ADA 442 HomeWork

Homework 3: Ridge vs LASSO Regression

Furkan

11/23/2021

Contents

1 Introduction

- The ultimate purpose of this research is to compare which Regression Model is better for prediction on 'College' Data set.

2 Methodology

Skeleton Formula that has been used for Simple Linear Regression : Y ~ B0 + B1x + epsilon

Loss function Formula that has been used for Ridge Regression : OLS + alpha * summation (squared coefficient values)

Loss function Formula that has been used for LASSO Regression : OLS + alpha * summation (absolute values of the magnitude of the coefficients)

- RMSE and R^2 comparison has been performed over different Regression Models by using formulas given above.

3 Data Set

```
library(ISLR2)
set.seed(73745) # for reproducible results
data(College)
```

- I have used U.S. News and World Report's College Data which contains 777 observations from 'ISLR2' library.
- College data frame has consisted of 18 variables.

4 Explaratory Data analysis

Brief information about the data set can be seen below

```
head(College)
```

```
##                              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
## Albertson College                Yes  587    479    158        38        62
##                              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
## Albertson College                    678          41    13500       3335   500
##                              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
## Albertson College                 675  67       73       9.4          11   9727
##                              Grad.Rate
## Abilene Christian University        60
## Adelphi University                  56
## Adrian College                      54
## Agnes Scott College                 59
## Alaska Pacific University           15
## Albertson College                   55
```

```
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
```

5 Model Fit

- Data set has been divided into two part as 80% and 20%.

```r
index = sample(1:nrow(College), 0.8*nrow(College))  # %80 for training, %20 for testing
train = College[index,] # Create the training data
test = College[-index,] # Create the test data
dim(train)
```

```
## [1] 621  18
```

```r
dim(test)
```

```
## [1] 156  18
```

- Superficial inference and relation between 'Accept' and 'Apps' can be seen as correlation matrix. *Correlation Matrix :*

```
cor(College[, c("Accept", "Apps")])
```

```
##           Accept      Apps
## Accept 1.0000000 0.9434506
## Apps   0.9434506 1.0000000
```

5.1 *Applying Linear Regression*

```
lm.fit = lm(Accept ~ Apps + Enroll + Outstate, data = train)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = Accept ~ Apps + Enroll + Outstate, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4966.8   -80.6    20.7   151.3  3993.7
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -18.306447  80.653932  -0.227    0.821
## Apps          0.383635   0.012887  29.770   <2e-16 ***
## Enroll        1.081982   0.055274  19.575   <2e-16 ***
## Outstate      0.005047   0.006845   0.737    0.461
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 636.6 on 617 degrees of freedom
## Multiple R-squared:  0.9388, Adjusted R-squared:  0.9386
## F-statistic:  3158 on 3 and 617 DF,  p-value: < 2.2e-16
```

- Since, if the predictors are meaningful, p-values of them must be smaller than 0.05. 'Apps' variable can be predictor.

Regression Linear Equation : Accept = 199.19774 + 0.61091*Apps *For every increase in Accept, the model predicts a increase of 0.61091 in Apps.*

- The most significant values to identify if a model well-explained are R^2 and adjusted R^2 values. R^2 values should be near 1.

5.1.1 Evaluate Linear Regression Model Performance

```
predictedAccept <- predict(lm.fit, test)
par(mfrow = c(1,2))


actuals_preds <- data.frame(cbind(actuals=test$Accept, predicteds=predictedAccept))  # make actuals_pre
correlation_accuracy <- cor(actuals_preds)
head(actuals_preds)
```

```
##                                    actuals predicteds
## Alaska Pacific University              146   153.4011
## Albright College                       839   704.2278
## Allegheny College                     1900  1608.9789
## American International College        1093   808.4018
## Appalachian State University          4664  4888.1524
## Assumption College                    1700  1392.5738
```

5.2 *Applying Rigde Regression*

- Loss function = OLS + alpha * summation (squared coefficient values)

```
library(glmnet)
```

