API 222 Problem Set 2

Machine Learning and Big Data Analytics: Fall 2024

Due at 11:59am on October 23 - submit on Gradescope

This problem set is worth 30 points in total. To get full credit, submit your code along with a write-up of your answers. This should either be done in R Markdown or Jupyter Notebook, submitted in one knitted PDF.

Final Project Groups (0 pts)

Please join one of 30 final project groups that have been created on this Canvas page.

Details about the final project can be found on this Canvas page. You just need to form your group by October 23. The main project milestones will be after the midterm. We recommend forming groups of 5 students. All students working together should join the same group. PhD students need to work individually (see details on Canvas). If you are a PhD student or are otherwise working alone, please form a group by yourself. Please email Jacob if you have questions about this assignment or the final project.

Conceptual Questions (15 pts)

1. Consider the four main classification methods that have been presented thus far this semester: logistic regression, k-Nearest Neighbors, linear discriminant analysis (LDA), and quadratic discriminant analysis (QDA. Which of these methods may be appropriate if you know the decision boundary between the classes is linear? (3pts)

   Linear discriminant analysis (LDA) and logistic regression will be appropriate if you know the decision boundary between the classes is linear. Both LDA and logistic regression assume that the decision boundary is linear.

2. Suppose you had the following data and you are using KNN Regression with Euclidean distance. Consider the prediction problem where you want to predict Y for the data point X1 = X2 = X3 = 0.

X1	X2	X3	Y
0	3.5	2	2
1	2.1	3	1
2	4.7	1	3
1	3.9	1	2
0	2.9	2	4
1	1.5	2	1
1	3.5	4	2

(a) Compute the Euclidean distance between each observation and the test point, X1 = X2 = X3 = 0. (1pt)

```r
# code
x1 <- c(0, 1, 2, 1, 0, 1, 1)
x2 <- c(3.5, 2.1, 4.7, 3.9, 2.9, 1.5, 3.5)
x3 <- c(2, 3, 1, 1, 2, 2, 4)
xt <- c(0, 0, 0)

distances <- sqrt((x1 - xt[1])^2 + (x2 - xt[2])^2 + (x3 - xt[3])^2)

for (i in 1:length(distances)) {
  print(paste("Distance between observation", i, "and test point is", distances[i]))
}
```

```
## [1] "Distance between observation 1 and test point is 4.03112887414927"
## [1] "Distance between observation 2 and test point is 3.79605057922046"
## [1] "Distance between observation 3 and test point is 5.20480547186924"
## [1] "Distance between observation 4 and test point is 4.14849370253831"
## [1] "Distance between observation 5 and test point is 3.52278299076171"
## [1] "Distance between observation 6 and test point is 2.69258240356725"
## [1] "Distance between observation 7 and test point is 5.40832691319598"
```

(b) What is your prediction with K = 2? Why? (1pt)

K is the number of nearest neighbors to consider. In this case, K = 2. The two nearest neighbors to the test point are the 5th (distance: 3.6) and 6th (distance: 2.7) observations. The Y values for these observations are 4 and 1, respectively. The prediction is the average of these two values, which is (4 + 1) / 2 = 2.5.

(c) If the Bayes decision boundary in this problem is highly nonlinear, then would we expect the best value for K to be large or small? Why? (1pt)

If the Bayes decision boundary is highly nonlinear, we would expect the best value for K to be small. A small value of K will allow the model to capture the nonlinearities in the data. A large value of K would smooth out the decision boundary and may not capture the nonlinearities in the data.

3. Consider we conduct a research study analyzing the risk factors for developing prostate cancer among men, with variables X_1 = age (years), X_2 = family history of prostate cancer (0 = no, 1 = yes), X_3 = smoking status (0 = non-smoker, 1 = smoker), and Y = probability of developing prostate cancer. A logistic regression analysis is performed, resulting in estimated coefficients $\hat{\beta}_1 = 0.06$, $\hat{\beta}_2 = 1.2$, $\hat{\beta}_3 = 0.8$, and $\hat{\beta}_0 = -3.5$.

(a) Interpret $\hat{\beta}_2$. (1 pt)

In logistic regression, the coefficients represent the change in the log-odds of the outcome which is developing prostate cancer. $\hat{\beta}_2$ is the connected to whether the person has a family history of prostate cancer. It is associated with an increase in the log-odds of developing prostate cancer by 1.2, when the person who has a family history of prostate cancer is compared to someone without such family history, while holding age and smoking status constant.

