Selección de Modelos y Regularización

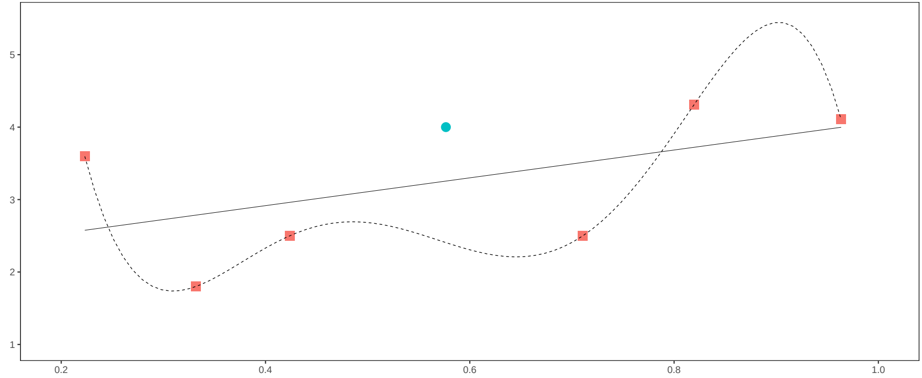
Ciencia de Datos para la toma de decisiones en Economía

Ignacio Sarmiento-Barbieri

Universidad de los Andes

Agenda

Overfit y Predicción fuera de Muestra

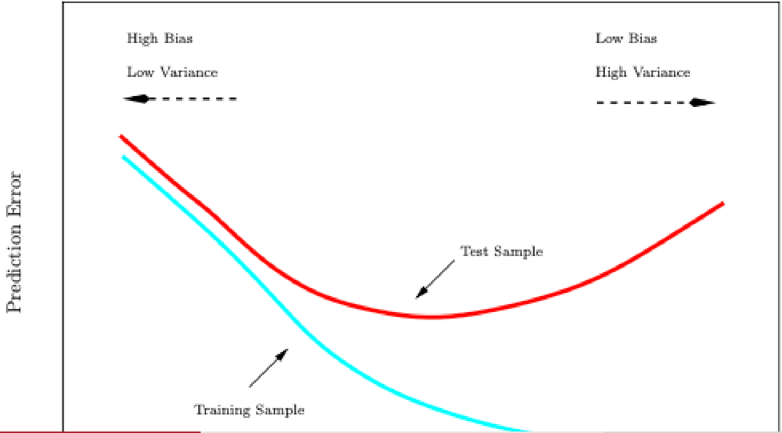


Overfit y Predicción fuera de Muestra

- ML nos interesa la predicción fuera de muestra

Overfit y Predicción fuera de Muestra

- ML nos interesa la predicción fuera de muestra
- Overfit: modelos complejos predicen muy bien dentro de muestra, pero tienden a hacer un mal trabajo fuera de muestra



Overfit y Predicción fuera de Muestra

- Hay que elegir el modelo que "mejor" prediga
  - AIC, BIC, C_p and Adjusted R^2
  - Métodos de Remuestreo
    - Enfoque del conjunto de validación
    - Loocv
    - Validación cruzada en K-partes (5 o 10)

Selección de Modelos: Motivación

```
model1<-lm(price~1,data=train)
test$model1<-predict(model1,newdata = test)
with(test,mean((price-model1)^2))
```

```
## [1] 22811540844
```

```
model2<-lm(price~bedrooms,data=train)
test$model2<-predict(model2,newdata = test)
with(test,mean((price-model2)^2))
```

```
## [1] 22490147170
```

```
model3<-lm(price~bedrooms+bathrooms+centair+fireplace+brick,data=train)
test$model3<-predict(model3,newdata = test)
with(test,mean((price-model3)^2))
```

```
## [1] 21982836467
```

Regularización

Lasso

- Para un $\lambda \geq 0$ dado, consideremos el siguiente problema de optimización

$$min_{\beta} E(\beta) = \sum_{i=1}^{n} (y_i - \beta_0 - x_{i1}\beta_1 - \cdots - x_{ip}\beta_p)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{1}$$

Lasso

- Para un $\lambda \geq 0$ dado, consideremos el siguiente problema de optimización

$$min_{\beta} E(\beta) = \sum_{i=1}^{n} (y_i - \beta_0 - x_{i1}\beta_1 - \cdots - x_{ip}\beta_p)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{1}$$

