Problem Set 7, Fall 2020

Adam Ten Hoeve

02/26/2021

```r
# Load any packages, if any, that you use as part of your answers here
# For example:
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(mlbench)
library(glmnet)  # For ridge regression
```

```
## Loading required package: Matrix

##
## Attaching package: 'Matrix'

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

## Loaded glmnet 4.1
```

CONTEXT - HOUSE VALUES IN BOSTON, CIRCA 1970

This dataset was obtained through the mlbench package, which contains a subset of data sets available through the UCI Machine Learning Repository. From the help file:

Housing data for 506 census tracts of Boston from the 1970 census. The dataframe BostonHousing contains the original data by Harrison and Rubinfeld (1979).

The original data are 506 observations on 14 variables, medv being the target variable:

Continous variables:

crim per capita crime rate by town zn proportion of residential land zoned for lots over 25,000 sq.ft indus proportion of non-retail business acres per town nox nitric oxides concentration (parts per 10 million) rm average number of rooms per dwelling age proportion of owner-occupied units built prior to 1940 dis weighted distances to five Boston employment centres rad index of accessibility to radial highways tax full-value property-tax rate per USD 10,000 ptratio pupil-teacher ratio by town b 1000(B - 0.63)^2 where B is the proportion of blacks by town lstat percentage of lower status of the population medv median value of owner-occupied homes in USD 1000's

Categorical variables:

chas Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)

Question 1 - 10 points

The BostonHousing data is contained inside of an R package, so you'll load the data into memory a little differently than usual. Run the following code chunk, confirm that the data in loaded into memory, and ensure that your variables are of the proper type.

```
data(BostonHousing) # loads the BostonHousing dataset into memory from the mlbench package

# Any processing code for changing variable types

str(BostonHousing)
```

```
## 'data.frame':    506 obs. of  14 variables:
##  $ crim   : num  0.00632 0.02731 0.02729 0.03237 0.06905 ...
##  $ zn     : num  18 0 0 0 0 0 12.5 12.5 12.5 12.5 ...
##  $ indus  : num  2.31 7.07 7.07 2.18 2.18 2.18 7.87 7.87 7.87 7.87 ...
##  $ chas   : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ nox    : num  0.538 0.469 0.469 0.458 0.458 0.458 0.524 0.524 0.524 0.524 ...
##  $ rm     : num  6.58 6.42 7.18 7 7.15 ...
##  $ age    : num  65.2 78.9 61.1 45.8 54.2 58.7 66.6 96.1 100 85.9 ...
##  $ dis    : num  4.09 4.97 4.97 6.06 6.06 ...
##  $ rad    : num  1 2 2 3 3 3 5 5 5 5 ...
##  $ tax    : num  296 242 242 222 222 222 311 311 311 311 ...
##  $ ptratio: num  15.3 17.8 17.8 18.7 18.7 18.7 15.2 15.2 15.2 15.2 ...
##  $ b      : num  397 397 393 395 397 ...
##  $ lstat  : num  4.98 9.14 4.03 2.94 5.33 ...
##  $ medv   : num  24 21.6 34.7 33.4 36.2 28.7 22.9 27.1 16.5 18.9 ...
```

Next, conduct a ridge regression with cross-validation. Use medv as the outcome and all of the other variables in the data set as the predictors. Be sure to display your lambda.min, lambda.1se, and the set of coefficients associated with both of these lambdas.

```
# Your code here
set.seed(123456)
X = data.matrix(dplyr::select(BostonHousing, -medv))
Y = BostonHousing$medv
ridge.model = cv.glmnet(x=X, y=Y, alpha=0)

# Don't forget to display lambda.min, lambda.1se, and the coefficients for both of these!
ridge.model$lambda.min
```

```
## [1] 0.6777654
```

```r
coef(ridge.model, s = "lambda.min")
```

```
## 14 x 1 sparse Matrix of class "dgCMatrix"
##                       1
## (Intercept)  25.101694135
## crim         -0.087572712
## zn            0.032681030
## indus        -0.038003639
## chas          2.899781646
## nox         -11.913360447
## rm            4.011308386
## age          -0.003731470
## dis          -1.118874605
## rad           0.153730052
## tax          -0.005751054
## ptratio      -0.854984614
## b             0.009073740
## lstat        -0.472423800
```

```r
ridge.model$lambda.1se
```

```
## [1] 3.969686
```

```r
coef(ridge.model, s = "lambda.1se")
```

```
## 14 x 1 sparse Matrix of class "dgCMatrix"
##                      1
## (Intercept) 18.052547088
## crim        -0.067876332
## zn           0.020418260
## indus       -0.068746853
## chas         2.763679207
## nox         -5.413124013
## rm           3.617133985
## age         -0.007695814
## dis         -0.519007656
## rad          0.032673066
## tax         -0.002893313
## ptratio     -0.670343318
## b            0.007610745
## lstat       -0.347893248
```