(b) Estimate the probability that a 60-year-old man with a family history of prostate cancer who is a smoker develops prostate cancer. (2 pts)

```
# code
B0 <- -3.5
B1 <- 0.06
B2 <- 1.2
B3 <- 0.8

X1 <- 60
X2 <- 1
X3 <- 1

log_odds <- B0 + B1 * X1 + B2 * X2 + B3 * X3

probability <- exp(log_odds) / (1 + exp(log_odds))

probability
```

```
## [1] 0.8909032
```

4. k-fold cross-validation

(a) Briefly explain how k-fold cross-validation is implemented. (2pts)

In k-fold cross-validation, the data is divided into k subsets of equal size. The model is trained on k-1 of the subsets and tested on the remaining subset. This process is repeated k times, with each of the k subsets used exactly once as the validation data. The k results from the folds are then averaged to produce a single estimation of model performance.

(b) What are the advantages of k-fold cross-validation relative to the validation set approach? (1pt)

The advantages of k-fold cross-validation relative to the validation set approach are that it provides a more accurate estimate of model performance. It uses all the data for training and testing, which can help reduce the variance of the performance estimate. It also allows for the validation of the model on multiple subsets of the data, which can help identify potential issues with overfitting.

5. Suppose you want to minimize the false negative rate in your classification. You run two models: A and B. AUC for Model A is 0.7 and for Model B is 0.8. Can you conclude that you should choose Model B? Why or why not? (3 pts)

The AUC is a measure of the model's ability to discriminate between the positive and negative classes. A higher AUC indicates better discrimination. However, the AUC does not directly measure the false negative rate. It is possible that Model B has a higher AUC but a higher false negative rate than Model A. Therefore, we cannot conclude that we should choose Model B based solely on the AUC values. We would need to evaluate the false negative rates of both models to determine which one minimizes the false negative rate.

Applied Questions (15 pts)

Predicting Hospital Length of Stay

For the next portion of this assignment you will be working with the `LengthOfStay.csv` dataset. This dataset has data points on patients admitted into hospital, indicators of their health condition and how long they were admitted in the hospital.

This is an important problem in healthcare. In order for hospitals to optimize resource allocation, it is important to predict accurately how long a newly admitted patient will stay in the hospital.

1. What are the dimensions of the dataset? (1 pt)

```r
# code
df <- read.csv("LengthOfStay.csv")
dim(df)
```

```
## [1] 3000   22
```