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

```
## Loaded glmnet 4.1-3
```

```
x = train
y_train = train$Accept

x_test = data.matrix(test)
y_test = test$Accept

lambdas <- 10^seq(2, -3, by = -.1)
ridge_reg = glmnet(x, y_train, nlambda = 25, alpha = 0, family = 'gaussian', lambda = lambdas)

summary(ridge_reg)
```

```
##           Length Class     Mode
## a0         51    -none-    numeric
## beta      918    dgCMatrix S4
## df         51    -none-    numeric
## dim         2    -none-    numeric
## lambda     51    -none-    numeric
## dev.ratio  51    -none-    numeric
## nulldev     1    -none-    numeric
## npasses     1    -none-    numeric
## jerr        1    -none-    numeric
## offset      1    -none-    logical
## call        7    -none-    call
## nobs        1    -none-    numeric
```

- Ridge Regression model been run several times for different values of lambda.
- When alpha=0, Ridge Model is fit and if alpha=1, a lasso model is fit.

```
x_mtrx = data.matrix(train)
cv_ridge <- cv.glmnet(x_mtrx, y_train, alpha = 0, lambda = lambdas)
optimal_lambda <- cv_ridge$lambda.min
optimal_lambda
```

```
## [1] 0.001
```

5.2.1 Evaluate Ridge Model Performance

```
eval_results <- function(true, predicted, df) {
  # Sum of Squared Error
  SSE <- sum((predicted - true)^2)
  # Sum of Squared Total
  SST <- sum((true - mean(true))^2)
  R_square <- 1 - SSE / SST
  RMSE = sqrt(SSE/nrow(df))


  # Model performance metrics
data.frame(
  RMSE = RMSE,
  Rsquare = R_square
)

}

# Prediction and evaluation on train data
predictions_train <- predict(ridge_reg, s = optimal_lambda, newx = x_mtrx)

# Prediction and evaluation on test data
predictions_test <- predict(ridge_reg, s = optimal_lambda, newx = x_test)

eval_results(y_train, predictions_train, train)
```

```
##        RMSE Rsquare
## 1 0.3927581       1
```

```
eval_results(y_test, predictions_test, test)
```

```
##       RMSE Rsquare
## 1 0.347633       1
```

- There is an improvement in the performance compared with linear regression model.

5.3 *Applying Lasso Regression*

- Loss function = OLS + alpha * summation (absolute values of the magnitude of the coefficients)
- When alpha=0, Ridge Model is fit and if alpha=1, a lasso model is fit.

```
lambdas <- 10^seq(2, -3, by = -.1)

# Setting alpha = 1 implements lasso regression
lasso_reg <- cv.glmnet(x_mtrx, y_train, alpha = 1, lambda = lambdas, standardize = TRUE, nfolds = 5)

# Best
lambda_best <- lasso_reg$lambda.min
lambda_best
```

```
## [1] 0.001
```

5.3.1 Evaluate LASSO Model Performance

```
lasso_model <- glmnet(x_mtrx, y_train, alpha = 1, lambda = lambda_best, standardize = TRUE)

predictions_train <- predict(lasso_model, s = lambda_best, newx = x_mtrx)
eval_results(y_train, predictions_train, train)
```

```
##       RMSE   Rsquare
## 1 3.289273 0.9999984
```

```
predictions_test <- predict(lasso_model, s = lambda_best, newx = x_test)
eval_results(y_test, predictions_test, test)
```

```
##       RMSE   Rsquare
## 1 2.917397 0.9999976
```

6 Conclusions

- Ridge regression shrinks the regression coefficients, so that variables, with minor contribution to the outcome, have their coefficients close to zero.
- Lasso stands for Least Absolute Shrinkage and Selection Operator. It shrinks the regression coefficients toward zero by penalizing the regression model with a penalty term called L1-norm, which is the sum of the absolute coefficients.
- We know that the most ideal result would be an RMSE value of zero and R-squared value of 1.
- To conclude that, R^2 of Linear Regression Model : 0.9315

  RMSE of Ridge Model : 0.3763321 R^2 of Ridge Model : 1

  RMSE of LASSO Model : 4.091901 R^2 of LASSO Model : 0.9999981

  So, The first salient value is R^2 of Linear Regression Model because it is lowest and worst value in respect to accuracy. Also, R^2 value of Ridge Model which is 1 has the best prediction accuracy.

7 References

- Lecture Slides
- https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62
- https://www.pluralsight.com/guides/linear-lasso-and-ridge-regression-with-r
- http://www.sthda.com/english/articles/37-model-selection-essentials-in-r/153-penalized-regression-essentials-ridge-lasso-elastic-net/