 V71 V72 V73 V74 V75 V76 V77 V78
## 1   0   0   0   0   0   0   0   0   1   0   0   0   0   0   0   0   0   0   0
## 2   0   0   0   0   0   2   0   0   0   0   0   0   0   0   0   0   0   0   0
## 3   0   0   0   0   0   1   0   0   1   0   0   0   0   0   0   0   0   0   0
## 4   0   0   0   0   0   0   0   0   1   0   0   0   0   0   0   0   0   0   0
## 5   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
## 6   0   0   0   0   0   0   0   0   1   0   0   0   0   0   0   0   0   0   0
##   V79 V80 V81 V82 V83 V84 V85 V86
## 1   0   1   0   0   0   0   0   0
## 2   0   1   0   0   0   0   0   0
## 3   0   1   0   0   0   0   0   0
## 4   0   1   0   0   0   0   0   0
## 5   0   1   0   0   0   0   0   0
## 6   0   0   0   0   0   0   0   0
```

```
dim(test_data)
```

```
## [1] 4000   86
```

```
head(test_data)
```

```
##   V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21
## 1 33  1  4  2  8  0  6  0  3   5   0   4   1   1   8   2   2   6   0   0   1
## 2  6  1  3  2  2  0  5  0  4   5   2   2   1   4   5   5   4   0   5   0   0
## 3 39  1  3  3  9  1  4  2  3   5   2   3   2   3   6   2   4   4   2   1   1
## 4  9  1  2  3  3  2  3  2  4   5   4   1   2   4   4   2   4   4   2   1   1
## 5 31  1  2  4  7  0  2  0  7   9   0   0   0   6   3   0   0   9   0   0   0
## 6 30  1  2  4  7  1  4  2  3   5   0   4   4   3   2   1   2   6   1   0   1
##   V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34 V35 V36 V37 V38 V39 V40
## 1   2   6   1   0   2   1   5   3   1   8   8   1   1   8   1   3   3   3   0
## 2   4   0   0   4   3   0   2   1   3   6   9   0   0   7   2   1   1   5   4
## 3   3   2   2   1   1   5   2   1   1   8   6   2   2   6   3   2   4   3   1
## 4   5   1   2   3   1   3   2   2   3   6   7   2   1   7   2   2   5   3   1
## 5   2   4   4   0   0   0   7   2   9   0   7   2   0   9   0   5   4   0   0
## 6   3   3   3   1   1   2   5   1   5   4   5   1   4   9   0   2   5   2   1
##   V41 V42 V43 V44 V45 V46 V47 V48 V49 V50 V51 V52 V53 V54 V55 V56 V57 V58 V59
## 1   0   3   3   1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   4
## 2   0   6   8   2   0   0   6   0   4   0   0   0   0   0   3   0   0   0   4
## 3   0   3   5   2   0   0   6   0   0   0   0   0   0   0   4   0   0   0   4
## 4   0   4   4   2   0   0   5   0   0   0   0   0   0   0   0   0   0   0   3
## 5   0   3   1   2   0   0   0   0   0   0   0   0   0   0   0   0   0   0   1
## 6   0   4   2   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   4
##   V60 V61 V62 V63 V64 V65 V66 V67 V68 V69 V70 V71 V72 V73 V74 V75 V76 V77 V78
## 1   0   0   0   0   0   1   0   0   0   0   0   0   0   0   0   0   0   0   0
## 2   0   0   0   0   0   1   0   0   1   0   1   0   0   0   0   0   2   0   0
## 3   0   0   0   0   0   1   0   0   1   0   0   0   0   0   0   0   1   0   0
## 4   0   0   0   0   0   1   0   0   1   0   0   0   0   0   0   0   0   0   0
## 5   0   0   0   0   0   1   0   0   0   0   0   0   0   0   0   0   0   0   0
## 6   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
##   V79 V80 V81 V82 V83 V84 V85 V86
## 1   0   1   0   0   0   0   0   0
## 2   0   1   0   0   0   0   0   1
## 3   0   1   0   0   0   0   0   0
## 4   0   1   0   0   0   0   0   0
## 5   0   1   0   0   0   0   0   0
## 6   0   2   0   0   0   0   0   0
```

```
head(targets)
```

```
##   V86
## 1   0
## 2   1
## 3   0
## 4   0
## 5   0
## 6   0
```

```
str(train_data)
```

```
## 'data.frame':    5822 obs. of  86 variables:
##  $ V1 : int  33 37 37 9 40 23 39 33 33 11 ...
##  $ V2 : int  1 1 1 1 1 1 2 1 1 2 ...
##  $ V3 : int  3 2 2 3 4 2 3 2 2 3 ...
##  $ V4 : int  2 2 2 3 2 1 2 3 4 3 ...
##  $ V5 : int  8 8 8 3 10 5 9 8 8 3 ...
##  $ V6 : int  0 1 0 2 1 0 2 0 0 3 ...
##  $ V7 : int  5 4 4 3 4 5 2 7 1 5 ...
##  $ V8 : int  1 1 2 2 1 0 0 0 3 0 ...
##  $ V9 : int  3 4 4 4 4 5 5 2 6 2 ...
##  $ V10: int  7 6 3 5 7 0 7 7 6 7 ...
##  $ V11: int  0 2 2 2 1 6 2 2 0 0 ...
##  $ V12: int  2 2 4 2 2 3 0 0 3 2 ...
##  $ V13: int  1 0 4 2 2 3 0 0 3 2 ...
##  $ V14: int  2 4 4 3 4 5 3 5 3 2 ...
##  $ V15: int  6 5 2 4 4 2 6 4 3 6 ...
##  $ V16: int  1 0 0 3 5 0 0 0 0 0 ...
##  $ V17: int  2 5 5 4 4 5 4 3 1 4 ...
##  $ V18: int  7 4 4 2 0 4 5 6 8 5 ...
##  $ V19: int  1 0 0 4 0 2 0 2 1 2 ...
##  $ V20: int  0 0 0 0 5 0 0 0 1 0 ...
##  $ V21: int  1 0 0 0 4 0 0 0 0 0 ...
##  $ V22: int  2 5 7 3 0 4 4 2 1 3 ...
##  $ V23: int  5 0 0 1 0 2 1 5 8 3 ...
##  $ V24: int  2 4 2 2 0 2 5 2 1 3 ...
##  $ V25: int  1 0 0 3 9 2 0 2 1 1 ...
##  $ V26: int  1 2 5 2 0 2 1 1 1 2 ...
##  $ V27: int  2 3 0 1 0 2 4 2 0 1 ...
##  $ V28: int  6 5 4 4 0 4 5 5 8 4 ...
##  $ V29: int  1 0 0 0 0 2 0 2 1 2 ...
##  $ V30: int  1 2 7 5 4 9 6 0 9 0 ...
##  $ V31: int  8 7 2 4 5 0 3 9 0 9 ...
##  $ V32: int  8 7 7 9 6 5 8 4 5 6 ...
##  $ V33: int  0 1 0 0 2 3 0 4 2 1 ...
##  $ V34: int  1 2 2 0 1 3 1 2 3 2 ...
##  $ V35: int  8 6 9 7 5 9 9 6 7 6 ...
##  $ V36: int  1 3 0 2 4 0 0 3 2 3 ...
##  $ V37: int  0 2 4 1 0 5 4 2 7 2 ...
##  $ V38: int  4 0 5 5 0 2 3 5 2 3 ...
##  $ V39: int  5 5 0 3 9 3 3 3 1 3 ...
##  $ V40: int  0 2 0 0 0 0 0 0 0 1 ...
##  $ V41: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V42: int  4 5 3 4 6 3 3 3 2 4 ...
##  $ V43: int  3 4 4 4 3 3 5 3 3 7 ...
##  $ V44: int  0 2 2 0 0 0 0 0 0 2 ...
##  $ V45: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V46: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V47: int  6 0 6 6 0 6 6 0 5 0 ...
##  $ V48: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V49: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V50: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V51: int  0 0 0 0 0 0 0 0 0 0 ...
```

```
##  $ V52: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V53: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V54: int  0 0 0 0 0 0 0 3 0 0 ...
##  $ V55: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V56: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V57: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V58: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V59: int  5 2 2 2 6 0 0 0 0 3 ...
##  $ V60: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V61: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V62: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V63: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V64: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V65: int  0 2 1 0 0 0 0 0 0 1 ...
##  $ V66: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V67: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V68: int  1 0 1 1 0 1 1 0 1 0 ...
##  $ V69: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V70: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V71: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V72: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V73: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V74: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V75: int  0 0 0 0 0 0 0 1 0 0 ...
##  $ V76: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V77: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V78: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V79: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V80: int  1 1 1 1 1 0 0 0 0 1 ...
##  $ V81: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V82: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V83: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V84: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V85: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V86: int  0 0 0 0 0 0 0 0 0 0 ...
```

forward selection

```
# Formula for scope

regfit.fwd <- regsubsets(V86~., data = train_data, nbest = 1, nvmax = ncol(train_data),
 method = "forward")
my_sum_fwd <- summary(regfit.fwd)
my_sum_fwd$outmat
```

```
##           V1  V2  V3  V4  V5  V6  V7  V8  V9  V10 V11 V12 V13 V14 V15 V16 V17
## 1  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 2  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 3  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 4  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 5  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 6  ( 1 )  " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 7  ( 1 )  " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 8  ( 1 )  " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 9  ( 1 )  " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 10  ( 1 ) " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 11  ( 1 ) " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 12  ( 1 ) " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 13  ( 1 ) " " " " " " " "*" " " " " " " " " " " " " "*" " " " " " " " " " " "
## 14  ( 1 ) " " " " " " " "*" " " " " " " " " " " " " "*" " " " " " " " " " " "
## 15  ( 1 ) " " " " " " " "*" " " " " " " " " " " " " "*" " " " " " " " " " " "
## 16  ( 1 ) " " " " " " " "*" " " " " " " " " " " " " "*" " " " " " " " " " " "
## 17  ( 1 ) " " " " " " " "*" " " " " " " " " " " " " "*" " " " " " " " " " " "
## 18  ( 1 ) " " " " " " " "*" " " " " " " " " " " " " "*" " " " " " " " " "*" " "
## 19  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 20  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 21  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 22  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 23  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 24  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 25  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 26  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 27  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 28  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 29  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 30  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 31  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 32  ( 1 ) " " " " " " " "*" " " " " " "*" " " " " " "*" " " " " " " " " "*" " "
## 33  ( 1 ) " " " " " " " "*" " " " " " "*" "*" " " " " "*" " " " " " " " " "*" " "
## 34  ( 1 ) " " " " " " " "*" " " " " " "*" "*" " " " " "*" " " " " " " " " "*" " "
## 35  ( 1 ) " " " " " " " "*" " " " " " "*" "*" " " " " "*" " " " " " " " " "*" " "
## 36  ( 1 ) " " " " " " " "*" " " " " " "*" "*" " " " " "*" " " " " " " " " "*" " "
## 37  ( 1 ) " " " " " " " "*" " " " " " "*" "*" " " " " "*" " " " " " " " "*" " " "*" " "
## 38  ( 1 ) " " "*" " " " " "*" " " " " " "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 39  ( 1 ) " " "*" " " " " "*" " " " " " "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 40  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 41  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 42  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 43  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 44  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 45  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 46  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 47  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 48  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 49  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " " " " " "*" " " "*" " "
## 50  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" " " "*" " "
## 51  ( 1 ) " " "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" " " "*" " "
```

```
## 52  ( 1 ) " " "*" " " "*" " " "*" "*" "*" " " "*" " " "*" " " "*" " " "*" " " "*" " "
## 53  ( 1 ) " " "*" " " "*" " " "*" "*" "*" " " "*" " " "*" " " "*" " " "*" " " "*" " "
## 54  ( 1 ) " " "*" " " "*" " " "*" "*" "*" " " "*" " " "*" " " "*" " " "*" " " "*" " "
## 55  ( 1 ) " " "*" " " "*" " " "*" "*" "*" " " "*" " " "*" " " "*" " " "*" " " "*" " "
## 56  ( 1 ) " " "*" " " "*" " " "*" "*" "*" " " "*" " " "*" " " "*" " " "*" " " "*" " "
## 57  ( 1 ) " " "*" " " "*" " " "*" "*" "*" " " "*" " " "*" " " "*" " " "*" " " "*" " "
## 58  ( 1 ) " " "*" " " "*" " " "*" "*" "*" " " "*" " " "*" " " "*" " " "*" " " "*" " "
## 59  ( 1 ) " " "*" " " "*" " " "*" "*" "*" " " "*" " " "*" " " "*" " " "*" " " "*" " "
## 60  ( 1 ) " " "*" " " "*" " " "*" "*" "*" " " "*" " " "*" " " "*" " " "*" " " "*" "*"
## 61  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " "*" " " "*" "*"
## 62  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " "*" "*" "*" "*"
## 63  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 64  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 65  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 66  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 67  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 68  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 69  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 70  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 71  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 72  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 73  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 74  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 75  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 76  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 77  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 78  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 79  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 80  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 81  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 82  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 83  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 84  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 85  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
##           V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34
## 1   ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 2   ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 3   ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 4   ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 5   ( 1 ) "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 6   ( 1 ) "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 7   ( 1 ) "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 8   ( 1 ) "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 9   ( 1 ) "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 10  ( 1 ) "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 11  ( 1 ) "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 12  ( 1 ) "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 13  ( 1 ) "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 14  ( 1 ) "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 15  ( 1 ) "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 16  ( 1 ) "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 17  ( 1 ) "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " "
```

```
## 18  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " "
## 19  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " "
## 20  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " "
## 21  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " "
## 22  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " "
## 23  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " "
## 24  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " " " " " " " " " " " " " " "
## 25  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " " " " " " " " " " "
## 26  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 27  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 28  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 29  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 30  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" " " " " " "
## 31  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" " " " " " "
## 32  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" " " " " " "
## 33  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" " " " " " "
## 34  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" " " " " " "
## 35  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" " " " " " "
## 36  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" "*" " " " "
## 37  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" "*" " " " "
## 38  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" "*" " " " "
## 39  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" " " " "
## 40  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" " " " "
## 41  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" " " " "
## 42  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" " " " "
## 43  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" " " " "
## 44  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" "*" " "
## 45  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" "*" "*"
## 46  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" "*" "*"
## 47  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" "*" "*"
## 48  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" "*" "*"
## 49  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" "*" "*"
## 50  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" "*" "*"
## 51  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " "*" " " "*" "*" "*" "*" "*"
## 52  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 53  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 54  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 55  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 56  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 57  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 58  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 59  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 60  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 61  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 62  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 63  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 64  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 65  ( 1 ) "*" " " " " "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 66  ( 1 ) "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 67  ( 1 ) "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 68  ( 1 ) "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 69  ( 1 ) "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
```