2. Use the `cor()` function to display the correlations of all **continuous** variables in the dataset. Which variables is most highly correlated with `lengthofstay`? (2 pts)

```r
# ignore X column from df

df <- df[, -1]

# code
continuous <- df[, sapply(df, is.numeric)]

correlations <- cor(continuous)

print(correlations)
```

```
##                            dialysisrenalendstage        asthma       irondef
## dialysisrenalendstage                 1.00000000 -0.0247830721  0.1078802732
## asthma                               -0.02478307  1.0000000000  0.0413275215
## irondef                               0.10788027  0.0413275215  1.0000000000
## pneum                                 0.11049415  0.0941012490  0.1606937669
## substancedependence                   0.02595747  0.0006701374  0.0884918877
## psychologicaldisordermajor            0.08285511 -0.0028660246  0.1450150239
## depress                              -0.01454039  0.0289611803  0.0210605835
## psychother                            0.35205041 -0.0233323053  0.1713331609
## fibrosisandother                     -0.01493529  0.0125407055  0.0356952788
## malnutrition                          0.15974813 -0.0102662922  0.2458381917
## hemo                                  0.14490098  0.0425159554  0.1657287397
## hematocrit                           -0.11517983 -0.0509783364 -0.0360440868
## neutrophils                          -0.08579520  0.0372100959 -0.0621343153
## sodium                               -0.02919437  0.0059390117 -0.0251371513
## glucose                              -0.01341865  0.0099580292  0.0003679501
## bloodureanitro                        0.34475447 -0.0316052377  0.1873691086
## creatinine                           -0.02401302 -0.0034094745 -0.0373387935
## bmi                                   0.02732097 -0.0000647703 -0.0146429085
## pulse                                -0.02557007  0.0104235628  0.0023455949
## respiration                          -0.02661488 -0.0257891585 -0.0494877118
## lengthofstay                          0.15658314  0.0839395080  0.1977895853
##                                   pneum substancedependence
## dialysisrenalendstage       0.110494146        0.0259574654
## asthma                      0.094101249        0.0006701374
## irondef                     0.160693767        0.0884918877
## pneum                       1.000000000        0.1008666669
## substancedependence         0.100866667        1.0000000000
## psychologicaldisordermajor  0.130839877        0.1259672427
## depress                     0.015570919        0.0300840822
## psychother                  0.082569349        0.0556527362
## fibrosisandother            0.048462376        0.0341346649
## malnutrition                0.156353636        0.0340796098
## hemo                        0.077943324        0.0633802547
## hematocrit                 -0.078898808       -0.0905505256
## neutrophils                -0.015999500       -0.0471491311
## sodium                     -0.001148303        0.0447314453
## glucose                     0.011050755       -0.0016741568
## bloodureanitro              0.058289288        0.0664002952
## creatinine                  0.007770806       -0.0044820762
## bmi                         0.017863741        0.0211109665
## pulse                      -0.004811295       -0.0096294657
## respiration                -0.067998590       -0.0846390116
## lengthofstay                0.147786137        0.1518672513
##                            psychologicaldisordermajor       depress
## dialysisrenalendstage                     0.082855114 -0.0145403854
## asthma                                   -0.002866025  0.0289611803
## irondef                                   0.145015024  0.0210605835
## pneum                                     0.130839877  0.0155709192
## substancedependence                       0.125967243  0.0300840822
## psychologicaldisordermajor                1.000000000  0.3260519058
## depress                                   0.326051906  1.0000000000
## psychother                                0.080579177 -0.0198847877
## fibrosisandother                          0.020444407  0.0024408146
## malnutrition                              0.094118523  0.0074457584
## hemo                                      0.073734651  0.0354861192
## hematocrit                                0.041506590  0.0249061950
## neutrophils                              -0.147005315 -0.0614849793
## sodium                                    0.016237171 -0.0065790364
## glucose                                   0.022772903  0.0455513322
## bloodureanitro                            0.107972884  0.0273809105
## creatinine                               -0.017973734 -0.0005915899
## bmi                                       0.007244589 -0.0079017862
## pulse                                     0.001968803  0.0098134940
## respiration                               0.043409217  0.0452117632
## lengthofstay                              0.279332638  0.1121589399
##                               psychother fibrosisandother  malnutrition
## dialysisrenalendstage       0.3520504137     -0.014935290  0.1597481323
## asthma                     -0.0233323053      0.012540706 -0.0102662922
## irondef                     0.1713331609      0.035695279  0.2458381917
## pneum                       0.0825693486      0.048462376  0.1563536359
## substancedependence         0.0556527362      0.034134665  0.0340796098
## psychologicaldisordermajor  0.0805791774      0.020444407  0.0941185227
## depress                    -0.0198847877      0.002440815  0.0074457584
## psychother                  1.0000000000      0.002684617  0.3151236504
## fibrosisandother            0.0026846171      1.000000000  0.0907181917
## malnutrition                0.3151236504      0.090718192  1.0000000000
## hemo                        0.1715068992      0.087473017  0.1164783880
## hematocrit                 -0.1332832088     -0.076887536 -0.0753788894
## neutrophils                -0.0919607083      0.074354035 -0.0200185919
## sodium                     -0.0161713443     -0.003289905 -0.0200846295
## glucose                    -0.0116576831     -0.027858219  0.0186660352
## bloodureanitro              0.4375847878      0.038979770  0.2930341624
## creatinine                 -0.0523413458     -0.009339381 -0.0142439554
## bmi                         0.0026609982      0.003129734 -0.0046175180
## pulse                      -0.0129961330      0.026331153  0.0001302477
## respiration                 0.0002497756     -0.050201442 -0.0275510008
## lengthofstay                0.1766111741      0.070724056  0.1532579669
##                                    hemo   hematocrit  