Question 2 - 10 points

Next, conduct a lasso regression with cross-validation. As before, use medv as the outcome and all of the other variables in the data set as the predictors. Be sure to display your lambda.min, lambda.1se, and the set of coefficients associated with both of these lambdas.

```r
# Your code here
lasso.model = cv.glmnet(x=X, y=Y, alpha=1)

# Don't forget to display lambda.min, lambda.1se, and the coefficients for both of these!
lasso.model$lambda.min
```

```
## [1] 0.01602569
```

```r
coef(lasso.model, s="lambda.min")
```

```
## 14 x 1 sparse Matrix of class "dgCMatrix"
##                         1
## (Intercept)  32.524810123
## crim         -0.102513087
## zn            0.043276692
## indus         .
## chas          2.700442679
## nox         -16.755818227
## rm            3.840016222
## age           .
## dis          -1.436656218
## rad           0.272117316
## tax          -0.010635851
## ptratio      -0.936920066
## b             0.009138192
## lstat        -0.522541068
```

```r
lasso.model$lambda.1se
```

```
## [1] 0.2168352
```

```r
coef(lasso.model, s="lambda.1se")
```

```
## 14 x 1 sparse Matrix of class "dgCMatrix"
##                        1
## (Intercept) 20.272667900
## crim        -0.037274105
## zn           0.014107613
## indus        .
## chas         2.378231692
## nox         -8.818849049
## rm           4.234229178
## age          .
## dis         -0.767115778
## rad          .
## tax          .
## ptratio     -0.819509409
## b            0.007282875
## lstat       -0.520737188
```

Question 3 - 5 points

Did the lasso regression set any coefficients to zero? If so, note which ones were set to zero below. If not, note that these models have the same number of coefficients.

Ridge vs Lasso, lambda.min: The lasso model set two coefficients to 0, for the predictors `indus` and `age`.

Ridge vs Lasso, lambda.1se: The lasso model set five coefficients to 0, for the predictors `zn`, `indus`, `age`, `rad`, and `tax`.

CONTEXT - NATIONAL INDONESIA CONTRACEPTIVE PRELAVENCE SURVEY (1987)

This dataset was obtained from the UCI Machine Learning Repository. From the description on https://archive.ics.uci.edu/ml/datasets/Contraceptive+Method+Choice:

This dataset is a subset of the 1987 National Indonesia Contraceptive Prevalence Survey. The samples are married women who were either not pregnant or do not know if they were at the time of interview. The problem is to predict the current contraceptive method choice (no use, long-term methods, or short-term methods) of a woman based on her demographic and socio-economic characteristics.

Continuous variables

Wife's age (w.age) Number of children ever born (children)

Categorical variables:

Wife's education (w.edu) 1=low, 2, 3, 4=high Husband's education (h.edu) 1=low, 2, 3, 4=high Wife's religion (w.relig) 0=Non-Islam, 1=Islam Wife's now working? (w.work) 0=Yes, 1=No Husband's occupation (h.occ) 1, 2, 3, 4 Standard-of-living index (sol.index) 1=low, 2, 3, 4=high Media exposure (media) 0=Good, 1=Not good Contraceptive method used (contra) 1=No-use, 2=Long-term, 3=Short-term

Question 4

First, load the data set into memory and change variables into their proper type. Please re-code the contra variable such that non-use of contraception (i.e., contra=1) is equal to zero and use of contraception (i.e., contra=2 or 3) is equal to one.

```r
contra <- read.csv("contra.csv", header=TRUE, sep=",")

# Any processing code for changing variable types

contra$w.edu = as.factor(contra$w.edu)
contra$h.edu = as.factor(contra$h.edu)
contra$w.relig = as.factor(contra$w.relig)
contra$w.work = as.factor(contra$w.work)
contra$h.occ = as.factor(contra$h.occ)
contra$sol.index = as.factor(contra$sol.index)
contra$media = as.factor(contra$media)
contra$contra = as.factor(contra$contra)

str(contra)
```

```
## 'data.frame':    1473 obs. of  10 variables:
##  $ w.age    : int  24 45 43 42 36 19 38 21 27 45 ...
##  $ w.edu    : Factor w/ 4 levels "1","2","3","4": 2 1 2 3 3 4 2 3 2 1 ...
##  $ h.edu    : Factor w/ 4 levels "1","2","3","4": 3 3 3 2 3 4 3 3 3 1 ...
##  $ children : int  3 10 7 9 8 0 6 1 3 8 ...
```

```
##  $ w.relig  : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ w.work   : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 1 2 2 ...
##  $ h.occ    : Factor w/ 4 levels "1","2","3","4": 2 3 3 3 3 3 3 3 3 2 ...
##  $ sol.index: Factor w/ 4 levels "1","2","3","4": 3 4 4 3 2 3 2 2 4 2 ...
##  $ media    : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 2 ...
##  $ contra   : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 1 1 ...
```