```
## 70  ( 1 ) "*" " " "*" "*" "*" "*" " " " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 71  ( 1 ) "*" " " "*" "*" "*" "*" " " " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 72  ( 1 ) "*" " " "*" "*" "*" "*" " " " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 73  ( 1 ) "*" " " "*" "*" "*" "*" " " " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 74  ( 1 ) "*" " " "*" "*" "*" "*" " " " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 75  ( 1 ) "*" "*" "*" "*" "*" "*" " " " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 76  ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 77  ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 78  ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*" " " " " "*" "*" "*" "*" "*"
## 79  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*" "*" "*"
## 80  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 81  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 82  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 83  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 84  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 85  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
##           V35 V36 V37 V38 V39 V40 V41 V42 V43 V44 V45 V46 V47 V48 V49 V50 V51
## 1   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " "
## 2   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " "
## 3   ( 1 )  " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " "*" " " " " " " " " "
## 4   ( 1 )  " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "*" " " " " " " " " "
## 5   ( 1 )  " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "*" " " " " " " " " "
## 6   ( 1 )  " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "*" " " " " " " " " "
## 7   ( 1 )  " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "*" " " " " " " " " "
## 8   ( 1 )  " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "*" " " " " " " " " "
## 9   ( 1 )  " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "*" " " " " " " " " "
## 10  ( 1 ) " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "*" " " " " " " " " "
## 11  ( 1 ) " " " " " " " " " " " " " " " " " " "*" "*" " " " " "*" "*" " " " " " " " " "
## 12  ( 1 ) " " " " " " " " " " " " " " " " " " "*" "*" " " " " "*" "*" " " " " " " " " "
## 13  ( 1 ) " " " " " " " " " " " " " " " " " " "*" "*" " " " " "*" "*" " " " " " " " " "
## 14  ( 1 ) " " " " " " " " " " " " " " " " " "*" " " " " "*" "*" " " " " "*" "*" " " " " " " " " "
## 15  ( 1 ) " " " " " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 16  ( 1 ) " " " " " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 17  ( 1 ) " " " " " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 18  ( 1 ) " " " " " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 19  ( 1 ) " " " " " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 20  ( 1 ) " " "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 21  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 22  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 23  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 24  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 25  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 26  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 27  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 28  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 29  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 30  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 31  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 32  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 33  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " " " "
## 34  ( 1 ) "*" "*" " " " " " " " " " " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " "*" " "
## 35  ( 1 ) "*" "*" " " " " " " "*" " " " " "*" "*" "*" "*" " " " " "*" "*" " " " " " " "*" " "
```

```
## 36  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" " " "
## 37  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" " " "
## 38  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" " " "
## 39  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" " " "
## 40  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" " " "
## 41  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" " " "
## 42  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 43  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 44  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 45  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 46  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 47  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 48  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 49  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 50  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 51  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 52  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 53  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 54  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 55  ( 1 ) "*" "*" " " " " " " "*" " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 56  ( 1 ) "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" " " " "*" "*" " " " " " " "*" "*" "
## 57  ( 1 ) "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " " " "*" "*" "
## 58  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " " " "*" "*" "
## 59  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " " " "*" "*" "
## 60  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " " " "*" "*" "
## 61  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " " " "*" "*" "
## 62  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " " " "*" "*" "
## 63  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " " " "*" "*" "
## 64  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 65  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 66  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 67  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 68  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 69  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 70  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 71  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 72  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 73  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 74  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 75  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 76  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 77  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 78  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 79  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 80  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*" "*" "*" "
## 81  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "
## 82  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "
## 83  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "
## 84  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "
## 85  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "
##           V52 V53 V54 V55 V56 V57 V58 V59 V60 V61 V62 V63 V64 V65 V66 V67 V68
## 1  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
```

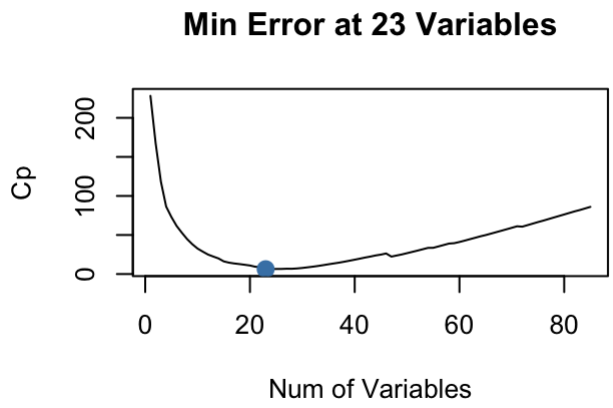
```
## 2  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 3  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 4  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 5  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 6  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 7  ( 1 )  " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 8  ( 1 )  " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 9  ( 1 )  " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 10  ( 1 ) " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 11  ( 1 ) " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 12  ( 1 ) " " " " " " " " " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 13  ( 1 ) " " " " " " " " " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 14  ( 1 ) " " " " " " " " " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 15  ( 1 ) " " " " " " " " " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 16  ( 1 ) " " " " " " " " " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 17  ( 1 ) " " " " " " " " " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 18  ( 1 ) " " " " " " " " " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 19  ( 1 ) " " " " " " " " " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 20  ( 1 ) " " " " " " " " " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 21  ( 1 ) " " " " " " " " " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 22  ( 1 ) " " " " " " " " " " "*" "*" "*" " " " " " " " " " " " " " " " " " "
## 23  ( 1 ) " " " " " " " " " " "*" "*" "*" " " " " " " " " " " " " " " " " " "
## 24  ( 1 ) " " " " " " " " " " "*" "*" "*" " " " " " " " " " " " " " " " " " "
## 25  ( 1 ) " " " " " " " " " " "*" "*" "*" " " " " " " " " " " " " " " " " " "
## 26  ( 1 ) " " " " " " " " " " "*" "*" "*" " " " " " " " " " " " " " " " " " "
## 27  ( 1 ) " " " " " " " " " " "*" "*" "*" " " " " " " " " " " " " " " " " " "
## 28  ( 1 ) " " " " " " " " " " "*" "*" "*" "*" " " " " " " " " " " " " " " " "
## 29  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" " " " " " " " " " " " " " " " "
## 30  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" " " " " " " " " " " " " " " " "
## 31  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" " " " " " " " " "*" " " " " " "
## 32  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 33  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 34  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 35  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 36  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 37  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 38  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 39  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 40  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 41  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 42  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 43  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 44  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 45  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 46  ( 1 ) " " "*" " " " " " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 47  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 48  ( 1 ) " " "*" " " "*" " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 49  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 50  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " " " " " "*" " " " " " "
## 51  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " " " "*" "*" " " " " " "
## 52  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " " " "*" "*" " " " " " "
## 53  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " " " "*" "*" " " " " " "
```

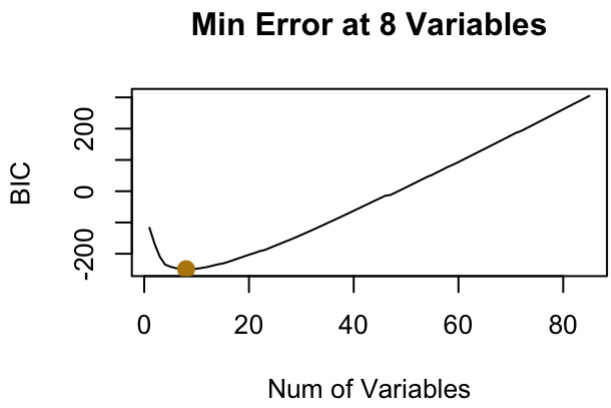
```
## 54  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 55  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 56  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 57  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 58  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 59  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 60  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 61  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 62  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 63  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 64  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 65  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 66  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 67  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 68  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " " "
## 69  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " " "
## 70  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " " "
## 71  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " " "
## 72  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " " "
## 73  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "
## 74  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "
## 75  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "
## 76  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "
## 77  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "
## 78  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "
## 79  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "
## 80  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "
## 81  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "
## 82  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "
## 83  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "
## 84  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 85  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
##           V69 V70 V71 V72 V73 V74 V75 V76 V77 V78 V79 V80 V81 V82 V83 V84 V85
## 1   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 2   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 3   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 4   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 5   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 6   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 7   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 8   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 9   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 10  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 11  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 12  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 13  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 14  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 15  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 16  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 17  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 18  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 19  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
```

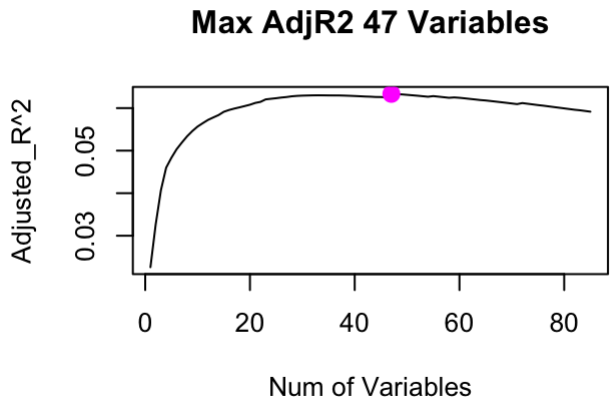
```
## 20  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" " " "*" "*" " " "*"
## 21  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" " " "*" "*" " " "*"
## 22  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" " " "*" "*" " " "*"
## 23  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" " " "*" "*" " " "*"
## 24  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" " " "*" "*" " " "*"
## 25  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" " " "*" "*" " " "*"
## 26  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" " " "*" "*" " " "*"
## 27  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 28  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 29  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 30  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 31  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 32  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 33  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 34  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 35  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 36  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 37  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 38  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 39  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 40  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 41  ( 1 ) " " " " " " " " "*" " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 42  ( 1 ) " " " " " " " " "*" " " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 43  ( 1 ) " " " " " " " "*" "*" " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 44  ( 1 ) " " " " " " " "*" "*" " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 45  ( 1 ) " " " " " " " "*" "*" " " " " " " " " " " " " "*" "*" "*" "*" "*" "*" " " "*"
## 46  ( 1 ) " " " " " " " "*" "*" " " " " " " " "*" " " " "*" "*" "*" "*" "*" "*" " " "*"
## 47  ( 1 ) " " " " " " " "*" "*" " " " " " " " "*" " " " "*" "*" "*" "*" "*" "*" " " "*"
## 48  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" " " " "*" "*" "*" "*" "*" "*" " " "*"
## 49  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" " " " "*" "*" "*" "*" "*" "*" " " "*"
## 50  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" " " " "*" "*" "*" "*" "*" "*" " " "*"
## 51  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" " " " "*" "*" "*" "*" "*" "*" " " "*"
## 52  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" " " " "*" "*" "*" "*" "*" "*" " " "*"
## 53  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" " " "*"
## 54  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" " " "*"
## 55  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 56  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 57  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 58  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 59  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 60  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 61  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 62  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 63  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 64  ( 1 ) "*" " " " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 65  ( 1 ) "*" "*" " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 66  ( 1 ) "*" "*" " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 67  ( 1 ) "*" "*" " " " "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 68  ( 1 ) "*" "*" "*" "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 69  ( 1 ) "*" "*" "*" "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 70  ( 1 ) "*" "*" "*" "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 71  ( 1 ) "*" "*" "*" "*" "*" " " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
```

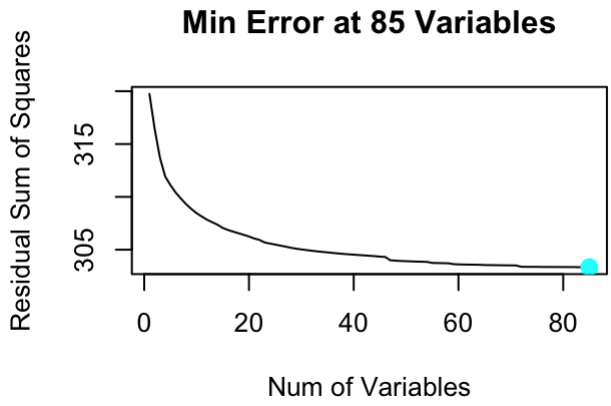
```
## 72  ( 1 ) "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 73  ( 1 ) "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 74  ( 1 ) "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 75  ( 1 ) "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 76  ( 1 ) "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 77  ( 1 ) "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 78  ( 1 ) "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 79  ( 1 ) "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 80  ( 1 ) "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 81  ( 1 ) "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 82  ( 1 ) "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 83  ( 1 ) "*" "*" "*" "*" "*" " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 84  ( 1 ) "*" "*" "*" "*" "*" " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 85  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
```

```
par(mfrow = c(2, 2))
plot(my_sum_fwd$cp,xlab = "Num of Variables", ylab = "Cp", type = "l", main = paste("Min
Error at",which.min(my_sum_fwd$cp), "Variables"))
points(which.min(my_sum_fwd$cp), my_sum_fwd$cp[which.min(my_sum_fwd$cp)], col = "steelbl
ue", cex = 2, pch = 20)

plot(my_sum_fwd$bic,xlab = "Num of Variables", ylab = "BIC", type = "l", main = paste("M
in Error at",which.min(my_sum_fwd$bic), "Variables" ))
points(which.min(my_sum_fwd$bic), my_sum_fwd$bic[which.min(my_sum_fwd$bic)], col = "dark
goldenrod", cex = 2, pch = 20)

plot(my_sum_fwd$adjr2,xlab = "Num of Variables", ylab = "Adjusted_R^2", type = "l", main
= paste("Max AdjR2",which.max(my_sum_fwd$adjr2), "Variables"))
points(which.max(my_sum_fwd$adjr2), my_sum_fwd$adjr2[which.max(my_sum_fwd$adjr2)], col =
"magenta", cex = 2, pch = 20)

plot(my_sum_fwd$rss,xlab = "Num of Variables", ylab = "Residual Sum of Squares", type =
"l", main = paste("Min Error at",which.min(my_sum_fwd$rss), "Variables"))
points(which.min(my_sum_fwd$rss), my_sum_fwd$rss[which.min(my_sum_fwd$rss)], col = "cya
n", cex = 2, pch = 20)
```