  - "LASSO's free lunch": selecciona automáticamente los predictores que van en el modelo ($\beta_j \neq 0$) y los que no ($\beta_j = 0$)
  - Porque? Los coeficientes que no van son soluciones de esquina
  - $L(\beta)$ es no differentiable

Lasso Intuición en 1 Dimension

- Lasso Intuición

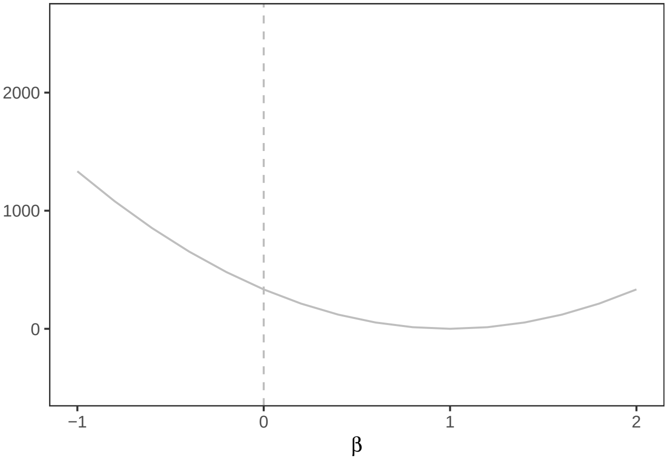
$$min_\beta E(\beta) = \sum_{i=1}^{n} (y_i - x_i\beta)^2 + \lambda|\beta| \tag{2}$$

- Un solo predictor, un solo coeficiente
- Si $\lambda = 0$

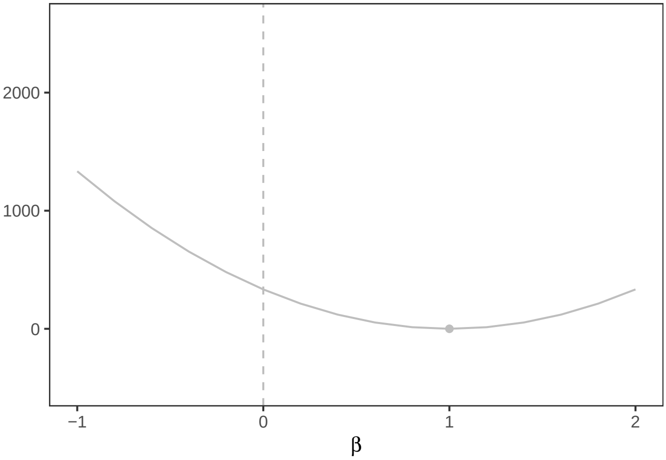
$$min_\beta E(\beta) = \sum_{i=1}^{n} (y_i - x_i\beta)^2 \tag{3}$$

- la solución es?