neutrophils       sodium
## dialysisrenalendstage       0.144900984 -0.115179830 -0.085795202 -0.029194375
## asthma                      0.042515955 -0.050978336  0.037210096  0.005939012
## irondef                     0.165728740 -0.036044087 -0.062134315 -0.025137151
## pneum                       0.077943324 -0.078898808 -0.015999500 -0.001148303
## substancedependence         0.063380255 -0.090550526 -0.047149131  0.044731445
## psychologicaldisordermajor  0.073734651  0.041506590 -0.147005315  0.016237171
## depress                     0.035486119  0.024906195 -0.061484979 -0.006579036
## psychother                  0.171506899 -0.133283209 -0.091960708 -0.016171344
## fibrosisandother            0.087473017 -0.076887536  0.074354035 -0.003289905
## malnutrition                0.116478388 -0.075378889 -0.020018592 -0.020084630
## hemo                        1.000000000 -0.316585218 -0.073478094 -0.017539773
## hematocrit                 -0.316585218  1.000000000  0.131460567  0.024343426
## neutrophils                -0.073478094  0.131460567  1.000000000  0.008076991
## sodium                     -0.017539773  0.024343426  0.008076991  1.000000000
## glucose                    -0.005117843  0.001657728  0.008061932 -0.013221399
## bloodureanitro              0.124077436 -0.097450193 -0.059656588 -0.024281498
## creatinine                  0.023274474  0.002356703 -0.014551173  0.050811659
## bmi                        -0.015415262  0.011712518  0.010711979  0.004984098
## pulse                       0.009573936  0.017631611 -0.021684801  0.006491825
## respiration                -0.073624086  0.220649388 -0.062965424 -0.002377212
## lengthofstay                0.234001005 -0.078573851 -0.034198747 -0.004106144
##                                  glucose bloodureanitro    creatinine
## dialysisrenalendstage      -0.0134186452     0.34475447 -0.0240130150
## asthma                      0.0099580292    -0.03160524 -0.0034094745
## irondef                     0.0003679501     0.18736911 -0.0373387935
## pneum                       0.0110507549     0.05828929  0.0077708063
## substancedependence        -0.0016741568     0.06640030 -0.0044820762
## psychologicaldisordermajor  0.0227729027     0.10797288 -0.0179737338
## depress                     0.0455513322     0.02738091 -0.0005915899
## psychother                 -0.0116576831     0.43758479 -0.0523413458
## fibrosisandother           -0.0278582191     0.03897977 -0.0093393806
## malnutrition                0.0186660352     0.29303416 -0.0142439554
## hemo                       -0.0051178426     0.12407744  0.0232744745
## hematocrit                  0.0016577282    -0.09745019  0.0023567029
## neutrophils                 0.0080619318    -0.05965659 -0.0145511726
## sodium                     -0.0132213991    -0.02428150  0.0508116591
## glucose                     1.0000000000     0.01021092  0.0238207285
## bloodureanitro              0.0102109173     1.00000000 -0.0172069827
## creatinine                  0.0238207285    -0.01720698  1.0000000000
## bmi                          0.0001641313     0.01175216 -0.0054429395
## pulse                       -0.0184095819    -0.01563025  0.0120159066
## respiration                  0.0069222501     0.02566680  0.0064235485
## lengthofstay                -0.0203796923     0.19183273 -0.0041122730
##                                       bmi         pulse   respiration
## dialysisrenalendstage        0.0273209692 -0.0255700743 -0.0266148816
## asthma                      -0.0000647703  0.0104235628 -0.0257891585
## irondef                     -0.0146429085  0.0023455949 -0.0494877118
## pneum                        0.0178637405 -0.0048112946 -0.0679985898
## substancedependence          0.0211109665 -0.0096294657 -0.0846390116
## psychologicaldisordermajor  0.0072445886  0.0019688030  0.0434092170
## depress                     -0.0079017862  0.0098134940  0.0452117632
## psychother                   0.0026609982 -0.0129961330  0.0002497756
## fibrosisandother             0.0031297343  0.0263311529 -0.0502014422
## malnutrition                -0.0046175180  0.0001302477 -0.0275510008
## hemo                        -0.0154152621  0.0095739357 -0.0736240859
## hematocrit                   0.0117125183  0.0176316108  0.2206493884
## neutrophils                  0.0107119794 -0.0216848008 -0.0629654237
## sodium                       0.0049840979  0.0064918249 -0.0023772121
## glucose                      0.0001641313 -0.0184095819  0.0069222501
## bloodureanitro               0.0117521641 -0.0156302531  0.0256667971
## creatinine                  -0.0054429395  0.0120159066  0.0064235485
## bmi                          1.0000000000 -0.0120018215  0.0065377627
## pulse                       -0.0120018215  1.0000000000  0.0034574928
## respiration                  0.0065377627  0.0034574928  1.0000000000
## lengthofstay                -0.0023549638 -0.0125859402 -0.0356967405
##                             lengthofstay
## dialysisrenalendstage        0.156583136
## asthma                       0.083939508
## irondef                      0.197789585
## pneum                        0.147786137
## substancedependence          0.151867251
## psychologicaldisordermajor  0.279332638
## depress                      0.112158940
## psychother                   0.176611174
## fibrosisandother             0.070724056
## malnutrition                 0.153257967
## hemo                         0.234001005
## hematocrit                  -0.078573851
## neutrophils                 -0.034198747
## sodium                      -0.004106144
## glucose                     -0.020379692
## bloodureanitro               0.191832733
## creatinine                  -0.004112273
## bmi                         -0.002354964
## pulse                       -0.012585940
## respiration                 -0.035696740
## lengthofstay                 1.000000000
```

```
# most highly correlated variable with lengthofstay
cor_with_lengthofstay <- correlations["lengthofstay", ]
cor_with_lengthofstay <- cor_with_lengthofstay[order(abs(cor_with_lengthofstay), decreasing = TRUE)]

#
```

```r
print(names(cor_with_lengthofstay[2:length(cor_with_lengthofstay)]))
```

```
##  [1] "psychologicaldisordermajor" "hemo"
##  [3] "irondef"                    "bloodureanitro"
##  [5] "psychother"                 "dialysisrenalendstage"
##  [7] "malnutrition"               "substancedependence"
##  [9] "pneum"                      "depress"
## [11] "asthma"                     "hematocrit"
## [13] "fibrosisandother"           "respiration"
## [15] "neutrophils"                "glucose"
## [17] "pulse"                      "creatinine"
## [19] "sodium"                     "bmi"
```