```
## using adj R2 as the metrics to select variables
length(rownames(data.frame(coef(regfit.fwd,47)))[-1])
```

```
## [1] 47
```

```
paste(rownames(data.frame(coef(regfit.fwd,47)))[-1], collapse = "+")
```

```
## [1] "V2+V4+V6+V7+V8+V10+V14+V16+V18+V21+V22+V23+V28+V30+V31+V32+V33+V34+V35+V36+V39+V
41+V42+V43+V44+V46+V47+V50+V51+V53+V55+V57+V58+V59+V60+V61+V65+V72+V73+V76+V78+V79+V80+V
81+V82+V83+V85"
```

```
glm_fwd <- glm(V86 ~ V2+V4+V6+V7+V8+V10+V14+V16+V18+V21+V22+V23+V28+V30+V31+V32+V33+V34+
V35+V36+V39+V41+V42+V43+V44+V46+V47+V50+V51+V53+V55+V57+V58+V59+V60+V61+V65+V72+V73+V76+
V78+V79+V80+V81+V82+V83+V85,
    data= train_data,
    family = binomial(link = "logit"))
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(glm_fwd)
```

```
##
## Call:
## glm(formula = V86 ~ V2 + V4 + V6 + V7 + V8 + V10 + V14 + V16 +
##     V18 + V21 + V22 + V23 + V28 + V30 + V31 + V32 + V33 + V34 +
##     V35 + V36 + V39 + V41 + V42 + V43 + V44 + V46 + V47 + V50 +
##     V51 + V53 + V55 + V57 + V58 + V59 + V60 + V61 + V65 + V72 +
##     V73 + V76 + V78 + V79 + V80 + V81 + V82 + V83 + V85, family = binomial(link = "lo
git"),
##     data = train_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8214  -0.3717  -0.2504  -0.1660   3.2063
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  237.48447 6909.48808   0.034  0.97258
## V2            -0.19387    0.18924  -1.024  0.30561
## V4             0.21907    0.08809   2.487  0.01288 *
## V6            -0.05083    0.06811  -0.746  0.45553
## V7             0.05556    0.04139   1.342  0.17950
## V8             0.04015    0.06131   0.655  0.51262
## V10            0.08436    0.04302   1.961  0.04988 *
## V14           -0.03332    0.03845  -0.867  0.38621
## V16            0.08680    0.04858   1.787  0.07400 .
## V18           -0.11268    0.04329  -2.603  0.00924 **
## V21           -0.17712    0.08480  -2.089  0.03673 *
## V22            0.04708    0.03661   1.286  0.19849
## V23           -0.04635    0.04878  -0.950  0.34208
## V28            0.09422    0.04892   1.926  0.05412 .
## V30          -13.66882  501.11021  -0.027  0.97824
## V31          -13.63624  501.11021  -0.027  0.97829
## V32            0.25085    0.14333   1.750  0.08009 .
## V33            0.20102    0.13034   1.542  0.12300
## V34            0.16928    0.13429   1.261  0.20747
## V35          -13.58409  581.62522  -0.023  0.98137
## V36          -13.63647  581.62522  -0.023  0.98129
## V39           -0.01121    0.03661  -0.306  0.75953
## V41           -0.21902    0.12534  -1.747  0.08058 .
## V42            0.08941    0.05888   1.519  0.12884
## V43            0.04353    0.03796   1.147  0.25156
## V44            0.60227    0.38792   1.553  0.12053
## V46           -0.31333    0.20693  -1.514  0.12997
## V47            0.22431    0.02459   9.122  < 2e-16 ***
## V50           -3.14773  222.86878  -0.014  0.98873
## V51            0.91098    0.90285   1.009  0.31297
## V53           -5.39004  238.24765  -0.023  0.98195
## V55           -0.23481    0.11459  -2.049  0.04045 *
## V57            1.40205    1.02280   1.371  0.17044
## V58            0.91854    0.59626   1.541  0.12344
## V59            0.22450    0.07354   3.053  0.00227 **
## V60           -8.52288 1978.09037  -0.004  0.99656
```

```
## V61           -0.18598    0.31415  -0.592  0.55385
## V65           -0.90184    0.77608  -1.162  0.24521
## V72           -1.19361    1.62494  -0.735  0.46261
## V73           -0.21160    0.42051  -0.503  0.61483
## V76            0.48168    0.21730   2.217  0.02664 *
## V78           -3.15653    2.68219  -1.177  0.23926
## V79           -3.75078    3.42218  -1.096  0.27307
## V80           -0.38230    0.26868  -1.423  0.15477
## V81           10.91297 1978.09195   0.006  0.99560
## V82            2.52389    0.98355   2.566  0.01029 *
## V83            0.46755    0.20305   2.303  0.02130 *
## V85            0.50890    0.31352   1.623  0.10456
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2635.5  on 5821  degrees of freedom
## Residual deviance: 2263.0  on 5774  degrees of freedom
## AIC: 2359
##
## Number of Fisher Scoring iterations: 16
```

```
y_hat_test = predict(glm_fwd, test_data, type = 'response')
summary(y_hat_test)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.01705 0.03523 0.05811 0.07462 0.93247
```

```
r4 <- roc(as.vector(targets$V86), as.vector(y_hat_test))
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
p = coords(r4,"best",ret="ppv")
predicted.classes <- ifelse(y_hat_test > p[1,1], 1, 0)
table(predicted.classes)
```

```
## predicted.classes
##    0    1
## 3590  410
```

```
mean(predicted.classes == targets)
```

```
## [1] 0.877
```

```
confusionMatrix(table(predicted.classes, targets[,1]), positive = "1")
```

```
## Confusion Matrix and Statistics
##
##
## predicted.classes    0    1
##                 0 3430  160
##                 1  332   78
##
##                Accuracy : 0.877
##                  95% CI : (0.8664, 0.887)
##     No Information Rate : 0.9405
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1789
##
##  Mcnemar's Test P-Value : 1.265e-14
##
##             Sensitivity : 0.3277
##             Specificity : 0.9117
##          Pos Pred Value : 0.1902
##          Neg Pred Value : 0.9554
##              Prevalence : 0.0595
##          Detection Rate : 0.0195
##    Detection Prevalence : 0.1025
##       Balanced Accuracy : 0.6197
##
##        'Positive' Class : 1
##
```

backward selection

```
regfit.bwd <- regsubsets(V86~., data = train_data, nbest = 1, nvmax = ncol(train_data),
 method = "backward")
my_sum_bwd <- summary(regfit.bwd)
my_sum_bwd$outmat
```

```
##           V1  V2  V3  V4  V5  V6  V7  V8  V9  V10 V11 V12 V13 V14 V15 V16 V17
## 1  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 2  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 3  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 4  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 5  ( 1 )  " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 6  ( 1 )  " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 7  ( 1 )  " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 8  ( 1 )  " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 9  ( 1 )  " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 10  ( 1 ) " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 11  ( 1 ) " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 12  ( 1 ) " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 13  ( 1 ) " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 14  ( 1 ) " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 15  ( 1 ) " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 16  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 17  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 18  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 19  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 20  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 21  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 22  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " " "
## 23  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " "*"
## 24  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " "*"
## 25  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " "*"
## 26  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " "*"
## 27  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " "*"
## 28  ( 1 ) " " " " " " "*" " " " " " " " " " " " " " "*" " " " " " " " " " "*"
## 29  ( 1 ) " " " " " " "*" " " "*" " " " " " " " " " "*" " " " " " " " " " "*"
## 30  ( 1 ) " " " " " " "*" " " "*" " " " " " " " " " "*" " " " " " " " " " "*"
## 31  ( 1 ) "*" " " " " "*" " " "*" " " " " " " " " " "*" " " " " " " " " " "*"
## 32  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " " "*" " " " " " " " " " "*"
## 33  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " " "*" " " " " " " " " " "*"
## 34  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " " "*" " " " " " " " " " "*"
## 35  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " " "*" " " " " " " " " " "*"
## 36  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " " " "*" " " " " " " " " "*" "*"
## 37  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " " "*" "*" " " " " " " " " "*" "*"
## 38  ( 1 ) "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 39  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 40  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 41  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 42  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 43  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 44  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 45  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 46  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 47  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 48  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 49  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 50  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
## 51  ( 1 ) "*" "*" " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" "*"
```

```
## 52  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" " " " " " "*"
## 53  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" " " " " " "*"
## 54  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" " " " " " "*"
## 55  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" " " " " " "*"
## 56  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " " " " " "*" " " " " " "*"
## 57  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" " " "*" " " " " " "*"
## 58  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" " " "*" " " " " " "*"
## 59  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" " " "*" " " " " " "*"
## 60  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" " " "*" " " " " " "*"
## 61  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" " " "*" " " " " " "*"
## 62  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" " " "*" " " " " " "*"
## 63  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" " " "*" " " " " " "*"
## 64  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" " " "*" " " " " " "*"
## 65  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" "*" "*" " " " " " "*"
## 66  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" "*" "*" "*" " " " "*"
## 67  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" "*" "*" "*" " " " "*"
## 68  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" "*" "*" "*" " " " "*"
## 69  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" "*" "*" "*" " " " "*"
## 70  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" " " " " "*" "*" "*" "*" " " " "*"
## 71  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" "*" " " "*" "*" "*" "*" " " " "*"
## 72  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 73  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 74  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 75  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 76  ( 1 ) "*" "*" " " " " "*" "*" "*" " " " " " " "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 77  ( 1 ) "*" "*" "*" "*" "*" "*" " " " " " " "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 78  ( 1 ) "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 79  ( 1 ) "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 80  ( 1 ) "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 81  ( 1 ) "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 82  ( 1 ) "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 83  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 84  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
## 85  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*"
##           V18 V19 V20 V21 V22 V23 V24 V25 V26 V27 V28 V29 V30 V31 V32 V33 V34
## 1   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 2   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 3   ( 1 )  "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 4   ( 1 )  "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 5   ( 1 )  "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 6   ( 1 )  "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 7   ( 1 )  "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 8   ( 1 )  "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 9   ( 1 )  "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 10  ( 1 )  "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 11  ( 1 )  "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 12  ( 1 )  "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 13  ( 1 )  "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 14  ( 1 )  "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 15  ( 1 )  "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 16  ( 1 )  "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 17  ( 1 )  "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
```

```
## 18  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " "
## 19  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " "*" " " " " " " " "
## 20  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " "*" " " " " " " " "
## 21  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " "*" " " " " " " " "
## 22  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " "*" " " " " " " " "
## 23  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " "*" " " " " " " " "
## 24  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " "*" " " " " " " " "
## 25  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " "*" " " " " " " " "
## 26  ( 1 ) "*" " " " " "*" " " " " " " " " " " " " " " " " "*" " " " " " " " "
## 27  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " " " " " "*" " " " " " " " "
## 28  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 29  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 30  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 31  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 32  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 33  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 34  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 35  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 36  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 37  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 38  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 39  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 40  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 41  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 42  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 43  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" " " " " " " " "
## 44  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" " " " " " "
## 45  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" " " " " " "
## 46  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" " " " " " "
## 47  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" "*" " " " "
## 48  ( 1 ) "*" " " " " "*" "*" " " " " " " " " " " "*" " " "*" "*" "*" "*" " "
## 49  ( 1 ) "*" " " " " "*" "*" " " "*" " " " " " " "*" " " "*" "*" "*" "*" " "
## 50  ( 1 ) "*" " " " " "*" "*" " " "*" " " " " " " "*" " " "*" "*" "*" "*" " "
## 51  ( 1 ) "*" " " " " "*" "*" " " "*" " " " " " " "*" " " "*" "*" "*" "*" " "
## 52  ( 1 ) "*" " " " " "*" "*" " " "*" " " " " " " "*" " " "*" "*" "*" "*" " "
## 53  ( 1 ) "*" "*" " " "*" "*" " " "*" " " " " " " "*" " " "*" "*" "*" "*" " "
## 54  ( 1 ) "*" "*" " " "*" "*" " " "*" " " " " " " "*" " " "*" "*" "*" "*" " "
## 55  ( 1 ) "*" "*" " " "*" "*" " " "*" " " " " " " "*" " " "*" "*" "*" "*" "*"
## 56  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 57  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 58  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 59  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 60  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 61  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 62  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 63  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 64  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 65  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 66  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 67  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 68  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
## 69  ( 1 ) "*" "*" " " "*" "*" " " "*" " " "*" " " "*" " " "*" "*" "*" "*" "*"
```

```
## 70  ( 1 ) "*" "*" " " " " "*" "*" " " " " "*" " " " " "*" " " " " "*" " " " " "*" "*" "*" "*" "*"
## 71  ( 1 ) "*" "*" " " " " "*" "*" " " " " "*" " " " " "*" " " " " "*" " " " " "*" "*" "*" "*" "*"
## 72  ( 1 ) "*" "*" " " " " "*" "*" " " " " "*" " " " " "*" " " " " "*" " " " " "*" "*" "*" "*" "*"
## 73  ( 1 ) "*" "*" " " " " "*" "*" " " " " "*" " " " " "*" "*" "*" " " " " "*" "*" "*" "*" "*"
## 74  ( 1 ) "*" "*" " " " " "*" "*" " " " " "*" "*" "*" "*" "*" " " " " "*" "*" "*" "*" "*"
## 75  ( 1 ) "*" "*" " " " " "*" "*" " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 76  ( 1 ) "*" "*" " " " " "*" "*" " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 77  ( 1 ) "*" "*" " " " " "*" "*" " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 78  ( 1 ) "*" "*" " " " " "*" "*" " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 79  ( 1 ) "*" "*" " " " " "*" "*" " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 80  ( 1 ) "*" "*" " " " " "*" "*" " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 81  ( 1 ) "*" "*" " " " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 82  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 83  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 84  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 85  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
##           V35 V36 V37 V38 V39 V40 V41 V42 V43 V44 V45 V46 V47 V48 V49 V50 V51
## 1   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " "
## 2   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " "
## 3   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " "
## 4   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " "
## 5   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " " "
## 6   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "
## 7   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "
## 8   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "
## 9   ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "
## 10  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "
## 11  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "
## 12  ( 1 ) " " " " " " " " " " " " " " " " " " " " " "*" " " " " " " " "*" "*" " " " " " " " "
## 13  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "*" "*" " " " " " " " "
## 14  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " " " " " "*" "*" " " " " " " " "
## 15  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 16  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 17  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 18  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 19  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 20  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 21  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 22  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 23  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 24  ( 1 ) " " " " " " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 25  ( 1 ) " " "*" " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 26  ( 1 ) "*" "*" " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 27  ( 1 ) "*" "*" " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 28  ( 1 ) "*" "*" " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 29  ( 1 ) "*" "*" " " " " " " " " " " " " " " "*" "*" " " " "*" " " " "*" "*" " " " " " " " "
## 30  ( 1 ) "*" "*" " " " " " " " " " " " " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " " "
## 31  ( 1 ) "*" "*" " " " " " " " " " " " " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " " "
## 32  ( 1 ) "*" "*" " " " " " " " " " " " " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " " "
## 33  ( 1 ) "*" "*" " " " " " " " " " " " " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " " "
## 34  ( 1 ) "*" "*" " " " " " " " " " " " " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " " "
## 35  ( 1 ) "*" "*" " " " " " " " " " " " " " " "*" "*" "*" "*" " " " "*" "*" " " " " " " " "
```