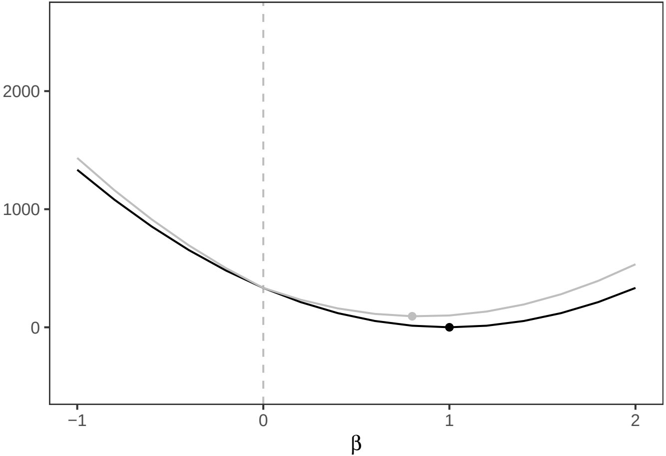
Intuición en 1 Dimension



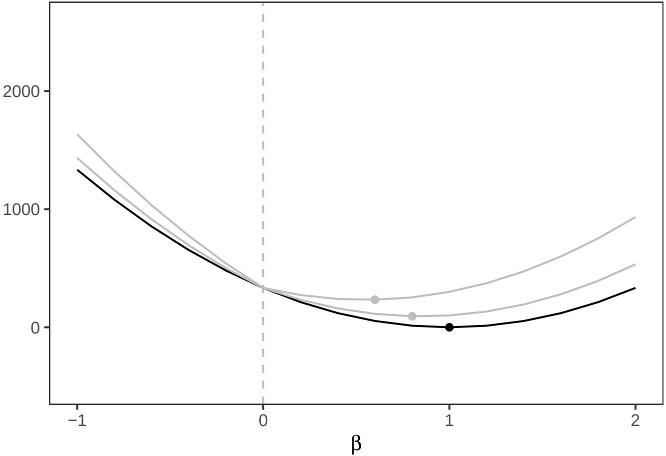
Intuición en 1 Dimension



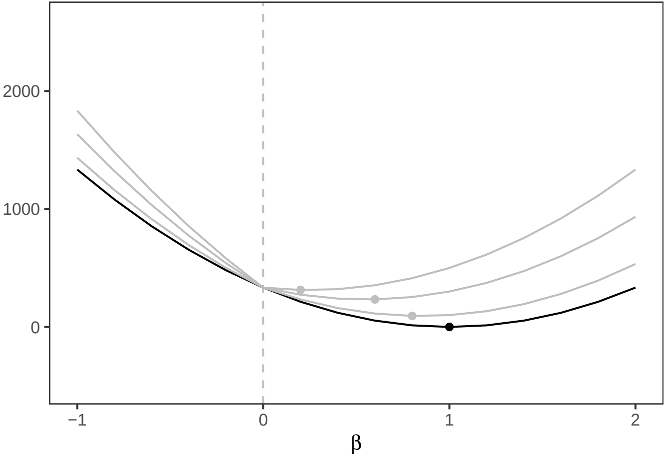
Intuición en 1 Dimension



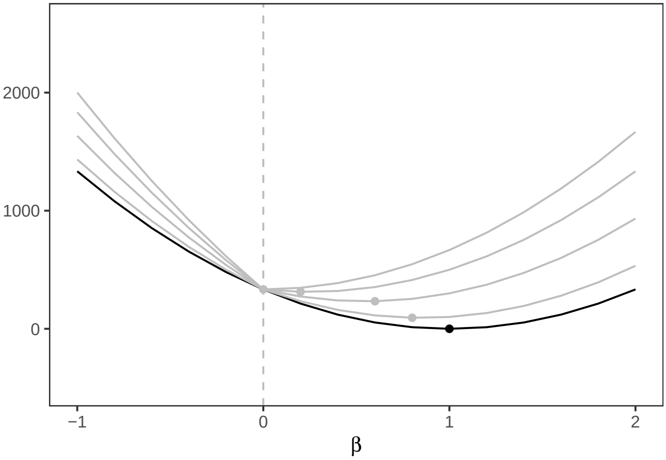
Intuición en 1 Dimension



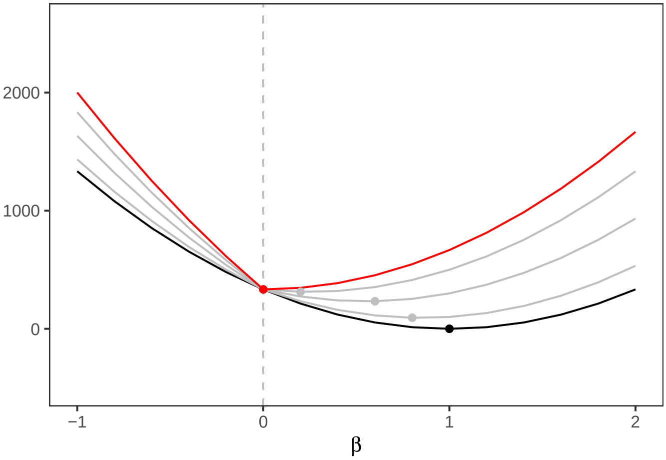
Intuición en 1 Dimension



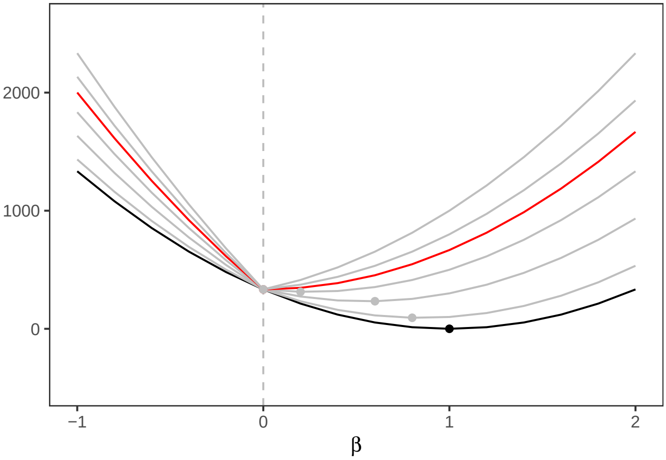
Intuición en 1 Dimension



Intuición en 1 Dimension



Intuición en 1 Dimension



Intuición en 1 Dimension

Ejemplo en R

```
require("glmnet")
X<-model.matrix(~bedrooms,matchdata)
y<-matchdata$price

lasso.mod <- glmnet(X, y, alpha = 1, lambda = 0)
lasso.mod$beta
```

```
## 2 x 1 sparse Matrix of class "dgCMatrix"
##                   s0
## (Intercept)        .
## bedrooms     20130.9
```

```
lm(y~X-1)$coef
```

```
## X(Intercept)    Xbedrooms
##     219722.3      20130.9
```

Intuición en 1 Dimension

Ejemplo en R

```r
lasso.mod <- glmnet(X, y, alpha = 1, lambda = 1000)
lasso.mod$beta
```

```
