Assignment #2

Janelle Morano

2/14/2022

This assignment will walk through some of the practicalities of applying the ridge and lasso regression methods. We will fit and test regular (OLS), ridge, and lasso regressions for average annual runoff vs. many other characteristics for 211 catchments across the Northeast US. A map of the watershed centroids is shown, along with a picture of a characteristic watershed.

a. Download data. These data are all from the Gages II database developed by the USGS. A description of the code for each covariate can be found in the excel file "gagesII_sept30_2011_var_desc". The dependent variable for prediction will be annual average runoff, and all other variables are independent covariates to be used to make the predictions.

b. Read in data.

```
Gages2 <- read.table("/Users/janellemorano/Git/Reference-R-scripts/Envtl-Multivariate-Stats/gages2.data.txt", sep
= " ", header = TRUE)
```

c. Create a function called scale2 that takes as input one argument (a vector x) and returns a standardized version of x, i.e., by subtracting off the mean and then dividing by the standard deviation. **Creating function, scale2**

```
scale2 <- function(x) {
  x <- (x- (mean(x))/sd(x))
  return(x)
}
```

d. Using the apply function and your scale2 function, standardize each variable in Gages2, and put these results into a new data frame called Gages2.scale. You can use the function data.frame() to turn the results of the apply function into a data frame. **Standardize each variable in the dataset and save to Gages2.scale**

```
Gages2.scale <- data.frame(apply(Gages2, MARGIN = 2, FUN=scale2)) #apply the function scale2 to each column
head(Gages2.scale)
```

```
##      runoff PPTAVG_BASIN DRAIN_SQKM.log BAS_COMPACTNESS T_AVG_BASIN      PET
## 1 599.1212     89.80182       4.6387559       -2.2244968   -0.868184 493.0266
## 2 591.0212     93.34182      -0.7785438       -0.1944968    0.531816 527.4266
## 3 755.0212    117.67182      -0.3023096       -0.9944968    2.031816 532.2266
## 4 753.9212    107.77182       1.2587728       -1.4344968    1.841816 542.5266
## 5 755.0212    114.84182       2.0064675       -2.0744968    2.031816 536.9266
## 6 754.9212    107.96182       2.4978460       -1.7344968    1.781816 538.2266
##    ARTIFPATH_PCT BFI_AVE     PERDUN      PERHOR     TOPWET  DEVNLCD06 FORESTNLCD06
## 1     15.6529509 46.5203 -0.624761  0.4257002  0.1068081  0.1366363    67.227198
## 2     -0.7470491 46.9203  1.075239  0.4257002 -0.6831919  6.8166363     8.257198
## 3      3.2929509 47.8203  3.075239 -1.3742998  2.1468081 -0.6133637    30.557198
## 4     11.0529509 