```
## 36  ( 1 ) "*" "*" " " " " " " " " " " "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 37  ( 1 ) "*" "*" " " " " " " " " " " "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 38  ( 1 ) "*" "*" " " " " " " " " " " "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 39  ( 1 ) "*" "*" " " " " " " " " " " "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 40  ( 1 ) "*" "*" " " " " " " " " "*" "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 41  ( 1 ) "*" "*" " " "*" " " " " "*" "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 42  ( 1 ) "*" "*" "*" "*" " " " " "*" "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 43  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 44  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 45  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 46  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 47  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 48  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 49  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " " " " "
## 50  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " "*" " "
## 51  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " "*" "*"
## 52  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " "*" "*"
## 53  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " "*" "*"
## 54  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " "*" "*"
## 55  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " "*" "*"
## 56  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " "*" "*"
## 57  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " "*" "*"
## 58  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " " "*" "*"
## 59  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " "*" "*" "*"
## 60  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " "*" "*" "*"
## 61  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " "*" "*" "*"
## 62  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " "*" "*" "*"
## 63  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " "*" "*" "*"
## 64  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " "*" "*" "*"
## 65  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " "*" "*" "*"
## 66  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" " " " " "*" "*" "*"
## 67  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 68  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 69  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 70  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 71  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 72  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 73  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 74  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 75  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 76  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 77  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 78  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 79  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" "*"
## 80  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 81  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 82  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 83  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 84  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 85  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
##           V52 V53 V54 V55 V56 V57 V58 V59 V60 V61 V62 V63 V64 V65 V66 V67 V68
## 1  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
```

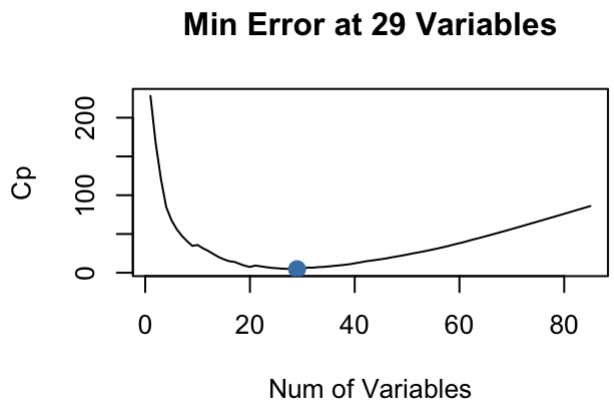
```
## 2  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 3  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 4  ( 1 )  " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 5  ( 1 )  " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 6  ( 1 )  " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 7  ( 1 )  " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 8  ( 1 )  " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 9  ( 1 )  " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 10  ( 1 ) " " " " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 11  ( 1 ) " " " " " " "*" " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 12  ( 1 ) " " " " " " "*" " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 13  ( 1 ) " " " " " " "*" " " " " " " "*" " " " " " " " " " " " " " " " " " "
## 14  ( 1 ) " " " " " " "*" " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 15  ( 1 ) " " " " " " "*" " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 16  ( 1 ) " " " " " " "*" " " " " "*" "*" " " " " " " " " " " " " " " " " " "
## 17  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " " " " " " " " " " " " "
## 18  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " " " " " " " " " " " " "
## 19  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " " " " " " " " " " " " "
## 20  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " " " " " " " " " " " " "
## 21  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 22  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 23  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 24  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 25  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 26  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 27  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 28  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 29  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 30  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 31  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 32  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 33  ( 1 ) " " " " " " "*" " " "*" "*" "*" " " " " " " "*" " " " " " " " " " "
## 34  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" " " " " "*" " " " " " " " " " "
## 35  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " " " " "
## 36  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " " " " "
## 37  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " " " " "
## 38  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " " " " "
## 39  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " " " " "
## 40  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " " " " "
## 41  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " " " " "
## 42  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " " " " "
## 43  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " " " " "
## 44  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" " " " " "*" " " "*" " " " " " "
## 45  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 46  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 47  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 48  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 49  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 50  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 51  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 52  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 53  ( 1 ) " " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
```

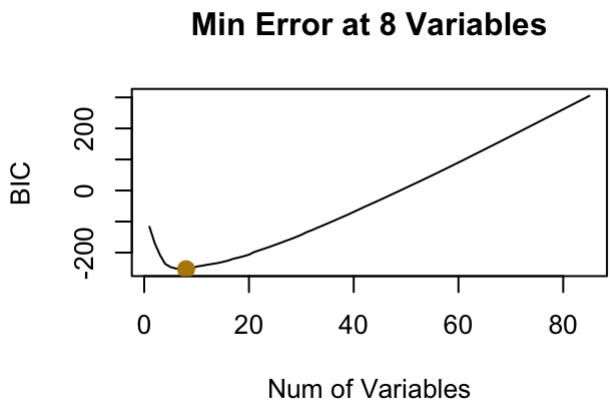
```
## 54  ( 1 ) "*" " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 55  ( 1 ) "*" " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 56  ( 1 ) "*" " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 57  ( 1 ) "*" " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" " " "*" " " " " " "
## 58  ( 1 ) "*" " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " "
## 59  ( 1 ) "*" " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " "
## 60  ( 1 ) "*" " " " " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " "
## 61  ( 1 ) "*" "*" " " "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " "
## 62  ( 1 ) "*" "*" "*" "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " "
## 63  ( 1 ) "*" "*" "*" "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " "
## 64  ( 1 ) "*" "*" "*" "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " "
## 65  ( 1 ) "*" "*" "*" "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " "
## 66  ( 1 ) "*" "*" "*" "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " "
## 67  ( 1 ) "*" "*" "*" "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" " " " " " "
## 68  ( 1 ) "*" "*" "*" "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " "
## 69  ( 1 ) "*" "*" "*" "*" " " "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " "
## 70  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " "
## 71  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " "
## 72  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " "
## 73  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " "
## 74  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " "
## 75  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*" "*" "*" "*" " " " "
## 76  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "
## 77  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "
## 78  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "
## 79  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " "
## 80  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " "
## 81  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " "
## 82  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " "
## 83  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " "
## 84  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 85  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
##           V69 V70 V71 V72 V73 V74 V75 V76 V77 V78 V79 V80 V81 V82 V83 V84 V85
## 1  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 2  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 3  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 4  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 5  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 6  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 7  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 8  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 9  ( 1 )  " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 10  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 11  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 12  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 13  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 14  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 15  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 16  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 17  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 18  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 19  ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
```

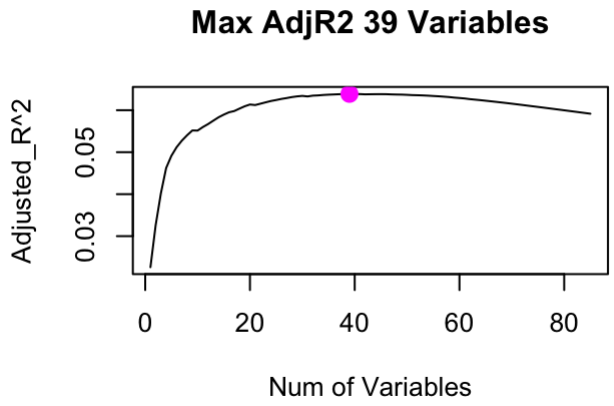
```
## 20  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" " " "*" " " "*" "*" " " "*"
## 21  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" " " "*" " " "*" "*" " " "*"
## 22  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" " " "*" " " "*" "*" "*" "*"
## 23  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" " " "*" " " "*" "*" "*" "*"
## 24  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" " " "*" "*" "*" "*"
## 25  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" " " "*" "*" "*" "*"
## 26  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" " " "*" "*" "*" "*"
## 27  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" " " "*" "*" "*" "*"
## 28  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" " " "*" "*" "*" "*"
## 29  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" " " "*" "*" "*" "*"
## 30  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" " " "*" "*" "*" "*"
## 31  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" " " "*" "*" "*" "*"
## 32  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" " " "*" "*" "*" "*"
## 33  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 34  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 35  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 36  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 37  ( 1 ) " " " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 38  ( 1 ) "*" " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 39  ( 1 ) "*" " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 40  ( 1 ) "*" " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 41  ( 1 ) "*" " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 42  ( 1 ) "*" " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 43  ( 1 ) "*" " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 44  ( 1 ) "*" " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 45  ( 1 ) "*" " " " " " " " " " " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 46  ( 1 ) "*" " " " " " " " "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 47  ( 1 ) "*" " " " " " " " "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 48  ( 1 ) "*" " " " " " " " "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 49  ( 1 ) "*" " " " " " " " "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 50  ( 1 ) "*" " " " " " " " "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 51  ( 1 ) "*" " " " " " " " "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 52  ( 1 ) "*" " " " " " "*" "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 53  ( 1 ) "*" " " " " " "*" "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 54  ( 1 ) "*" " " " " " "*" "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 55  ( 1 ) "*" " " " " " "*" "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 56  ( 1 ) "*" " " " " " "*" "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 57  ( 1 ) "*" " " " " " "*" "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 58  ( 1 ) "*" " " " " " "*" "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 59  ( 1 ) "*" " " " " " "*" "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 60  ( 1 ) "*" "*" " " "*" "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 61  ( 1 ) "*" "*" " " "*" "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 62  ( 1 ) "*" "*" " " "*" "*" " " " "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 63  ( 1 ) "*" "*" " " "*" "*" " " "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
## 64  ( 1 ) "*" "*" " " "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 65  ( 1 ) "*" "*" " " "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 66  ( 1 ) "*" "*" " " "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 67  ( 1 ) "*" "*" " " "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 68  ( 1 ) "*" "*" " " "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 69  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 70  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 71  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
```

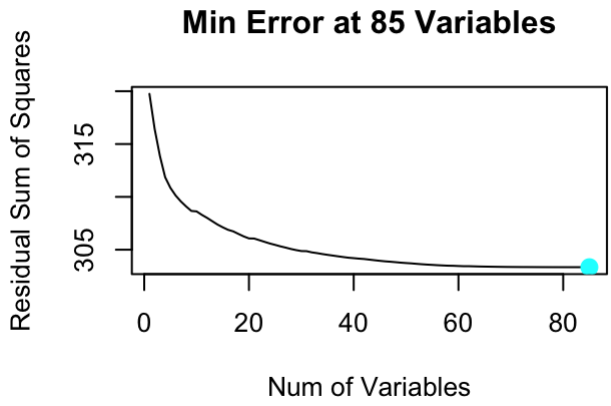
```
## 72  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 73  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 74  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 75  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 76  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 77  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 78  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 79  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 80  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 81  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 82  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 83  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 84  ( 1 ) "*" "*" "*" "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 85  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
```

```
par(mfrow = c(2, 2))
plot(my_sum_bwd$cp,xlab = "Num of Variables", ylab = "Cp", type = "l", main = paste("Min
Error at",which.min(my_sum_bwd$cp), "Variables"))
points(which.min(my_sum_bwd$cp), my_sum_bwd$cp[which.min(my_sum_bwd$cp)], col = "steelbl
ue", cex = 2, pch = 20)

plot(my_sum_bwd$bic,xlab = "Num of Variables", ylab = "BIC", type = "l", main = paste("M
in Error at",which.min(my_sum_bwd$bic), "Variables" ))
points(which.min(my_sum_bwd$bic), my_sum_bwd$bic[which.min(my_sum_bwd$bic)], col = "dark
goldenrod", cex = 2, pch = 20)

plot(my_sum_bwd$adjr2,xlab = "Num of Variables", ylab = "Adjusted_R^2", type = "l", main
= paste("Max AdjR2",which.max(my_sum_bwd$adjr2), "Variables"))
points(which.max(my_sum_bwd$adjr2), my_sum_bwd$adjr2[which.max(my_sum_bwd$adjr2)], col =
"magenta", cex = 2, pch = 20)

plot(my_sum_bwd$rss,xlab = "Num of Variables", ylab = "Residual Sum of Squares", type =
"l", main = paste("Min Error at",which.min(my_sum_bwd$rss), "Variables"))
points(which.min(my_sum_bwd$rss), my_sum_bwd$rss[which.min(my_sum_bwd$rss)], col = "cya
n", cex = 2, pch = 20)
```










```
## using adj R2 as the metrics to select variables
length(rownames(data.frame(coef(regfit.bwd,39)))[-1])
```

```
## [1] 39
```

```
paste(rownames(data.frame(coef(regfit.bwd,39)))[-1], collapse = "+")
```

```
## [1] "V1+V2+V4+V5+V6+V9+V10+V14+V17+V18+V21+V22+V28+V30+V35+V36+V41+V42+V43+V44+V46+V4
7+V55+V57+V58+V59+V60+V63+V65+V69+V76+V78+V79+V80+V81+V82+V83+V84+V85"
```