## 2 x 1 sparse Matrix of class "dgCMatrix"
##                      s0
## (Intercept)       .
## bedrooms     18780.04
```

```r
lasso.mod <- glmnet(X, y, alpha = 1, lambda = 1e5)
lasso.mod$beta
```

```
## 2 x 1 sparse Matrix of class "dgCMatrix"
##                s0
## (Intercept)  0
## bedrooms     .
```

Intuición en 1 Dimension

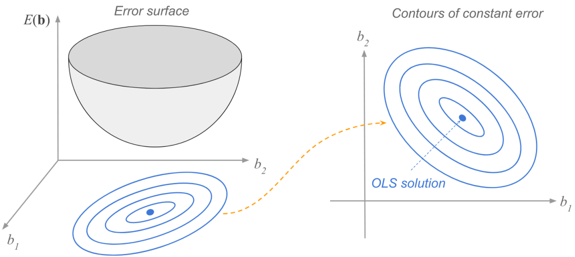
$$min_{\beta}E(\beta) = \sum_{i=1}^{n}(y_i - x_i\beta)^2 + \lambda|\beta| \tag{4}$$

la solución analítica es

$$\hat{\beta}_{lasso} = \begin{cases} 0 & \text{si } \lambda \geq \lambda^* \\ \hat{\beta}_{OLS} - \frac{\lambda}{2} & \text{si } \lambda < \lambda^* \end{cases} \tag{5}$$

Intuición en 2 Dimensiones (OLS)

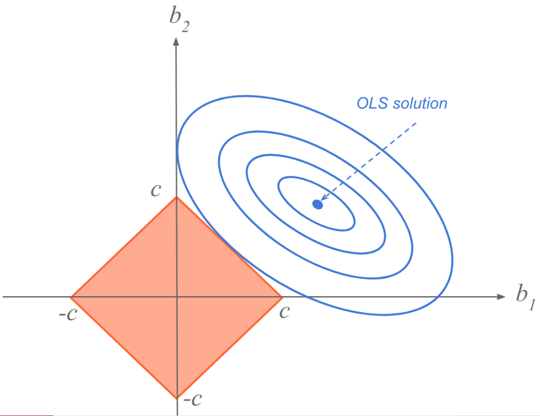
$$min_\beta E(\beta) = \sum_{i=1}^{n}(y_i - x_{i1}\beta_1 - x_{i1}\beta_2)^2 \quad (6)$$



Fuente: https://allmodelsarewrong.github.io

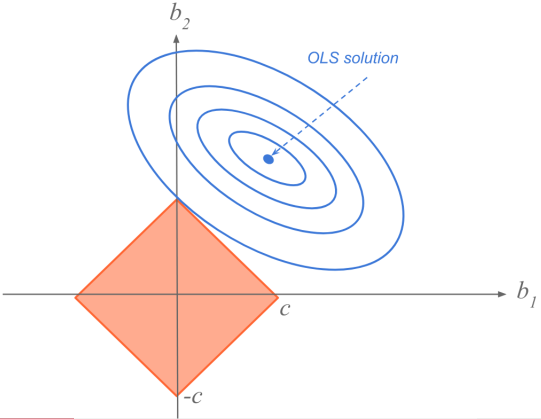
Intuición en 2 Dimensiones (Lasso)

$$min_{\beta} E(\beta) = \sum_{i=1}^{n} (y_i - x_{i1}\beta_1 - x_{i1}\beta_2)^2 \text{ s.a } (|\beta_1| + |\beta_2|) \leq c \tag{7}$$