49.1203  3.175239 -0.2742998 -0.1031919  0.6466363    69.487198
## 5      9.4529509 45.7203  3.175239 -0.4742998  0.6268081  0.1166363    49.377198
## 6     10.0729509 42.3203  0.475239  0.4257002  0.4368081  0.5766363    71.227198
##    PLANTNLCD06 AWCAVE.log PERMAVE.log      BDAVE OMAVE.log    WTDEPAVE ROCKDEPAVE
## 1   3.9719358   9.061186 -1.17297068 -18.40729 0.9131675 -1.06539945   41.54543
## 2  57.7419358   9.014666 -1.20055864 -18.35729 0.6540056  0.44460055   44.78543
## 3  37.9319358   8.167368  1.07796302 -18.43729 1.9997581 -0.02539945   53.19543
## 4  -0.2180642   8.678194  0.21332368 -18.29729 1.2775485 -0.77539945   45.28543
## 5  12.1819358   8.742733  0.49845147 -18.47729 2.2240467 -1.05539945   49.44543
## 6   0.7619358   8.609201  0.09426432 -18.23729 1.0136102 -0.80539945   48.52543
##      NO4AVE NO200AVE  NO10AVE  KFACT_UP     RFACT ELEV_MEAN_M_BASIN
## 1 70.06562 49.90357 63.40573 -4.963723 64.26061          271.17852
## 2 74.31562 50.20357 64.24573 -4.923723 62.75061          196.47852
## 3 62.39562 10.78357 59.11573 -5.013723 94.85061           62.47852
## 4 66.23562 33.05357 60.81573 -4.983723 84.66061           95.17852
## 5 67.58562 30.45357 62.83573 -4.993723 93.41061           88.27852
## 6 68.74562 31.89357 62.38573 -5.003723 86.62061          133.97852
##    ELEV_MAX_M_BASIN ELEV_MIN_M_BASIN ELEV_MEDIAN_M_BASIN ELEV_STD_M_BASIN
## 1         601.95649         155.41555           250.89211         80.213615
## 2         247.95649         173.41555           195.89211         10.613615
## 3          85.95649          40.41555            62.89211          6.713615
## 4         193.95649          49.41555            88.89211         22.313615
## 5         381.95649          39.41555            79.89211         37.913615
## 6         445.95649          46.41555           128.89211         44.013615
##    ELEV_SITE_M    RRMEAN RRMEDIAN.log SLOPE_PCT.log ASPECT_DEGREES
## 1   155.38693 -3.447607    1.1088262     -1.193158       127.74765
## 2   174.38693 -3.393607    1.4650359     -2.249210       132.84765
## 3    40.38693 -3.216607    1.9527963     -2.719214       101.44765
## 4    50.38693 -3.388607    1.3585891     -1.930757        65.94765
## 5    39.38693 -3.563607    0.5256799     -2.026067       113.54765
## 6    48.38693 -3.487607    1.0757263     -1.598623       184.14765
##   ASPECT_NORTHNESS ASPECT_EASTNESS
## 1       -0.06246996       0.6572659
## 2       -0.12946996       0.5962659
## 3        0.34253004       0.8622659
## 4        0.95053004       0.8192659
## 5        0.14453004       0.7912659
## 6       -0.41446996      -0.2207341
```