Next, conduct a ridge regression with cross-validation. Use contra as the outcome and all of the other variables in the data set as the predictors. Be sure to display your lambda.min, lambda.1se, and the set of coefficients associated with both of these lambdas.

```
# Your code here
X.contra = model.matrix(contra~., data=contra)
X.contra = X.contra[,-1]
Y.contra = contra$contra

contra.ridge = cv.glmnet(x=X.contra, y=Y.contra, alpha=0, family="multinomial")

# Don't forget to display lambda.min, lambda.1se, and the coefficients for both of these!

contra.ridge$lambda.min
```

```
## [1] 0.01064753
```

```
coef(contra.ridge, s="lambda.min")
```

```
## $`1`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                      1
## (Intercept)  0.20817527
## w.age        0.04675161
## w.edu2      -0.08649102
## w.edu3      -0.42098337
## w.edu4      -0.89170758
## h.edu2      -0.04184945
## h.edu3      -0.16955462
## h.edu4      -0.13115000
## children    -0.21050338
## w.relig1     0.28966183
## w.work1     -0.06670754
## h.occ2       0.12663891
## h.occ3      -0.01710357
## h.occ4      -0.29538292
## sol.index2  -0.19895899
## sol.index3  -0.33767316
## sol.index4  -0.48451147
## media1       0.40310562
##
## $`2`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                      1
## (Intercept) -0.91141459
## w.age        0.00452998
```

```
## w.edu2       0.13103447
## w.edu3       0.49353605
## w.edu4       0.87219997
## h.edu2      -0.52388371
## h.edu3      -0.42257587
## h.edu4      -0.30040735
## children     0.10668575
## w.relig1    -0.24553673
## w.work1     -0.05051948
## h.occ2      -0.27760481
## h.occ3      -0.25186293
## h.occ4       0.10815938
## sol.index2   0.10244820
## sol.index3   0.27178365
## sol.index4   0.34859457
## media1      -0.24781994
##
## $`3`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                       1
## (Intercept)  0.70323932
## w.age       -0.05128159
## w.edu2      -0.04454345
## w.edu3      -0.07255267
## w.edu4       0.01950761
## h.edu2       0.56573316
## h.edu3       0.59213049
## h.edu4       0.43155735
## children     0.10381763
## w.relig1    -0.04412510
## w.work1      0.11722702
## h.occ2       0.15096590
## h.occ3       0.26896650
## h.occ4       0.18722354
## sol.index2   0.09651079
## sol.index3   0.06588951
## sol.index4   0.13591690
## media1      -0.15528568
```

```r
contra.ridge$lambda.1se
```

```
## [1] 0.07511623
```

```r
coef(contra.ridge, s="lambda.1se")
```

```
## $`1`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                        1
## (Intercept)  0.068093285
## w.age        0.029702823
## w.edu2       0.107254919
## w.edu3      -0.141903266
## w.edu4      -0.500106022
```

```
## h.edu2       0.053425474
## h.edu3      -0.090769207
## h.edu4      -0.120880705
## children    -0.136481335
## w.relig1     0.214386382
## w.work1     -0.070383075
## h.occ2       0.104398999
## h.occ3      -0.003637516
## h.occ4      -0.138191912
## sol.index2  -0.015471360
## sol.index3  -0.134080684
## sol.index4  -0.258159799
## media1       0.412674620
##
## $`2`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                        1
## (Intercept) -0.784969886
## w.age        0.006865696
## w.edu2      -0.143605004
## w.edu3       0.116872063
## w.edu4       0.443805725
## h.edu2      -0.246067115
## h.edu3      -0.117488278
## h.edu4       0.067577847
## children     0.073868383
## w.relig1    -0.208507672
## w.work1     -0.041375403
## h.occ2      -0.196107727
## h.occ3      -0.196820181
## h.occ4       0.038836819
## sol.index2  -0.055157275
## sol.index3   0.087205167
## sol.index4   0.181870755
## media1      -0.251644052
##
## $`3`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                       1
## (Intercept)  0.71687660
## w.age       -0.03656852
## w.edu2       0.03635009
## w.edu3       0.02503120
## w.edu4       0.05630030
## h.edu2       0.19264164
## h.edu3       0.20825749
## h.edu4       0.05330286
## children     0.06261295
## w.relig1    -0.00587871
## w.work1      0.11175848
## h.occ2       0.09170873
## h.occ3       0.20045770
## h.occ4       0.09935509
## sol.index2   0.07062863
```

```
## sol.index3   0.04687552
## sol.index4   0.07628904
## media1      -0.16103057
```