Intuición en 2 Dimensiones (Lasso)

$$min_\beta E(\beta) = \sum_{i=1}^{n}(y_i - x_{i1}\beta_1 - x_{i1}\beta_2)^2 \text{ s.a } (|\beta_1| + |\beta_2|) \leq c \tag{8}$$



Ridge

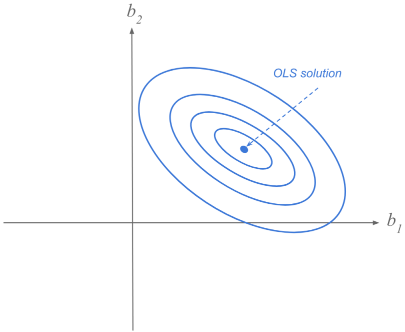
- Para un $\lambda \geq 0$ dado, consideremos ahora el siguiente problema de optimización

$$min_\beta E(\beta) = \sum_{i=1}^{n}(y_i - \beta_0 - x_{i1}\beta_1 - \cdots - x_{ip}\beta_p)^2 + \lambda \sum_{j=1}^{p}(\beta_j)^2 \tag{9}$$

- La intuición es similar a lasso, pero la vamos a extender a 2-Dim

Intuición en 2 Dimensiones (OLS)

$$min_\beta E(\beta) = \sum_{i=1}^{n}(y_i - x_{i1}\beta_1 - x_{i1}\beta_2)^2 \tag{10}$$



Fuente: https://allmodelsarewrong.github.io

Intuición en 2 Dimensiones (Ridge)

- Al problema

$$min_{\beta} E(\beta) = \sum_{i=1}^{n} (y_i - \beta_0 - x_{i1}\beta_1 - \cdots - x_{ip}\beta_p)^2 + \lambda \sum_{j=1}^{p} (\beta_j)^2 \tag{11}$$

- podemos escribirlo como

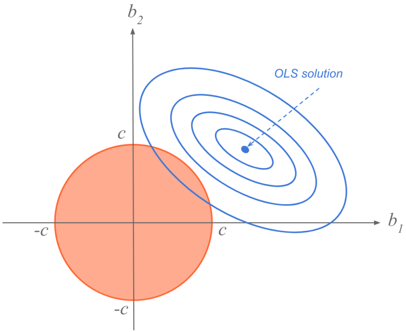
$$min_{\beta} E(\beta) = \sum_{i=1}^{n} (y_i - x_{i1}\beta_1 - x_{i1}\beta_2)^2 \tag{12}$$

$$\text{sujeto a}$$

$$\left((\beta_1)^2 + (\beta_2)^2\right) \leq c$$

Intuición en 2 Dimensiones (Ridge)

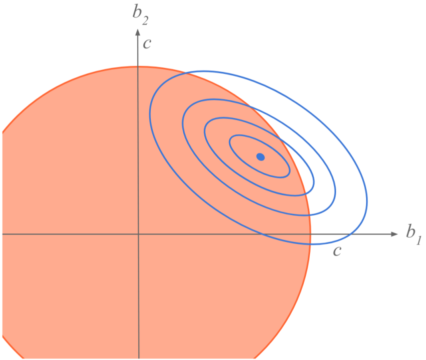
$$min_{\beta}E(\beta) = \sum_{i=1}^{n}(y_i - x_{i1}\beta_1 - x_{i1}\beta_2)^2 \text{ s.a } \left((\beta_1)^2 + (\beta_2)^2\right) \leq c \tag{13}$$



Fuente: `https://allmodelsarewrong.github.io`

Intuición en 2 Dimensiones (Ridge)