```
glm_bwd <- glm(V86 ~ V1+V2+V4+V5+V6+V9+V10+V14+V17+V18+V21+V22+V28+V30+V35+V36+V41+V42+V
43+V44+V46+V47+V55+V57+V58+V59+V60+V63+V65+V69+V76+V78+V79+V80+V81+V82+V83+V84+V85,
    data= train_data,
    family = binomial(link = "logit"))
summary(glm_bwd)
```

```
##
## Call:
## glm(formula = V86 ~ V1 + V2 + V4 + V5 + V6 + V9 + V10 + V14 +
##     V17 + V18 + V21 + V22 + V28 + V30 + V35 + V36 + V41 + V42 +
##     V43 + V44 + V46 + V47 + V55 + V57 + V58 + V59 + V60 + V63 +
##     V65 + V69 + V76 + V78 + V79 + V80 + V81 + V82 + V83 + V84 +
##     V85, family = binomial(link = "logit"), data = train_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7926  -0.3698  -0.2512  -0.1681   3.1833
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  120.59416 3435.29958   0.035 0.971996
## V1             0.07093    0.04539   1.563 0.118128
## V2            -0.21617    0.18764  -1.152 0.249313
## V4             0.22035    0.08443   2.610 0.009059 **
## V5            -0.30258    0.20312  -1.490 0.136310
## V6            -0.08611    0.06201  -1.389 0.164908
## V9            -0.04159    0.03718  -1.119 0.263339
## V10            0.10575    0.03929   2.691 0.007120 **
## V14           -0.04195    0.03749  -1.119 0.263087
## V17           -0.08724    0.05032  -1.734 0.082983 .
## V18           -0.20522    0.05557  -3.693 0.000221 ***
## V21           -0.17562    0.08451  -2.078 0.037696 *
## V22            0.06218    0.03380   1.840 0.065825 .
## V28            0.07027    0.04431   1.586 0.112762
## V30           -0.03204    0.02452  -1.307 0.191336
## V35          -13.88694  381.69996  -0.036 0.970978
## V36          -13.92713  381.69996  -0.036 0.970894
## V41           -0.23590    0.12080  -1.953 0.050835 .
## V42            0.09213    0.05668   1.625 0.104063
## V43            0.07042    0.04449   1.583 0.113486
## V44            0.61447    0.38821   1.583 0.113462
## V46           -0.34397    0.18276  -1.882 0.059825 .
## V47            0.22609    0.02455   9.210  < 2e-16 ***
## V55           -0.24818    0.11459  -2.166 0.030334 *
## V57            1.42340    1.02297   1.391 0.164092
## V58            0.84166    0.58293   1.444 0.148787
## V59            0.23012    0.07416   3.103 0.001915 **
## V60           -7.83019 1199.77277  -0.007 0.994793
## V63           -1.16809    0.90756  -1.287 0.198072
## V65           -0.95204    0.77690  -1.225 0.220410
## V69           -0.79466    0.72217  -1.100 0.271164
## V76            0.51467    0.21822   2.358 0.018349 *
## V78           -3.18889    2.68674  -1.187 0.235266
## V79           -3.43099    3.35048  -1.024 0.305821
## V80           -0.38687    0.27120  -1.426 0.153725
## V81            9.76832 1199.77590   0.008 0.993504
## V82            1.93481    0.39432   4.907 9.26e-07 ***
## V83            0.45372    0.20285   2.237 0.025302 *
```

```
## V84            2.07003    1.44598   1.432 0.152265
## V85            0.52027    0.31331   1.661 0.096808 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2635.5  on 5821  degrees of freedom
## Residual deviance: 2269.8  on 5782  degrees of freedom
## AIC: 2349.8
##
## Number of Fisher Scoring iterations: 15
```

```
y_hat_test = predict(glm_bwd, test_data, type = 'response')
summary(y_hat_test)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.01721 0.03552 0.05803 0.07374 0.90701
```

```
r4 <- roc(as.vector(targets$V86), as.vector(y_hat_test))
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
p = coords(r4,"best",ret="ppv")
predicted.classes <- ifelse(y_hat_test > p[1,1], 1, 0)
table(predicted.classes)
```

```
## predicted.classes
##    0    1
## 3551  449
```

```
mean(predicted.classes == targets)
```

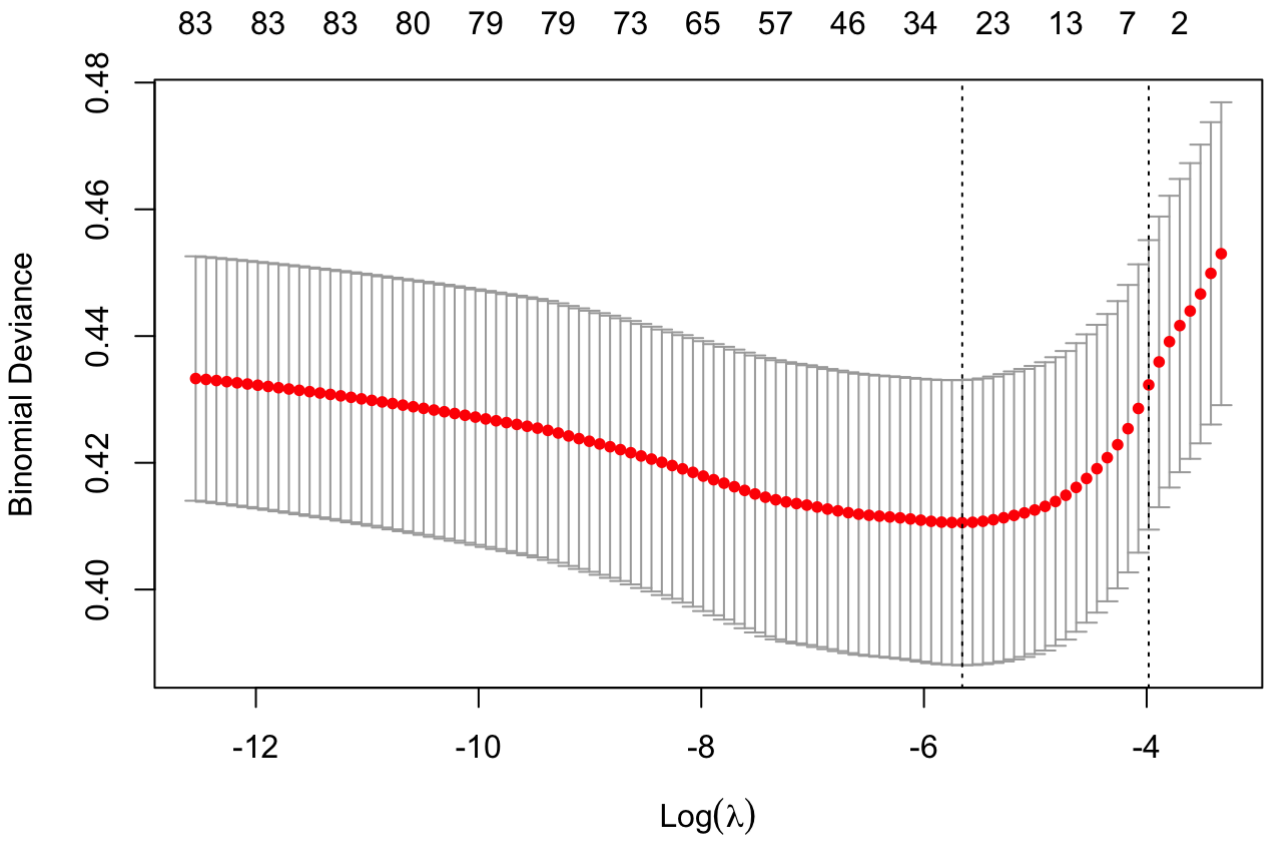
```
## [1] 0.87025
```

```
confusionMatrix(table(predicted.classes, targets[,1]), positive = "1")
```

```
## Confusion Matrix and Statistics
##
##
## predicted.classes    0    1
##                 0 3397  154
##                 1  365   84
##
##                Accuracy : 0.8702
##                  95% CI : (0.8594, 0.8805)
##     No Information Rate : 0.9405
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1808
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.3529
##             Specificity : 0.9030
##          Pos Pred Value : 0.1871
##          Neg Pred Value : 0.9566
##              Prevalence : 0.0595
##          Detection Rate : 0.0210
##    Detection Prevalence : 0.1123
##       Balanced Accuracy : 0.6280
##
##        'Positive' Class : 1
##
```

lasso

```
train_x <- model.matrix(V86 ~ . , data = train_data)[, -1]
train_y <- train_data$V86

set.seed(1234)

cv.lasso <- cv.glmnet(train_x, train_y, alpha = 1, family = "binomial")
plot(cv.lasso)
```




```
lasso_min_lambda <- cv.lasso$lambda.min
# coef(cv.lasso_3, cv.lasso_3$lambda.min)
lasso_min_lambda
```

```
## [1] 0.003495312
```

```
lasso_model <- glmnet(train_x, train_y, alpha = 1, family = "binomial", lambda = lasso_m
in_lambda)
coef(lasso_model)[,1]
```

```
##  (Intercept)           V1           V2           V3           V4           V5
## -4.748054219  0.000000000  0.000000000  0.000000000  0.020122809  0.000000000
##           V6           V7           V8           V9          V10          V11
##  0.000000000  0.019162374  0.000000000 -0.006438858  0.048078784 -0.010684081
##          V12          V13          V14          V15          V16          V17
##  0.000000000  0.000000000  0.000000000  0.000000000  0.046059986  0.000000000
##          V18          V19          V20          V21          V22          V23
## -0.050286855  0.000000000  0.000000000 -0.115173896  0.025048267  0.000000000
##          V24          V25          V26          V27          V28          V29
##  0.000000000  0.000000000  0.000000000  0.000000000  0.000000000  0.000000000
##          V30          V31          V32          V33          V34          V35
## -0.017975410  0.000000000  0.044478705  0.000000000  0.000000000  0.000000000
##          V36          V37          V38          V39          V40          V41
##  0.000000000 -0.003426102  0.000000000  0.000000000  0.015814191 -0.069311503
##          V42          V43          V44          V45          V46          V47
##  0.046708136  0.041777132  0.123041685  0.000000000 -0.106446896  0.198520224
##          V48          V49          V50          V51          V52          V53
##  0.000000000  0.000000000  0.000000000  0.000000000  0.000000000  0.000000000
##          V54          V55          V56          V57          V58          V59
##  0.000000000  0.000000000  0.000000000  0.072372809  0.133727195  0.098179247
##          V60          V61          V62          V63          V64          V65
##  0.000000000  0.000000000  0.002901700  0.000000000  0.000000000  0.000000000
##          V66          V67          V68          V69          V70          V71
##  0.000000000  0.000000000  0.000000000  0.000000000  0.000000000  0.000000000
##          V72          V73          V74          V75          V76          V77
##  0.000000000 -0.036258735  0.000000000  0.000000000  0.000000000  0.000000000
##          V78          V79          V80          V81          V82          V83
##  0.000000000  0.000000000  0.000000000  0.647308468  1.791192578  0.307057136
##          V84          V85
##  0.000000000  0.375176762
```

lasso will penalize the coefficient to 0

```
lasso_coef <- data.frame("coef" = coef(lasso_model)[,1])
lasso_var <- paste(rownames(lasso_coef[lasso_coef$coef != 0, , drop=F][-1, ,drop=F]) , c
ollapse = "+", sep = "")
lasso_var
```

```
## [1] "V4+V7+V9+V10+V11+V16+V18+V21+V22+V30+V32+V37+V40+V41+V42+V43+V44+V46+V47+V57+V58
+V59+V62+V73+V81+V82+V83+V85"
```

```
glm_lasso <- glm(V86 ~ V4+V7+V9+V10+V11+V16+V18+V21+V22+V30+V32+V37+V40+V41+V42+V43+V44+
V46+V47+V57+V58+V59+V62+V73+V81+V82+V83+V85
                  , data = train_data,
                  family = binomial(link = "logit"))
glm_lasso_sum <- summary(glm_lasso)
glm_lasso_sum
```

```
##
## Call:
## glm(formula = V86 ~ V4 + V7 + V9 + V10 + V11 + V16 + V18 + V21 +
##     V22 + V30 + V32 + V37 + V40 + V41 + V42 + V43 + V44 + V46 +
##     V47 + V57 + V58 + V59 + V62 + V73 + V81 + V82 + V83 + V85,
##     family = binomial(link = "logit"), data = train_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5642  -0.3732  -0.2545  -0.1731   3.1995
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.6155557  0.7811251  -7.189 6.52e-13 ***
## V4           0.1319072  0.0814566   1.619 0.105371
## V7           0.0492915  0.0513488   0.960 0.337088
## V9          -0.0008437  0.0526909  -0.016 0.987225
## V10          0.0560935  0.0460060   1.219 0.222744
## V11         -0.0373562  0.0829601  -0.450 0.652500
## V16          0.0632904  0.0455135   1.391 0.164351
## V18         -0.0546794  0.0362356  -1.509 0.131300
## V21         -0.1855011  0.0811876  -2.285 0.022322 *
## V22          0.0584699  0.0326030   1.793 0.072910 .
## V30         -0.0233741  0.0250309  -0.934 0.350403
## V32          0.0562971  0.0439107   1.282 0.199814
## V37         -0.0209369  0.0445846  -0.470 0.638641
## V40          0.0565114  0.0607543   0.930 0.352287
## V41         -0.2386951  0.1237698  -1.929 0.053788 .
## V42          0.0379688  0.0772155   0.492 0.622914
## V43          0.0443467  0.0364189   1.218 0.223344
## V44          0.1210745  0.0736733   1.643 0.100301
## V46         -0.2625910  0.2020034  -1.300 0.193624
## V47          0.2295506  0.0242029   9.484  < 2e-16 ***
## V57          0.1807275  0.1901201   0.951 0.341809
## V58          0.2385385  0.1022127   2.334 0.019609 *
## V59          0.1340980  0.0396497   3.382 0.000719 ***
## V62          0.5302408  0.8098783   0.655 0.512650
## V73         -0.3034708  0.3987833  -0.761 0.446662
## V81          1.5194948  1.3798320   1.101 0.270802
## V82          2.0544848  0.3844581   5.344 9.10e-08 ***
## V83          0.1505632  0.5558923   0.271 0.786507
## V85          0.4729622  0.3087898   1.532 0.125605
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2635.5  on 5821  degrees of freedom
## Residual deviance: 2300.4  on 5793  degrees of freedom
## AIC: 2358.4
##
## Number of Fisher Scoring iterations: 6
```