Question 5

Next, conduct a lasso logistic regression with cross-validation. As before, use contra as the outcome and all of the other variables in the data set as the predictors. Be sure to display your lambda.min, lambda.1se, and the set of coefficients associated with both of these lambdas.

```r
# Your code here
contra.lasso = cv.glmnet(x=X.contra, y=Y.contra, alpha=1, family="multinomial")

# Don't forget to display lambda.min, lambda.1se, and the coefficients for both of these!
contra.lasso$lambda.min
```

```
## [1] 0.0008437986
```

```r
coef(contra.lasso, s="lambda.min")
```

```
## $`1`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                       1
## (Intercept)  0.39177931
## w.age        0.04718195
## w.edu2      -0.04532555
## w.edu3      -0.36751608
## w.edu4      -0.97208683
## h.edu2       .
## h.edu3       .
## h.edu4       .
## children    -0.34650859
## w.relig1     0.35359022
## w.work1     -0.01587050
## h.occ2       .
## h.occ3       .
## h.occ4      -0.46348318
## sol.index2  -0.34749558
## sol.index3  -0.46007345
## sol.index4  -0.69188758
## media1       0.52862506
##
## $`2`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                       1
## (Intercept) -0.88925758
## w.age        .
## w.edu2       0.65036723
## w.edu3       1.10749680
## w.edu4       1.41497184
## h.edu2      -0.83346747
## h.edu3      -0.67340186
```

```
## h.edu4       -0.64791713
## children      .
## w.relig1     -0.18242685
## w.work1       .
## h.occ2       -0.41281330
## h.occ3       -0.22104623
## h.occ4        .
## sol.index2    0.05674664
## sol.index3    0.27744388
## sol.index4    0.27795849
## media1        .
##
## $`3`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                        1
## (Intercept)   0.49747827
## w.age        -0.05928515
## w.edu2        .
## w.edu3        .
## w.edu4        .
## h.edu2        1.11381348
## h.edu3        1.27599673
## h.edu4        1.06011431
## children      .
## w.relig1      .
## w.work1       0.15243738
## h.occ2        0.01976536
## h.occ3        0.29414875
## h.occ4        0.04178500
## sol.index2    .
## sol.index3    .
## sol.index4    .
## media1        .
```

```
contra.lasso$lambda.1se
```

```
## [1] 0.02189678
```

```
coef(contra.lasso, s="lambda.1se")
```

```
## $`1`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                        1
## (Intercept)   0.13583586
## w.age         0.01181708
## w.edu2        .
## w.edu3       -0.12274854
## w.edu4       -0.53865504
## h.edu2        .
## h.edu3        .
## h.edu4        .
## children     -0.19073088
## w.relig1      0.08116575
```

```
## w.work1      .
## h.occ2       .
## h.occ3       .
## h.occ4       .
## sol.index2   .
## sol.index3   .
## sol.index4  -0.15137063
## media1       0.48756360
##
## $`2`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                       1
## (Intercept) -1.30904446
## w.age        .
## w.edu2       .
## w.edu3       .
## w.edu4       0.60852212
## h.edu2      -0.04102677
## h.edu3       .
## h.edu4       0.14127163
## children     .
## w.relig1    -0.04170843
## w.work1      .
## h.occ2       .
## h.occ3       .
## h.occ4       .
## sol.index2   .
## sol.index3   .
## sol.index4   0.02082810
## media1       .
##
## $`3`
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                       1
## (Intercept)  1.17320860
## w.age       -0.05313358
## w.edu2       .
## w.edu3       .
## w.edu4       .
## h.edu2       .
## h.edu3       .
## h.edu4       .
## children     .
## w.relig1     .
## w.work1      .
## h.occ2       .
## h.occ3       0.08593819
## h.occ4       .
## sol.index2   .
## sol.index3   .
## sol.index4   .
## media1       .
```

Question 6 - 5 points

Did the lasso regression set any coefficients to zero? If so, note which ones were set to zero below. If not, note that these models have the same number of coefficients.

Ridge vs Lasso, lambda.min: 9 coefficients where set to zero with lambda.min. Those coefficients were for the predictors: `w.age`, `h.edu3`, `children`, `w.work1`, `h.occ3`, `h.occ4`, `sol.index2`, `sol.index3`, `media1`.

Ridge vs Lasso, lambda.1se: 14 coefficients where set to zero with lambda.1se. Those coefficients were for the predictors: `w.edu2`, `w.edu3`, `w.edu4`, `h.edu2`, `h.edu4`, `children`, `w.relig1`, `w.work1`, `h.occ2`, `h.occ4`, `sol.index2`, `sol.index3`, `sol.index4`, `media1`.