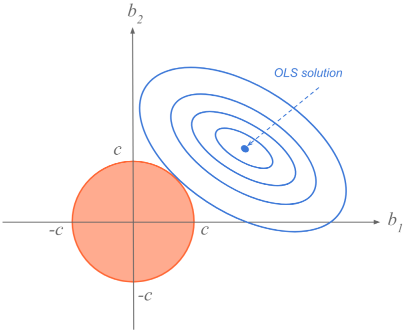
$$min_{\beta}E(\beta) = \sum_{i=1}^{n}(y_i - x_{i1}\beta_1 - x_{i1}\beta_2)^2 \text{ s.a } \left((\beta_1)^2 + (\beta_2)^2\right) \leq c \tag{14}$$



Fuente: https://allmodelsarewrong.github.io

Intuición en 2 Dimensiones (Ridge)

$$min_{\beta}E(\beta) = \sum_{i=1}^{n}(y_i - x_{i1}\beta_1 - x_{i1}\beta_2)^2 \text{ s.a } \left((\beta_1)^2 + (\beta_2)^2\right) \leq c \tag{15}$$



Fuente: `https://allmodelsarewrong.github.io`

Comentarios técnicos

- Lasso y ridge son sesgados, pero las disminuciones en varianza pueden compensar esto y llevar a un MSE menor
- Lasso encoje a cero, Ridge no tanto
- Importante para aplicación:
  - Estandarizar los datos (media 0, y varianza 1)
  - Como elegimos λ?

Comentarios técnicos: selección de λ

- Como elegimos λ?
- λ es un parámetro y lo elegimos usando validación cruzada
  1. Partimos la muestra de entrenamiento en K Partes: $M_{train} = M_{fold\,1} \cup M_{fold\,2} \cdots \cup M_{fold\,K}$
  2. Cada conjunto $M_{fold\,K}$ va a jugar el rol de una muestra de evaluación $M_{eval\,k}$. Entonces para cada muestra
     - $M_{train-1} = M_{train} - M_{fold\,1}$
     - $\vdots$
     - $M_{train-k} = M_{train} - M_{fold\,k}$
  3. Luego hacemos el siguiente loop
     1. Para $\lambda_i = 0, 0.001, 0.002, \ldots, \lambda_{max}$
        - Para $k = 1, \ldots, K$
          - Ajustar el modelo $m_{i,k}$ con λ_i en $M_{train-k}$
          - Calcular y guardar el $MSE(m_{i,k})$ usando M_{eval-k}
        - fin para k
        - Calcular y guardar $MSE_i = \frac{1}{K} MSE(m_{i,k})$
     2. fin para λ
  4. Encontrar el menor MSE_i y usar ese $\lambda_i = \lambda^*$

Naive Elastic Net

- Elastic net: happy medium.
  - Good job at prediction and selecting variables

$$min_\beta NEL(\beta) = \sum_{i=1}^{n}(y_i - x_i'\beta)^2 + \lambda_1 \sum_{s=2}^{p} |\beta_s| + \lambda_2 \sum_{s=2}^{p} \beta_s^2 \tag{16}$$

- Mixes Ridge and Lasso
- Lasso selects predictors
- Strict convexity part of the penalty (ridge) solves the grouping instability problem
- H.W.: $\beta_{OLS} > 0$ one predictor standarized

$$\hat{\beta}_{naive\,EN} = \frac{\left(\hat{\beta}_{OLS} - \frac{\lambda_1}{2}\right)_+}{1+\lambda_2} \tag{17}$$

Elastic Net

- Elastic Net: reescaled version
- Double Shrinkage introduces "too" much bias, *final* version "corrects" for this

$$\hat{\beta}_{EN} = \frac{1}{\sqrt{1+\lambda_2}}\hat{\beta}_{naive\,EN} \tag{18}$$

- Careful sometimes software asks.
- How to choose (λ_1, λ_2)? $\rightarrow$ Bidimensional Crossvalidation
- Zou, H. & Hastie, T. (2005)

Regularization Demo

```r
#Load the required packages
library("dplyr") #for data wrangling
library("caret") #ML

data(swiss) #loads the data set

set.seed(123) #set the seed for replication purposes
str(swiss) #conmpact display
```

```
## 'data.frame':    47 obs. of  6 variables:
##  $ Fertility       : num  80.2 83.1 92.5 85.8 76.9 76.1 83.8 92.4 82.4 82.9 ...
##  $ Agriculture     : num  17 45.1 39.7 36.5 43.5 35.3 70.2 67.8 53.3 45.2 ...
##  $ Examination     : int  15 6 5 12 17 9 16 14 12 16 ...
##  $ Education       : int  12 9 5 7 15 7 7 8 7 13 ...
##  $ Catholic        : num  9.96 84.84 93.4 33.77 5.16 ...
##  $ Infant.Mortality: num  22.2 22.2 20.2 20.3 20.6 26.6 23.6 24.9 21 24.4 ...
```