```
y_hat_test = predict(glm_lasso, test_data, type = 'response')
summary(y_hat_test)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.002023 0.018122 0.036483 0.058670 0.074608 0.909459
```

```
r4 <- roc(as.vector(targets$V86), as.vector(y_hat_test))
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
p = coords(r4,"best",ret="ppv")
predicted.classes <- ifelse(y_hat_test > p[1,1], 1, 0)
table(predicted.classes)
```

```
## predicted.classes
##    0    1
## 3442  558
```

```
mean(predicted.classes == targets)
```

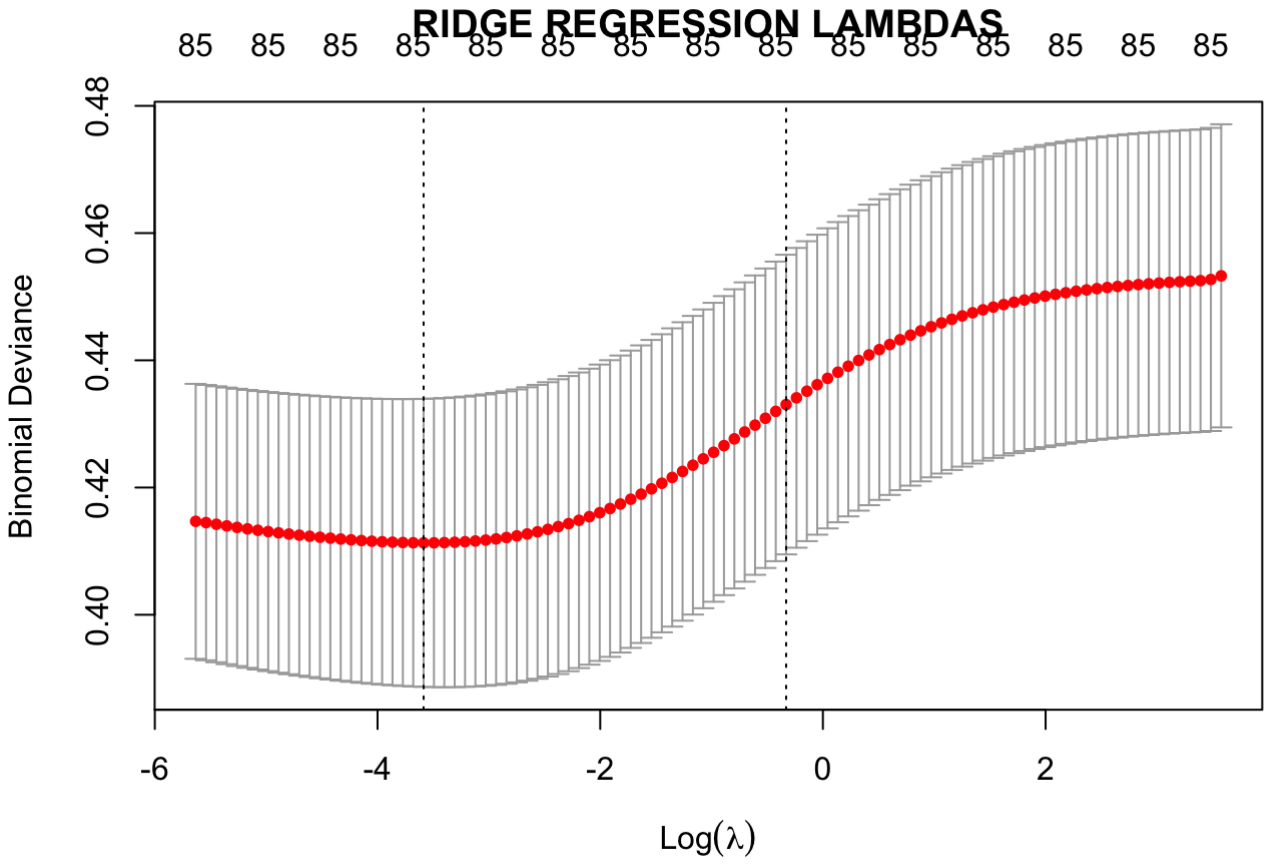
```
## [1] 0.8455
```

```
confusionMatrix(table(predicted.classes, targets[,1]), positive = "1")
```

```
## Confusion Matrix and Statistics
##
##
## predicted.classes    0    1
##                 0 3293  149
##                 1  469   89
##
##                Accuracy : 0.8455
##                  95% CI : (0.8339, 0.8566)
##     No Information Rate : 0.9405
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.153
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.37395
##             Specificity : 0.87533
##          Pos Pred Value : 0.15950
##          Neg Pred Value : 0.95671
##              Prevalence : 0.05950
##          Detection Rate : 0.02225
##    Detection Prevalence : 0.13950
##       Balanced Accuracy : 0.62464
##
##        'Positive' Class : 1
##
```

Ridge Regression

```
set.seed(1234)

cv.ridge <- cv.glmnet(train_x, train_y, alpha = 0, family = "binomial")

plot(cv.ridge, main = "RIDGE REGRESSION LAMBDAS")
```



We choose the optimal value of lambda for Regressing which is one standard error away from the maximum AUC.

```
ridge_min_lambda <- cv.ridge$lambda.min
ridge_min_lambda
```

```
## [1] 0.02769975
```

```
ridge_model <- glmnet(train_x, train_y, alpha = 0, family = "binomial", lambda = ridge_m
in_lambda)
coefficients(ridge_model)[,1]
```

```
##   (Intercept)            V1            V2            V3            V4
## -4.7888587497 -0.0001140872 -0.0980728425 -0.0177057004  0.0930721164
##            V5            V6            V7            V8            V9
## -0.0056647134 -0.0391214660  0.0216705473  0.0269619102 -0.0209295040
##           V10           V11           V12           V13           V14
##  0.0297767571 -0.0447275855 -0.0088922051 -0.0072511506 -0.0154157831
##           V15           V16           V17           V18           V19
##  0.0094036213  0.0558142734  0.0147845923 -0.0376324657  0.0154291985
##           V20           V21           V22           V23           V24
##  0.0130138249 -0.0988251664  0.0438133346 -0.0071603480  0.0064785522
##           V25           V26           V27           V28           V29
##  0.0117750947  0.0029346267 -0.0049937082  0.0223532147 -0.0329941621
##           V30           V31           V32           V33           V34
## -0.0142475228  0.0122787272  0.0414502587  0.0034795274 -0.0178282911
##           V35           V36           V37           V38           V39
##  0.0089596826 -0.0128487728 -0.0124539053  0.0199242925  0.0116615195
##           V40           V41           V42           V43           V44
##  0.0502734428 -0.1184440148  0.0437210728  0.0373014556  0.0959709353
##           V45           V46           V47           V48           V49
## -0.0361523138 -0.0904082163  0.1006322080 -0.0253937272 -0.0221713003
##           V50           V51           V52           V53           V54
## -0.1063132976  0.1410838665  0.0049254047 -0.1546225401 -0.0481003627
##           V55           V56           V57           V58           V59
## -0.0576369674 -0.0734870784  0.1966413691  0.1946119104  0.0734647594
##           V60           V61           V62           V63           V64
## -0.3083586933  0.1201133675  0.2731901983 -0.1888949645  0.0499295497
##           V65           V66           V67           V68           V69
##  0.1207270361  0.0037021307 -0.2898393916  0.2888506579 -0.1679672945
##           V70           V71           V72           V73           V74
##  0.0601182118 -0.3400205742  0.0492292176 -0.1604410712 -0.2283574256
##           V75           V76           V77           V78           V79
## -0.1757855791  0.1421846174 -0.1872221650 -0.0528934139 -0.0022479352
##           V80           V81           V82           V83           V84
##  0.0342453069  1.3485311603  1.4127895878  0.2115997272  0.2646998291
##           V85
##  0.3413402999
```

ridge regression don't panel the coefficient to 0

```
y_hat_test <- predict(ridge_model ,newx = as.matrix(test_data[1:85]), type = "response")
summary(y_hat_test)
```

```
##        s0
##  Min.   :0.001884
##  1st Qu.:0.025224
##  Median :0.043195
##  Mean   :0.058438
##  3rd Qu.:0.072601
##  Max.   :0.779889
```

```
r4 <- roc(as.vector(targets$V86), as.vector(y_hat_test))
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
p = coords(r4,"best",ret="ppv")
predicted.classes <- ifelse(y_hat_test > p[1,1], 1, 0)
table(predicted.classes)
```

```
## predicted.classes
##    0    1
## 3451  549
```

```
mean(predicted.classes == targets)
```

```
## [1] 0.84825
```

```
confusionMatrix(table(predicted.classes, targets[,1]), positive = "1")
```

```
## Confusion Matrix and Statistics
##
##
## predicted.classes    0    1
##                 0 3303  148
##                 1  459   90
##
##                Accuracy : 0.8482
##                  95% CI : (0.8368, 0.8592)
##     No Information Rate : 0.9405
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1589
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.3782
##             Specificity : 0.8780
##          Pos Pred Value : 0.1639
##          Neg Pred Value : 0.9571
##              Prevalence : 0.0595
##          Detection Rate : 0.0225
##    Detection Prevalence : 0.1373
##       Balanced Accuracy : 0.6281
##
##        'Positive' Class : 1
##
```

forward:

```
          Accuracy : 0.877
       Sensitivity : 0.3277
       Specificity : 0.9117
    Pos Pred Value : 0.1902
    Neg Pred Value : 0.9554
```

backward:

```
          Accuracy : 0.8702
       Sensitivity : 0.3529
       Specificity : 0.9030
    Pos Pred Value : 0.1871
    Neg Pred Value : 0.9566
```

lasso

```
          Accuracy : 0.8455
       Sensitivity : 0.37395
       Specificity : 0.87533
    Pos Pred Value : 0.15950
    Neg Pred Value : 0.95671
```

ridge:

```
          Accuracy : 0.8482
       Sensitivity : 0.3782
       Specificity : 0.8780
    Pos Pred Value : 0.1639
    Neg Pred Value : 0.9571
```

conclusion

they have a very similar performance however in terms of the best prediction in buying the insurance (PPV) I think the forward selection wins with 0.19. And also forward selection have a better accuracy rate with 0.87.

Q3

ESL textbook exercise 2.8 modified: Compare the classification performance of linear regression and k-nearest neighbor classification on the zipcode data. In particular, consider only the 7's and 9's for this problem, and k = 1, 3, 5, 7, 9, 11, 13,15. Show the test error for each choice of k. Describe your results – are you surprised by the differences in performance?

```
# Read in the training data
X <- as.matrix(read.table("zip.train"))
y7or9 <- which(X[, 1] == 7 | X[, 1] == 9)
train_x <- X[y7or9, -1]
train_y <- ifelse(X[y7or9, 1] == 7, 1,0 )

# Read in the test data
X <- as.matrix(read.table("zip.test"))
y7or9 <- which(X[, 1] == 7 | X[, 1] == 9)
test_x <- X[y7or9, -1]
test_y <- ifelse(X[y7or9, 1] == 7, 1,0 )
```

```
# Classification by linear regression
L <- lm(train_y ~ train_x)
summary(L)
```

```
##
## Call:
## lm(formula = train_y ~ train_x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.63340 -0.08907 -0.00414  0.08596  0.90124
##
## Coefficients: (19 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.313e+12  7.157e+11   1.835 0.066799 .
## train_xV2    5.319e-02  1.070e-01   0.497 0.619108
## train_xV3   -1.299e-02  5.438e-02  -0.239 0.811284
## train_xV4    1.705e-03  3.977e-02   0.043 0.965821
## train_xV5   -4.681e-02  2.959e-02  -1.582 0.113927
## train_xV6    6.275e-02  2.671e-02   2.350 0.018968 *
## train_xV7    4.039e-03  2.419e-02   0.167 0.867414
## train_xV8    2.831e-02  2.361e-02   1.199 0.230864
## train_xV9    1.488e-02  2.534e-02   0.587 0.557383
## train_xV10  -5.520e-02  2.818e-02  -1.959 0.050419 .
## train_xV11  -1.541e-02  2.740e-02  -0.562 0.573896
## train_xV12  -1.625e-02  2.288e-02  -0.710 0.477816
## train_xV13   5.339e-02  2.108e-02   2.532 0.011481 *
## train_xV14  -2.434e-02  2.378e-02  -1.024 0.306185
## train_xV15   1.388e-02  3.400e-02   0.408 0.683264
## train_xV16   2.278e-02  5.850e-02   0.389 0.697046
## train_xV17   1.540e-01  1.365e-01   1.128 0.259396
## train_xV18  -1.288e-01  8.333e-02  -1.545 0.122576
## train_xV19  -6.598e-02  4.632e-02  -1.424 0.154645
## train_xV20   3.290e-02  3.212e-02   1.024 0.305872
## train_xV21  -3.422e-03  2.430e-02  -0.141 0.888028
## train_xV22   1.182e-02  2.235e-02   0.529 0.597120
## train_xV23   4.533e-02  2.193e-02   2.067 0.038985 *
## train_xV24   3.918e-02  2.404e-02   1.630 0.103374
## train_xV25  -1.036e-01  2.780e-02  -3.726 0.000205 ***
## train_xV26   1.207e-01  2.599e-02   4.643 3.87e-06 ***
## train_xV27  -2.311e-02  2.175e-02  -1.062 0.288370
## train_xV28   8.282e-03  1.851e-02   0.447 0.654654
## train_xV29   1.738e-02  1.843e-02   0.943 0.345907
## train_xV30   1.160e-02  2.078e-02   0.558 0.576879
## train_xV31  -5.736e-03  2.915e-02  -0.197 0.844079
## train_xV32  -7.008e-02  4.231e-02  -1.657 0.097917 .
## train_xV33  -1.644e-02  7.343e-02  -0.224 0.822868
## train_xV34   7.548e-02  6.602e-02   1.143 0.253180
## train_xV35   5.357e-03  4.107e-02   0.130 0.896264
## train_xV36   3.718e-03  2.704e-02   0.138 0.890661
## train_xV37  -2.816e-02  2.191e-02  -1.285 0.199094
## train_xV38   1.808e-03  1.965e-02   0.092 0.926703
## train_xV39  -1.430e-03  2.069e-02  -0.069 0.944913
## train_xV40   3.990e-02  2.234e-02   1.786 0.074310 .
## train_xV41  -1.438e-02  2.482e-02  -0.579 0.562388
## train_xV42   2.618e-02  2.394e-02   1.094 0.274342
```