The most highly correlated **continuous** variable with `lengthofstay` is `psychologicaldisordermajor`.

Consider the prediction problem where you want to predict the length of stay for a patient (`lengthofstay`) against all other variables available in the data set.

3. Run ridge regression with cross-validation and standardized features using the canned function `cv.glmnet` from the package `glmnet`. You can use the λ sequence generated by `cv.glment` (you do not need to provide your own λ sequence). In order to receive credit for this question, make the line immediately preceding this command say `set.seed(222)` and run the two lines together. Please report all numbers by rounding to three decimal places. (2 pts)

```r
# code

index_of_lengthofstay <- which(colnames(df) == "lengthofstay")

set.seed(222)
model <- cv.glmnet(x = as.matrix(df[, -index_of_lengthofstay]),
                   y = as.numeric(df[, index_of_lengthofstay]),
                   alpha = 0, standardize = TRUE)

model
```

```
##
## Call:  cv.glmnet(x = as.matrix(df[, -index_of_lengthofstay]), y = as.numeric(df[,      index_of_leng
##
## Measure: Mean-Squared Error
##
##     Lambda Index Measure     SE Nonzero
## min 0.1887    89   4.874 0.1479      20
## 1se 2.8019    60   5.015 0.1409      20
```

(a) Which λ had the lowest mean cross-validation error? (1 pt)

```r
# code
print(paste("The lambda value is", round(model$lambda.min, 3)))
```

```
## [1] "The lambda value is 0.189"
```

(b) What was the cross-validation error? (1 pt)

```r
# code
print(paste("The cross-validation error is",
            round(model$cvm[model$lambda == model$lambda.min], 3)))
```

```
## [1] "The cross-validation error is 4.874"
```

(c) What was the standard error of the mean cross-validation error for this value of λ? (1 pt)

```r
# code
print(paste("The standard error is",
            round(model$cvsd[model$lambda == model$lambda.min], 3)))
```

```
## [1] "The standard error is 0.148"
```

(d) What was the largest value of λ whose mean cross validation error was within one standard deviation of the lowest cross-validation error? (1 pt)

```r
# code
print(paste("The largest value of lambda is",
            round(model$lambda.1se, 3)))
```

```
## [1] "The largest value of lambda is 2.802"
```