Regularization Demo

```r
ols <- train(Fertility ~ .,    # model to fit
                    data = swiss,
                    trControl = trainControl(method = "cv", number = 10),
                    # Method: crossvalidation, 10 folds
                    method = "lm")
                    # specifying regression model
ols
```

```
## Linear Regression
##
## 47 samples
##  5 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 42, 42, 42, 42, 42, 44, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   7.424916  0.6922072  6.31218
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

Regularization Demo

```
lambda <- 10^seq(-2, 3, length = 100)
lasso <- train(
  Fertility ~., data = swiss, method = "glmnet",
  trControl = trainControl("cv", number = 10),
  tuneGrid = expand.grid(alpha = 1, lambda=lambda), preProcess = c("center", "scale")
  )

lasso
```

```
## glmnet
##
## 47 samples
##  5 predictor
##
## Pre-processing: centered (5), scaled (5)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 43, 43, 43, 42, 42, 41, ...
## Resampling results across tuning parameters:
##
##  ...
##
## Tuning parameter 'alpha' was held constant at a value of 1
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were alpha = 1 and lambda = 0.02009233.
```

Regularization Demo

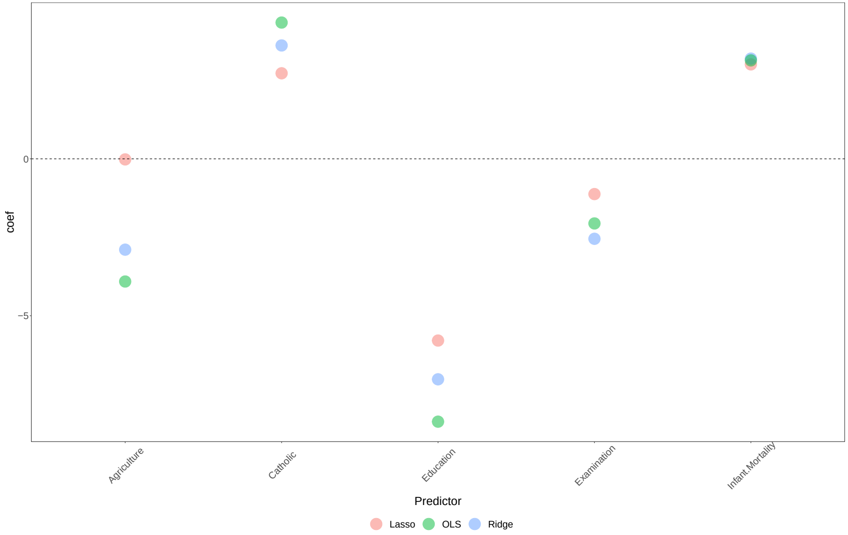
```r
ridge <- train(
  Fertility ~., data = swiss, method = "glmnet",
  trControl = trainControl("cv", number = 10),
  tuneGrid = expand.grid(alpha = 0,lambda=lambda), preProcess = c("center", "scale")
  )
ridge
```

```
## glmnet
##
## 47 samples
##  5 predictor
##
## Pre-processing: centered (5), scaled (5)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 42, 42, 43, 44, 42, 42, ...
## Resampling results across tuning parameters:
##
##   ...
##
## Tuning parameter 'alpha' was held constant at a value of 0
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were alpha = 0 and lambda = 0.7390722.
```

Regularization Demo

```
##
## Call:
## summary.resamples(object = ., metric = "RMSE")
##
## Models: ridge, lasso
## Number of resamples: 10
##
## RMSE
##           Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
## ridge 2.615430 4.674108 7.627190 6.923531 8.939798 10.55026    0
## lasso 3.205868 5.553161 5.961622 7.324069 8.587818 13.46074    0
```

Regularization Demo



Further Readings

- Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical learning (Vol. 1, No. 10). New York: Springer series in statistics.
- Kuhn, M. (2012). The caret package. R Foundation for Statistical Computing, Vienna, Austria. https://topepo.github.io/caret/index.html
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer.
- Zou, H. & Hastie, T. (2005). Regularization and variable selection via the elastic net. Journal of the Royal StatisticalSociety, Series B.67: pp. 301–320