```
## train_xV43   1.936e-02  1.994e-02   0.971 0.331934
## train_xV44  -2.136e-02  1.671e-02  -1.278 0.201459
## train_xV45   5.763e-03  1.790e-02   0.322 0.747575
## train_xV46  -4.846e-02  2.108e-02  -2.299 0.021721 *
## train_xV47   4.879e-03  2.958e-02   0.165 0.869003
## train_xV48   2.129e-02  4.842e-02   0.440 0.660265
## train_xV49   1.949e-01  8.461e-02   2.304 0.021431 *
## train_xV50  -7.901e-02  7.302e-02  -1.082 0.279542
## train_xV51  -8.370e-02  4.499e-02  -1.861 0.063080 .
## train_xV52   2.310e-02  2.662e-02   0.868 0.385581
## train_xV53  -2.611e-02  2.183e-02  -1.196 0.232008
## train_xV54  -3.013e-02  1.993e-02  -1.512 0.130854
## train_xV55  -2.673e-02  1.981e-02  -1.350 0.177452
## train_xV56   6.074e-02  2.299e-02   2.643 0.008350 **
## train_xV57   8.572e-02  2.786e-02   3.077 0.002147 **
## train_xV58  -1.151e-02  2.589e-02  -0.445 0.656541
## train_xV59  -1.651e-02  2.010e-02  -0.821 0.411684
## train_xV60   1.685e-02  1.847e-02   0.913 0.361713
## train_xV61   2.588e-02  2.035e-02   1.271 0.203881
## train_xV62   5.231e-02  2.295e-02   2.279 0.022873 *
## train_xV63  -1.427e-02  3.277e-02  -0.435 0.663376
## train_xV64   8.120e-02  5.263e-02   1.543 0.123150
## train_xV65  -4.163e-02  9.528e-02  -0.437 0.662299
## train_xV66   9.163e-02  7.420e-02   1.235 0.217119
## train_xV67   6.343e-03  4.470e-02   0.142 0.887194
## train_xV68  -7.013e-03  3.100e-02  -0.226 0.821054
## train_xV69   2.413e-02  2.340e-02   1.031 0.302567
## train_xV70   9.826e-03  1.966e-02   0.500 0.617317
## train_xV71  -2.283e-02  2.013e-02  -1.134 0.256961
## train_xV72  -2.575e-02  2.497e-02  -1.031 0.302796
## train_xV73  -7.794e-02  3.114e-02  -2.503 0.012476 *
## train_xV74  -1.055e-02  2.698e-02  -0.391 0.695781
## train_xV75  -3.047e-03  2.132e-02  -0.143 0.886395
## train_xV76   1.233e-02  2.160e-02   0.571 0.568105
## train_xV77  -1.121e-02  2.324e-02  -0.482 0.629597
## train_xV78  -1.863e-02  2.714e-02  -0.686 0.492695
## train_xV79  -1.397e-02  3.624e-02  -0.385 0.700050
## train_xV80  -1.305e-01  5.905e-02  -2.209 0.027361 *
## train_xV81   7.713e-02  1.342e-01   0.575 0.565595
## train_xV82  -1.914e-01  8.325e-02  -2.299 0.021721 *
## train_xV83   5.943e-02  5.916e-02   1.005 0.315345
## train_xV84  -2.834e-03  3.358e-02  -0.084 0.932754
## train_xV85   1.075e-02  2.501e-02   0.430 0.667325
## train_xV86  -5.042e-02  2.035e-02  -2.478 0.013384 *
## train_xV87   5.522e-02  2.180e-02   2.533 0.011442 *
## train_xV88  -5.889e-02  2.773e-02  -2.124 0.033934 *
## train_xV89  -1.745e-01  3.069e-02  -5.686 1.69e-08 ***
## train_xV90   1.319e-03  2.763e-02   0.048 0.961951
## train_xV91  -5.823e-02  2.252e-02  -2.586 0.009844 **
## train_xV92  -1.182e-02  2.517e-02  -0.470 0.638691
## train_xV93  -5.961e-02  2.661e-02  -2.240 0.025284 *
## train_xV94   3.032e-02  3.102e-02   0.978 0.328537
```

```
## train_xV95   1.184e-02  4.157e-02   0.285 0.775905
## train_xV96   7.168e-02  7.551e-02   0.949 0.342690
## train_xV97  -1.566e-01  1.639e-01  -0.955 0.339619
## train_xV98   5.850e-02  1.115e-01   0.525 0.599920
## train_xV99  -1.606e-02  5.992e-02  -0.268 0.788679
## train_xV100 -1.996e-02  3.542e-02  -0.563 0.573296
## train_xV101 -9.393e-03  2.829e-02  -0.332 0.739938
## train_xV102  1.636e-02  2.215e-02   0.739 0.460256
## train_xV103 -5.502e-02  2.505e-02  -2.196 0.028276 *
## train_xV104 -7.633e-02  3.235e-02  -2.360 0.018464 *
## train_xV105 -1.632e-02  3.397e-02  -0.480 0.630990
## train_xV106 -5.988e-02  2.583e-02  -2.318 0.020638 *
## train_xV107  3.952e-02  2.391e-02   1.653 0.098635 .
## train_xV108 -1.859e-02  2.932e-02  -0.634 0.526332
## train_xV109  2.845e-02  2.933e-02   0.970 0.332363
## train_xV110 -2.961e-02  3.310e-02  -0.894 0.371268
## train_xV111  3.215e-02  5.523e-02   0.582 0.560596
## train_xV112  1.421e-01  1.204e-01   1.180 0.238336
## train_xV113  3.533e-01  2.592e-01   1.363 0.173179
## train_xV114 -3.212e-01  2.235e-01  -1.437 0.151029
## train_xV115 -3.458e-02  8.195e-02  -0.422 0.673175
## train_xV116 -7.088e-02  4.935e-02  -1.436 0.151205
## train_xV117  4.824e-04  3.347e-02   0.014 0.988502
## train_xV118 -2.982e-02  2.698e-02  -1.105 0.269298
## train_xV119 -5.992e-02  2.933e-02  -2.043 0.041290 *
## train_xV120 -5.985e-02  3.712e-02  -1.612 0.107218
## train_xV121 -3.544e-02  3.390e-02  -1.045 0.296137
## train_xV122 -5.449e-03  2.455e-02  -0.222 0.824396
## train_xV123 -1.042e-01  2.622e-02  -3.973 7.57e-05 ***
## train_xV124  7.347e-02  3.342e-02   2.198 0.028153 *
## train_xV125 -1.292e-02  3.457e-02  -0.374 0.708631
## train_xV126  5.915e-03  4.101e-02   0.144 0.885349
## train_xV127 -1.362e-01  8.418e-02  -1.618 0.105912
## train_xV128 -1.203e-01  2.077e-01  -0.579 0.562663
## train_xV129  3.226e+00  8.103e-01   3.982 7.31e-05 ***
## train_xV130  1.528e+00  6.878e-01   2.221 0.026534 *
## train_xV131 -1.284e-01  1.440e-01  -0.892 0.372623
## train_xV132  1.288e-01  6.575e-02   1.959 0.050401 .
## train_xV133  7.437e-02  4.865e-02   1.529 0.126656
## train_xV134 -5.161e-02  3.938e-02  -1.311 0.190247
## train_xV135 -7.482e-02  4.350e-02  -1.720 0.085749 .
## train_xV136  1.418e-03  4.440e-02   0.032 0.974525
## train_xV137 -4.291e-02  3.189e-02  -1.345 0.178758
## train_xV138 -5.884e-02  2.592e-02  -2.270 0.023413 *
## train_xV139  1.438e-02  3.038e-02   0.473 0.636174
## train_xV140 -1.228e-01  3.529e-02  -3.479 0.000524 ***
## train_xV141 -8.113e-02  3.588e-02  -2.261 0.023966 *
## train_xV142 -9.471e-02  5.424e-02  -1.746 0.081079 .
## train_xV143 -1.571e-01  1.230e-01  -1.277 0.201839
## train_xV144 -1.053e+00  6.199e-01  -1.699 0.089700 .
## train_xV145 -1.712e+00  3.195e+00  -0.536 0.592221
## train_xV146 -1.381e+14  1.027e+14  -1.345 0.178983
```

```
## train_xV147  2.448e-01  4.572e-01   0.535 0.592462
## train_xV148  5.061e-01  1.476e-01   3.429 0.000629 ***
## train_xV149 -2.397e-01  7.898e-02  -3.035 0.002465 **
## train_xV150 -9.765e-02  6.656e-02  -1.467 0.142635
## train_xV151 -2.032e-02  6.757e-02  -0.301 0.763677
## train_xV152 -3.002e-02  5.556e-02  -0.540 0.589105
## train_xV153  1.949e-02  3.509e-02   0.555 0.578729
## train_xV154  3.194e-02  2.884e-02   1.108 0.268286
## train_xV155 -3.279e-02  3.335e-02  -0.983 0.325700
## train_xV156  9.288e-02  3.727e-02   2.492 0.012853 *
## train_xV157  1.124e-01  3.955e-02   2.843 0.004553 **
## train_xV158  1.499e-01  7.479e-02   2.004 0.045351 *
## train_xV159  3.421e-01  2.283e-01   1.498 0.134381
## train_xV160  2.505e+00  1.010e+00   2.480 0.013309 *
## train_xV161         NA         NA      NA       NA
## train_xV162         NA         NA      NA       NA
## train_xV163  3.563e+11  2.636e+11   1.352 0.176672
## train_xV164 -4.763e+00  1.449e+00  -3.287 0.001048 **
## train_xV165  5.623e-01  1.827e-01   3.077 0.002143 **
## train_xV166 -2.522e-01  1.228e-01  -2.054 0.040205 *
## train_xV167 -6.096e-02  9.835e-02  -0.620 0.535517
## train_xV168 -1.207e-02  5.442e-02  -0.222 0.824518
## train_xV169  9.574e-03  3.500e-02   0.274 0.784490
## train_xV170 -4.127e-02  3.481e-02  -1.185 0.236102
## train_xV171  3.756e-02  3.798e-02   0.989 0.322925
## train_xV172 -5.752e-02  3.694e-02  -1.557 0.119795
## train_xV173 -5.257e-02  5.279e-02  -0.996 0.319565
## train_xV174 -3.368e-01  1.445e-01  -2.331 0.019922 *
## train_xV175 -1.701e+00  9.006e-01  -1.888 0.059280 .
## train_xV176  1.742e+01  8.502e+00   2.048 0.040771 *
## train_xV177         NA         NA      NA       NA
## train_xV178         NA         NA      NA       NA
## train_xV179  9.807e+13  7.254e+13   1.352 0.176672
## train_xV180 -3.646e+11  2.697e+11  -1.352 0.176672
## train_xV181  1.717e+00  9.266e-01   1.853 0.064222 .
## train_xV182 -8.805e-01  2.581e-01  -3.412 0.000670 ***
## train_xV183  3.463e-01  1.082e-01   3.199 0.001419 **
## train_xV184  1.162e-01  4.955e-02   2.346 0.019154 *
## train_xV185  4.839e-02  3.829e-02   1.264 0.206565
## train_xV186  2.373e-02  4.429e-02   0.536 0.592286
## train_xV187  7.796e-03  3.784e-02   0.206 0.836791
## train_xV188  3.289e-02  3.946e-02   0.834 0.404735
## train_xV189  9.639e-02  7.676e-02   1.256 0.209498
## train_xV190  7.479e-01  2.066e-01   3.620 0.000309 ***
## train_xV191 -1.155e+01  7.511e+00  -1.538 0.124436
## train_xV192 -9.275e+00  5.781e+01  -0.160 0.872553
## train_xV193         NA         NA      NA       NA
## train_xV194         NA         NA      NA       NA
## train_xV195         NA         NA      NA       NA
## train_xV196         NA         NA      NA       NA
## train_xV197  1.043e+01  2.712e+00   3.848 0.000126 ***
## train_xV198 -8.917e-01  2.760e-01  -3.230 0.001275 **
```