4. Produce the regression coefficients for the ridge regression model with the λ value that minimizes the cross-validation error. Compare these coefficients with a standard linear regression model. (2 pts)

```r
# code

# Ridge regression coefficients
ridge_coefs <- coef(model, s = model$lambda.min)
# round by 3
ridge_coefs <- round(ridge_coefs, 3)
# print(ridge_coefs)

# standard linear regression model
lm_model <- lm(lengthofstay ~ ., data = df)
lm_coefs <- coef(lm_model)
lm_coefs <- round(lm_coefs, 3)
# print(lm_coefs)

for (i in 1:length(lm_coefs)) {
  print(paste("Coefficient for", names(lm_coefs)[i], ", ridge:", lm_coefs[i],
              "standard linear regression:", ridge_coefs[i]))
}
```

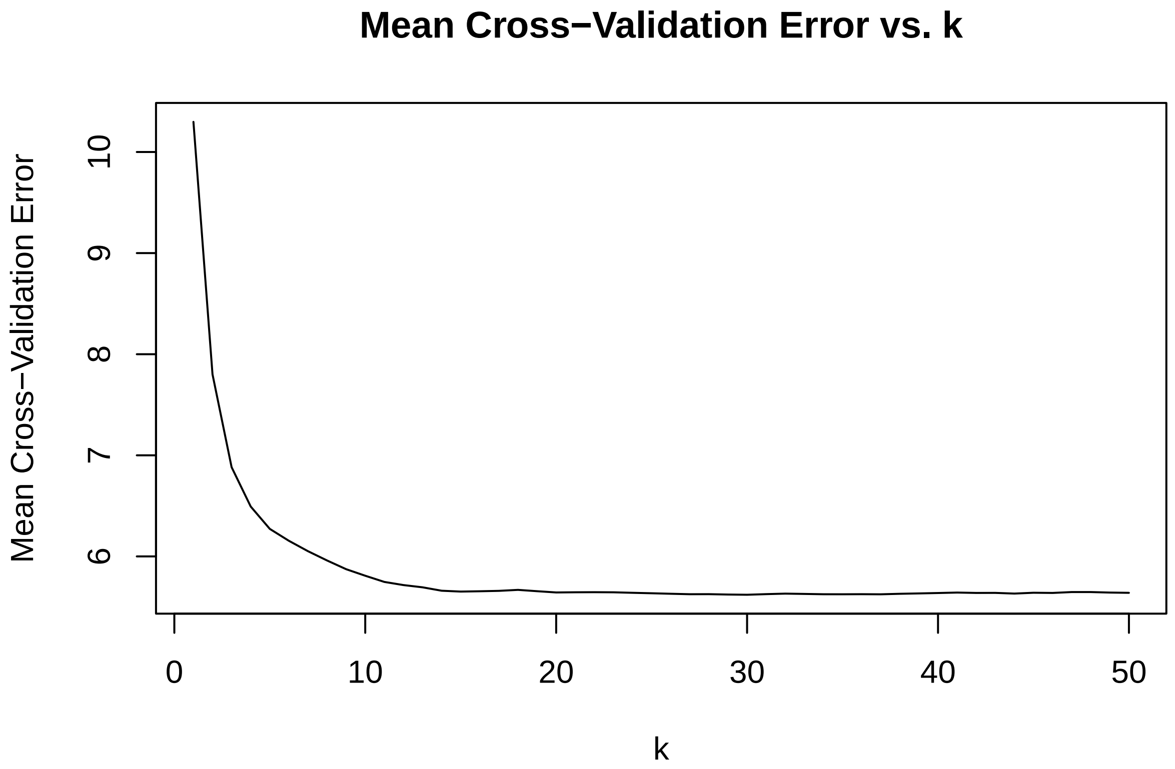
```
## [1] "Coefficient for (Intercept) , ridge: 4.245 standard linear regression: 4.234"
## [1] "Coefficient for dialysisrenalendstage , ridge: 0.687 standard linear regression: 0.674"
## [1] "Coefficient for asthma , ridge: 1.022 standard linear regression: 0.952"
## [1] "Coefficient for irondef , ridge: 0.683 standard linear regression: 0.668"
## [1] "Coefficient for pneum , ridge: 0.669 standard linear regression: 0.661"
## [1] "Coefficient for substancedependence , ridge: 0.889 standard linear regression: 0.847"
## [1] "Coefficient for psychologicaldisordermajor , ridge: 1.202 standard linear regression: 1.118"
## [1] "Coefficient for depress , ridge: 0.364 standard linear regression: 0.391"
## [1] "Coefficient for psychother , ridge: 0.56 standard linear regression: 0.56"
## [1] "Coefficient for fibrosisandother , ridge: 1.119 standard linear regression: 1.087"
## [1] "Coefficient for malnutrition , ridge: 0.398 standard linear regression: 0.413"
## [1] "Coefficient for hemo , ridge: 1.387 standard linear regression: 1.293"
## [1] "Coefficient for hematocrit , ridge: 0.009 standard linear regression: 0.004"
## [1] "Coefficient for neutrophils , ridge: 0.015 standard linear regression: 0.012"
## [1] "Coefficient for sodium , ridge: -0.002 standard linear regression: -0.002"
## [1] "Coefficient for glucose , ridge: -0.002 standard linear regression: -0.002"
## [1] "Coefficient for bloodureanitro , ridge: 0.017 standard linear regression: 0.016"
## [1] "Coefficient for creatinine , ridge: 0.084 standard linear regression: 0.074"
## [1] "Coefficient for bmi , ridge: -0.007 standard linear regression: -0.006"
## [1] "Coefficient for pulse , ridge: -0.003 standard linear regression: -0.002"
## [1] "Coefficient for respiration , ridge: -0.065 standard linear regression: -0.062"
```

