# Regularization in Linear Regression

Applied Machine Learning for Educational Data Science

true

10/20/2021

## Contents



[Updated: Wed, Oct 20, 2021 - 14:31:20 ]

## Regularization

Regularization is a general strategy to incorporate additional penalty terms into the model fitting process and used not just for regression but a variety of other types of models. The idea behind the regularization is to constrain the size of regression coefficients with the purpose of reducing their sampling variation and, hence, reducing the variance of model predictions. These constrains are typically incorporated into the loss function to be optimized. There are two commonly used regularization strategy: **ridge penalty** and **lasso penalty**. In addition, there is also **elastic net**, a mixture of these two strategies.

### Ridge Penalty

Remember that we formulated the loss function for the linear regression as the sum of squared residuals across all observations. For ridge regression, we add a penalty term to this loss function and this penalty term is a function of all the regression coefficients in the model. Assuming that there are P regression coefficients in the model, the penalty term for the ridge regression would be

$$\lambda \sum_{i=1}^{P} \beta_p^2,$$

where $\lambda$ is a parameter that penalizes the regression coefficients when they get larger. Therefore, when we fit a regression model with ridge penalty, the loss function to minimize becomes

$$Loss = \sum_{i=1}^{N} \epsilon_{(i)}^2 + \lambda \sum_{i=1}^{P} \beta_p^2,$$

$$Loss = SSR + \lambda \sum_{i=1}^{P} \beta_p^2.$$

Let's consider the same example from the previous class. Suppose we fit a simple linear regression model such that the readability score is the outcome ($Y$) and average word length is the predictor($X$). Our regression model is

$$Y = \beta_0 + \beta_1 X + \epsilon,$$

and let's assume the set of coefficients are $\{\beta_0, \beta_1\} = \{7.5,-2\}$, so my model is

$$Y = 7.5 - 2X + \epsilon.$$

Then, the value of the loss function when $\lambda = 0.2$ would be equal to 27.433.

```
readability_sub <- read.csv('https://raw.githubusercontent.com/uo-datasci-specialization/c4-ml-fall-202

d <-  readability_sub[,c('mean.wl','target')]

b0 = 7.5
b1 = -2

d$predicted <- b0 + b1*d$mean.wl
d$error     <- d$target - d$predicted

d
```

```
    mean.wl      target  predicted       error
1  4.603659 -2.58590836 -1.7073171 -0.87859129
2  3.830688  0.45993224 -0.1613757  0.62130790
3  4.180851 -1.07470758 -0.8617021 -0.21300545
4  4.015544 -1.81700402 -0.5310881 -1.28591594
5  4.686047 -1.81491744 -1.8720930  0.05717559
6  4.211340 -0.94968236 -0.9226804 -0.02700194
7  4.025000 -0.12103065 -0.5500000  0.42896935
8  4.443182 -2.82200582 -1.3863636 -1.43564218
9  4.089385 -0.74845172 -0.6787709 -0.06968077
10 4.156757  0.73948755 -0.8135135  1.55300107
11 4.463277 -0.96218937 -1.4265537  0.46436430
12 5.478261 -2.21514888 -3.4565217  1.24137286
13 4.770492 -1.21845136 -2.0409836  0.82253224
14 4.568966 -1.89544351 -1.6379310 -0.25751247
15 4.735751 -0.04101056 -1.9715026  1.93049203
16 4.372340 -1.83716516 -1.2446809 -0.59248431
17 4.103448 -0.18818586 -0.7068966  0.51871069
18 4.042857 -0.81739314 -0.5857143 -0.23167886
19 4.202703 -1.86307557 -0.9054054 -0.95767016
20 3.853535 -0.41630158 -0.2070707 -0.20923088
```

```
lambda = 0.2

loss <- sum((d$error)^2) + lambda*(b0^2 + b1^2)

loss
```

```
[1] 27.43364
```

Now, as we did in the previous lecture, imagine that we computed the loss function with the ridge penalty term for every possible combination of the intercept ($\beta_0$) and the slope ($\beta_1$). Let's say the plausible range for the intercept is from -10 to 10 and the plausible range for the slope is from -2 to 2.

and I will consider every single possible value from -2 to 2 with increments of .01. Given that every single combination of $\beta_0$ and $\beta_1$ indicates a different model, these settings suggest a total of 80,601 models to explore.

Notice that when $\lambda$ is equal to 0, the loss function is identical to SSR; therefore, it becomes a linear regression with no regularization.

## Lasso Penalty

## Elastic Net