e. Using the data in Gages2.scale, fit a standard linear regression for annual average runoff vs. the other covariates for the 211 catchments. Do not include an intercept in this regression (since you already centered the runoff data around 0).

Fit a linear regression for annual average runoff vs. all other covariates and do not include the intercept because the data were standardized in the previous step.

The estimate of PPTAVG_BASIN was 0.24902, which is not what I have. Where's the problem?

```
# covar <- Gages2.scale[,2:37]
lm.Gages2.scale <- lm(runoff ~ . -1, data = Gages2.scale) #-1 or +0 removes intercept
summary(lm.Gages2.scale)
```

```
##
## Call:
## lm(formula = runoff ~ . - 1, data = Gages2.scale)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -247.379  -43.659    1.111   48.109  187.683
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## PPTAVG_BASIN            1.9353     1.0397   1.861   0.0644 .
## DRAIN_SQKM.log          5.7405     8.6215   0.666   0.5064
## BAS_COMPACTNESS       -16.6500    15.5194  -1.073   0.2848
## T_AVG_BASIN            -1.7823    32.0912  -0.056   0.9558
## PET                    -1.0774     1.2094  -0.891   0.3742
## ARTIFPATH_PCT          -1.4165     1.8987  -0.746   0.4566
## BFI_AVE                 2.0470     1.5839   1.292   0.1979
## PERDUN                  9.1186    10.8570   0.840   0.4021
## PERHOR                 -8.0990     6.4785  -1.250   0.2129
## TOPWET                 19.9173    15.0795   1.321   0.1883
## DEVNLCD06              -1.2163     2.4425  -0.498   0.6191
## FORESTNLCD06            1.2439     1.4949   0.832   0.4065
## PLANTNLCD06            -0.5904     1.5650  -0.377   0.7064
## AWCAVE.log            -33.1760    83.0109  -0.400   0.6899
## PERMAVE.log            -2.1688    26.3860  -0.082   0.9346
## BDAVE                  67.3826    98.2418   0.686   0.4937
## OMAVE.log               3.1326    15.5320   0.202   0.8404
## WTDEPAVE                4.8346    15.1394   0.319   0.7499
## ROCKDEPAVE              3.6135     1.8654   1.937   0.0543 .
## NO4AVE                 -0.8911     6.4468  -0.138   0.8902
## NO200AVE                1.5353     1.9803   0.775   0.4392
## NO10AVE                 5.9813     6.5581   0.912   0.3630
## KFACT_UP             -382.5415   264.3389  -1.447   0.1496
## RFACT                   0.6800     0.6290   1.081   0.2811
## ELEV_MEAN_M_BASIN       0.1349     0.9413   0.143   0.8862
## ELEV_MAX_M_BASIN       -0.1587     0.1095  -1.449   0.1492
## ELEV_MIN_M_BASIN       -0.9420     0.8009  -1.176   0.2411
## ELEV_MEDIAN_M_BASIN     0.1290     0.7802   0.165   0.8689
## ELEV_STD_M_BASIN        0.1230     0.4966   0.248   0.8048
## ELEV_SITE_M             0.7419     0.7311   1.015   0.3116
## RRMEAN                 14.0195   254.0235   0.055   0.9561
## RRMEDIAN.log          -42.3124    87.4062  -0.484   0.6289
## SLOPE_PCT.log          -1.6786    24.7020  -0.068   0.9459
## ASPECT_DEGREES          0.1215     0.1630   0.745   0.4570
## ASPECT_NORTHNESS      -15.5502    10.7522  -1.446   0.1499
## ASPECT_EASTNESS       -11.7269    19.2720  -0.608   0.5436
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 77.82 on 175 degrees of freedom
## Multiple R-squared:  0.9864, Adjusted R-squared:  0.9836
## F-statistic: 351.6 on 36 and 175 DF,  p-value: < 2.2e-16
```

Remember Betahat is your Estimate/StdError = tvalue pt(estimate/se, df =) then use the cdf pt(-estimate/se)*2 is the area under the curve

Which of the different predictors have a statistically significant relationship to flow based on a standard t-test framework? At what significance level?

Currently, none of the predictors have a statitically significant relationship to flow at <0.05, but PPTAVG_BASIN and ROCKDEPAVE do at <0.10.

How would you interpret the magnitude of the regression coefficients for those covariates, i.e., how would you articulate how much runoff changes per change in the covariates? Think about the standardization you did in step c for your answer here.

The estimate for PPTAVG_BASIN is 1.9353 and ROCKDEPAVE is 3.6135, so the runoff would increase by a factor of 1.9353 and 3.6135 from the average runoff for every incremental increase of PPTAVG_BASIN and ROCKDEPAVE, respectively.

f. Create a vector of runoff predictions and calculate the root mean squared error (RMSE) of these predictions.

$$RMSE = \sqrt{\frac{\Sigma(Pred_i - Obs_i)^2}{n}}$$

```
#Create a vector of runoff predictions based on the model. These are predictions
pred.runoff <- predict(lm.Gages2.scale)
#Calculate the root mean squared error (RMSE) of the predictions.
sqrt(mean((Gages2.scale$runoff - pred.runoff)^2))
```

```
## [1] 70.87552
```

```
# RMSE(pred.runoff, lm.Gages2.scale$PPTAVG_BASIN)
```

Plot your predicted values against the observed values. Be sure to include a 1:1 line and to label your axes appropriately (what exactly is being plotted on each axis?).

```
# plot()
# pred.runoff <- predict(lm.Gages2.scale)
# #Calculate the root mean squared error (RMSE) of the predictions.
# sqrt(mean((Gages2.scale$runoff - pred.runoff)^2))
# # RMSE(pred.runoff, lm.Gages2.scale$PPTAVG_BASIN)
```