5. Now consider the same prediction problem. Implement your own 5-fold cross-validation routine for KNN for $K = 1, ..., 50$ (write the cross-validation routine yourself rather than using a canned package). Include the snippet of code you wrote here. It should not exceed 20 lines. (4pts)

```r
# code

set.seed(222)

cross_validate_knn <- function(full_data, y_index, k_max=50, folds=5) {
  fold_ids <- rep(seq(folds), ceiling(nrow(full_data) / folds))
  fold_ids <- fold_ids[1:nrow(full_data)]
  fold_ids <- sample(fold_ids, length(fold_ids))
  cv_errors <- c()
  for (k in 1:k_max) {
    fold_errors <- c()
    for (fold in 1:folds) {
      train_data <- full_data[fold_ids != fold, ]
      test_data <- full_data[fold_ids == fold, ]
      knn_model <- knn.reg(train = train_data[, -y_index],
                           test = test_data[, -y_index],
                           y = train_data[, y_index], k = k)
      fold_errors <- c(fold_errors, mean((knn_model$pred - test_data[, y_index])^2))
    }
    cv_errors <- c(cv_errors, mean(fold_errors))
  }
  return (cv_errors)
}
```

(a) Plot of mean cross-validation MSE as a function of k.

```r
# code

cv_errors <- cross_validate_knn(df, index_of_lengthofstay)
# plot
plot(1:50, cv_errors, type = "l", xlab = "k", ylab = "Mean Cross-Validation Error", main = "Mean Cross-
```



(b) The best k according to CV is

```
# code
best_k <- which.min(cv_errors)
print(paste("The best k according to cross-validation is", best_k))
```

```
## [1] "The best k according to cross-validation is 30"
```

(c) The cross-validation error for the best k is

```
# code
print(paste("The cross-validation error for the best k is", round(cv_errors[best_k], 3)))
```

```
## [1] "The cross-validation error for the best k is 5.621"
```

Challenge Problem (just for fun 0 pts)

Tasks:

(a) Predict Length of Stay with LASSO Regression with Cross-Validation:

- Use the `cv.glmnet` function from the `glmnet` package to perform LASSO regression with cross-validation.
- Set `alpha = 1` to specify LASSO regression.

(b) Selecting the Best Model:

- Identify the value of lambda that minimizes the cross-validation error.
- Report this value of lambda and the corresponding cross-validation error (rounded to three decimal places).

(c) Visualization:

- Plot the cross-validation curve (cross-validation error versus log(lambda)) using `plot(cv.glmnet_object)`.
- Plot the coefficient paths as a function of log(lambda) to visualize how coefficients change with different lambda values.

(d) Interpretation:

- Discuss the variables that have been selected by the LASSO model.
- Compare these variables to those in the standard linear regression model you previously fitted.
- Comment on any differences and provide possible explanations for why certain variables were eliminated.

```
# code
```