```
## train_xV199 -3.964e-02  7.698e-02  -0.515 0.606669
## train_xV200  2.554e-02  4.254e-02   0.601 0.548296
## train_xV201  7.755e-02  4.124e-02   1.880 0.060359 .
## train_xV202  1.064e-01  4.249e-02   2.504 0.012415 *
## train_xV203  2.366e-02  3.223e-02   0.734 0.462946
## train_xV204  6.466e-03  3.959e-02   0.163 0.870310
## train_xV205 -1.121e-01  8.363e-02  -1.341 0.180231
## train_xV206 -3.014e-01  2.535e-01  -1.189 0.234636
## train_xV207  2.182e+00  2.177e+00   1.002 0.316424
## train_xV208 -2.140e+13  1.587e+13  -1.349 0.177649
## train_xV209         NA         NA      NA       NA
## train_xV210         NA         NA      NA       NA
## train_xV211 -9.401e+01  2.997e+01  -3.137 0.001757 **
## train_xV212 -1.053e-02  3.245e+00  -0.003 0.997410
## train_xV213 -8.794e-01  1.103e+00  -0.797 0.425365
## train_xV214  1.195e-01  1.164e-01   1.027 0.304545
## train_xV215  1.055e-02  5.380e-02   0.196 0.844579
## train_xV216 -6.878e-02  3.796e-02  -1.812 0.070331 .
## train_xV217 -4.212e-02  3.767e-02  -1.118 0.263731
## train_xV218 -1.174e-03  3.228e-02  -0.036 0.970988
## train_xV219 -1.355e-02  2.767e-02  -0.490 0.624341
## train_xV220  4.258e-02  4.060e-02   1.049 0.294483
## train_xV221  8.482e-02  8.522e-02   0.995 0.319857
## train_xV222  6.022e-03  1.933e-01   0.031 0.975146
## train_xV223  2.360e+01  8.498e+00   2.777 0.005584 **
## train_xV224 -5.627e+11  4.183e+11  -1.345 0.178925
## train_xV225         NA         NA      NA       NA
## train_xV226         NA         NA      NA       NA
## train_xV227         NA         NA      NA       NA
## train_xV228  5.390e-01  2.208e+00   0.244 0.807161
## train_xV229 -3.577e-03  2.837e-01  -0.013 0.989941
## train_xV230  2.865e-03  8.372e-02   0.034 0.972705
## train_xV231 -1.141e-02  3.979e-02  -0.287 0.774382
## train_xV232  2.404e-02  2.957e-02   0.813 0.416352
## train_xV233  4.481e-03  2.809e-02   0.160 0.873288
## train_xV234  1.473e-02  2.438e-02   0.604 0.546011
## train_xV235  8.682e-03  2.470e-02   0.352 0.725274
## train_xV236  2.758e-02  4.223e-02   0.653 0.513887
## train_xV237 -1.346e-01  8.703e-02  -1.547 0.122145
## train_xV238  8.717e-02  3.130e-01   0.279 0.780663
## train_xV239 -1.442e+01  5.242e+00  -2.752 0.006033 **
## train_xV240         NA         NA      NA       NA
## train_xV241         NA         NA      NA       NA
## train_xV242         NA         NA      NA       NA
## train_xV243         NA         NA      NA       NA
## train_xV244 -2.237e-01  2.921e-01  -0.766 0.444012
## train_xV245  7.713e-02  1.243e-01   0.621 0.534911
## train_xV246 -1.485e-02  6.940e-02  -0.214 0.830633
## train_xV247  9.279e-03  3.557e-02   0.261 0.794252
## train_xV248 -2.106e-02  2.169e-02  -0.971 0.331827
## train_xV249 -2.515e-02  1.884e-02  -1.335 0.182154
## train_xV250  1.300e-02  1.737e-02   0.748 0.454410
```

```
## train_xV251 -9.122e-03  2.261e-02  -0.403 0.686703
## train_xV252  2.002e-02  4.958e-02   0.404 0.686415
## train_xV253  1.203e-02  1.126e-01   0.107 0.914963
## train_xV254  2.976e-01  5.794e-01   0.514 0.607550
## train_xV255  6.333e+13  4.708e+13   1.345 0.178925
## train_xV256         NA         NA      NA       NA
## train_xV257         NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1716 on 1051 degrees of freedom
## Multiple R-squared:  0.904,  Adjusted R-squared:  0.8824
## F-statistic: 41.76 on 237 and 1051 DF,  p-value: < 2.2e-16
```

```
# from the result we could see that there are high collinearity and so some of the coeff
icients are NA.
# drop the NA term
L$coefficients[is.na(L$coefficients)]=0
L$coefficients
```

```
##   (Intercept)     train_xV2     train_xV3     train_xV4     train_xV5
##  1.313239e+12  5.318996e-02 -1.298811e-02  1.704599e-03 -4.681458e-02
##     train_xV6     train_xV7     train_xV8     train_xV9    train_xV10
##  6.275482e-02  4.039405e-03  2.830646e-02  1.487528e-02 -5.519650e-02
##    train_xV11    train_xV12    train_xV13    train_xV14    train_xV15
## -1.541035e-02 -1.624660e-02  5.338522e-02 -2.434028e-02  1.387531e-02
##    train_xV16    train_xV17    train_xV18    train_xV19    train_xV20
##  2.278237e-02  1.540254e-01 -1.287749e-01 -6.597925e-02  3.290169e-02
##    train_xV21    train_xV22    train_xV23    train_xV24    train_xV25
## -3.422078e-03  1.181851e-02  4.533116e-02  3.918465e-02 -1.035735e-01
##    train_xV26    train_xV27    train_xV28    train_xV29    train_xV30
##  1.206868e-01 -2.310835e-02  8.281637e-03  1.737839e-02  1.159763e-02
##    train_xV31    train_xV32    train_xV33    train_xV34    train_xV35
## -5.735559e-03 -7.007998e-02 -1.644271e-02  7.547889e-02  5.356673e-03
##    train_xV36    train_xV37    train_xV38    train_xV39    train_xV40
##  3.717763e-03 -2.815629e-02  1.808549e-03 -1.429885e-03  3.990203e-02
##    train_xV41    train_xV42    train_xV43    train_xV44    train_xV45
## -1.438218e-02  2.617974e-02  1.935751e-02 -2.136343e-02  5.763395e-03
##    train_xV46    train_xV47    train_xV48    train_xV49    train_xV50
## -4.846033e-02  4.879216e-03  2.129020e-02  1.949109e-01 -7.900588e-02
##    train_xV51    train_xV52    train_xV53    train_xV54    train_xV55
## -8.370364e-02  2.310256e-02 -2.611124e-02 -3.012532e-02 -2.672844e-02
##    train_xV56    train_xV57    train_xV58    train_xV59    train_xV60
##  6.074470e-02  8.571656e-02 -1.151468e-02 -1.650604e-02  1.684971e-02
##    train_xV61    train_xV62    train_xV63    train_xV64    train_xV65
##  2.587659e-02  5.230652e-02 -1.426598e-02  8.120311e-02 -4.162497e-02
##    train_xV66    train_xV67    train_xV68    train_xV69    train_xV70
##  9.163289e-02  6.342969e-03 -7.013093e-03  2.413389e-02  9.825916e-03
##    train_xV71    train_xV72    train_xV73    train_xV74    train_xV75
## -2.283108e-02 -2.574683e-02 -7.794463e-02 -1.055352e-02 -3.046588e-03
##    train_xV76    train_xV77    train_xV78    train_xV79    train_xV80
##  1.233432e-02 -1.121315e-02 -1.862577e-02 -1.396601e-02 -1.304621e-01
##    train_xV81    train_xV82    train_xV83    train_xV84    train_xV85
##  7.712905e-02 -1.913642e-01  5.942525e-02 -2.833838e-03  1.075334e-02
##    train_xV86    train_xV87    train_xV88    train_xV89    train_xV90
## -5.042045e-02  5.521678e-02 -5.888928e-02 -1.744859e-01  1.318542e-03
##    train_xV91    train_xV92    train_xV93    train_xV94    train_xV95
## -5.822678e-02 -1.182296e-02 -5.961084e-02  3.032321e-02  1.183564e-02
##    train_xV96    train_xV97    train_xV98    train_xV99   train_xV100
##  7.167657e-02 -1.565835e-01  5.850297e-02 -1.606331e-02 -1.995676e-02
##   train_xV101   train_xV102   train_xV103   train_xV104   train_xV105
## -9.392877e-03  1.636465e-02 -5.502096e-02 -7.633318e-02 -1.632250e-02
##   train_xV106   train_xV107   train_xV108   train_xV109   train_xV110
## -5.987598e-02  3.952474e-02 -1.858596e-02  2.844582e-02 -2.960743e-02
##   train_xV111   train_xV112   train_xV113   train_xV114   train_xV115
##  3.214880e-02  1.420750e-01  3.533239e-01 -3.211714e-01 -3.457526e-02
##   train_xV116   train_xV117   train_xV118   train_xV119   train_xV120
## -7.088004e-02  4.824427e-04 -2.981498e-02 -5.992346e-02 -5.985309e-02
##   train_xV121   train_xV122   train_xV123   train_xV124   train_xV125
## -3.543962e-02 -5.448608e-03 -1.041932e-01  7.347183e-02 -1.292314e-02
##   train_xV126   train_xV127   train_xV128   train_xV129   train_xV130
##  5.914682e-03 -1.362160e-01 -1.203013e-01  3.226362e+00  1.527914e+00
```

```
##   train_xV131   train_xV132   train_xV133   train_xV134   train_xV135
## -1.284150e-01  1.287915e-01  7.437147e-02 -5.161362e-02 -7.482164e-02
##   train_xV136   train_xV137   train_xV138   train_xV139   train_xV140
##  1.418236e-03 -4.291308e-02 -5.883658e-02  1.437731e-02 -1.227542e-01
##   train_xV141   train_xV142   train_xV143   train_xV144   train_xV145
## -8.112683e-02 -9.471261e-02 -1.571285e-01 -1.052959e+00 -1.711711e+00
##   train_xV146   train_xV147   train_xV148   train_xV149   train_xV150
## -1.381087e+14  2.448124e-01  5.061396e-01 -2.397086e-01 -9.765175e-02
##   train_xV151   train_xV152   train_xV153   train_xV154   train_xV155
## -2.032046e-02 -3.002053e-02  1.948987e-02  3.194294e-02 -3.279089e-02
##   train_xV156   train_xV157   train_xV158   train_xV159   train_xV160
##  9.288367e-02  1.124493e-01  1.498562e-01  3.420764e-01  2.505397e+00
##   train_xV161   train_xV162   train_xV163   train_xV164   train_xV165
##  0.000000e+00  0.000000e+00  3.563457e+11 -4.762714e+00  5.622775e-01
##   train_xV166   train_xV167   train_xV168   train_xV169   train_xV170
## -2.521724e-01 -6.095703e-02 -1.207077e-02  9.573671e-03 -4.127024e-02
##   train_xV171   train_xV172   train_xV173   train_xV174   train_xV175
##  3.755979e-02 -5.751695e-02 -5.257210e-02 -3.367855e-01 -1.700525e+00
##   train_xV176   train_xV177   train_xV178   train_xV179   train_xV180
##  1.741547e+01  0.000000e+00  0.000000e+00  9.806862e+13 -3.646054e+11
##   train_xV181   train_xV182   train_xV183   train_xV184   train_xV185
##  1.716578e+00 -8.804730e-01  3.462922e-01  1.162457e-01  4.839136e-02
##   train_xV186   train_xV187   train_xV188   train_xV189   train_xV190
##  2.372658e-02  7.796531e-03  3.289356e-02  9.639183e-02  7.478527e-01
##   train_xV191   train_xV192   train_xV193   train_xV194   train_xV195
## -1.154890e+01 -9.275394e+00  0.000000e+00  0.000000e+00  0.000000e+00
##   train_xV196   train_xV197   train_xV198   train_xV199   train_xV200
##  0.000000e+00  1.043493e+01 -8.917278e-01 -3.964228e-02  2.554286e-02
##   train_xV201   train_xV202   train_xV203   train_xV204   train_xV205
##  7.754617e-02  1.064085e-01  2.366353e-02  6.466032e-03 -1.121465e-01
##   train_xV206   train_xV207   train_xV208   train_xV209   train_xV210
## -3.014262e-01  2.182226e+00 -2.140107e+13  0.000000e+00  0.000000e+00
##   train_xV211   train_xV212   train_xV213   train_xV214   train_xV215
## -9.401084e+01 -1.053381e-02 -8.794221e-01  1.195331e-01  1.054906e-02
##   train_xV216   train_xV217   train_xV218   train_xV219   train_xV220
## -6.877789e-02 -4.212070e-02 -1.174260e-03 -1.355298e-02  4.258237e-02
##   train_xV221   train_xV222   train_xV223   train_xV224   train_xV225
##  8.481613e-02  6.022236e-03  2.360012e+01 -5.626641e+11  0.000000e+00
##   train_xV226   train_xV227   train_xV228   train_xV229   train_xV230
##  0.000000e+00  0.000000e+00  5.390326e-01 -3.577114e-03  2.865345e-03
##   train_xV231   train_xV232   train_xV233   train_xV234   train_xV235
## -1.140913e-02  2.404352e-02  4.481052e-03  1.472523e-02  8.682288e-03
##   train_xV236   train_xV237   train_xV238   train_xV239   train_xV240
##  2.757770e-02 -1.346459e-01  8.716846e-02 -1.442467e+01  0.000000e+00
##   train_xV241   train_xV242   train_xV243   train_xV244   train_xV245
##  0.000000e+00  0.000000e+00  0.000000e+00 -2.236686e-01  7.713121e-02
##   train_xV246   train_xV247   train_xV248   train_xV249   train_xV250
## -1.484777e-02  9.278554e-03 -2.106136e-02 -2.514850e-02  1.300185e-02
##   train_xV251   train_xV252   train_xV253   train_xV254   train_xV255
## -9.122132e-03  2.002305e-02  1.202586e-02  2.976459e-01  6.332529e+13
##   train_xV256   train_xV257
##  0.000000e+00  0.000000e+00
```

```
# Classification by linear regression

yhat <- (cbind(1, test_x) %*% L$coef) >= 0.5
L.error <- mean(yhat != test_y)
L.error
```

```
## [1] 0.0462963
```

```
# Classification by k-nearest neighbors
library(class)
k <- c(1, 3, 5, 7,9,11,13, 15)
k.error <- rep(NA, length(k))
for (i in 1:length(k)) {
    yhat <- knn(train_x, test_x, train_y, k[i])
    k.error[i] <- mean(yhat != test_y)
}
```

```
# Compare results
error <- matrix(c(L.error, k.error), ncol = 1)
colnames(error) <- c("Error Rate")
rownames(error) <- c("Linear Regression", paste("k-NN with k =", k))
error
```

```
##                   Error Rate
## Linear Regression 0.04629630
## k-NN with k = 1   0.02469136
## k-NN with k = 3   0.02469136
## k-NN with k = 5   0.02469136
## k-NN with k = 7   0.02777778
## k-NN with k = 9   0.03703704
## k-NN with k = 11  0.04012346
## k-NN with k = 13  0.04012346
## k-NN with k = 15  0.03703704
```

```
#                    Error Rate
# Linear Regression 0.04629630
# k-NN with k = 1   0.02469136
# k-NN with k = 3   0.02469136
# k-NN with k = 5   0.02469136
# k-NN with k = 7   0.02777778
# k-NN with k = 9   0.03703704
# k-NN with k = 11  0.04012346
# k-NN with k = 13  0.04012346
# k-NN with k = 15  0.03703704

# the result is not that different and the best kNN result is when k = 1 which is very s
urprising to me.

plot(c(1, 15), c(0, 1.1 * max(error)), type = "n", main = "Comparing Classifiers (Linear
vs KNN (different K))",
     ylab = "Error Rate", xlab = "k")
abline(h = L.error, col = 2, lty = 4,  lwd= 3)
points(k, k.error, col = 4)
lines(k, k.error, col = 4, lty = 2)
```