Machine Learning for Economists **Summer 2024**

University of Cologne **Due: April 24, 23:59**

Lab 1: Regularization

Please send your solutions (RMarkdown/Quarto or Jupyter Notebook with code and answers plus a version compiled into pdf format) to tom.zimmermann@uni-koeln.de

In this lab, we investigate the effect of regularization on prediction.[1] Our setting is very simple and in line with the linear model in the lecture notes. In particular, we estimate the model

$$y_i = \beta_1 x_{1i} + \beta_2 x_{2i} + \dots \beta_p x_{pi} + \epsilon_i$$

and we set

$$\beta_1 = 2, \beta_2 = 3, \beta_j = 0 \text{ for } j > 2$$

We will work on a simulation exercise, i.e. we

- simulate data according to the model above many times
- estimate the model parameters using standard OLS, LASSO or Ridge
- compare bias and variance for an out-of-sample prediction at some point x_0.

Here is some code for the simulation to get you started.

```r
library(glmnet)
library(tidyverse)

nsim = 100  # Number of simulations
nobs = 100  # Number of observations in the simulated dataset
p = 10      # Number of regressors
beta = c(2,3, rep(0,p-2))  # Coefficient vector
                          # (2 for beta1, 3 for beta2,
                          # 0s for remaining betas)

x0 = c(rep(2, p))  # out-of-sample observation

# Initialize vectors so that we can store results
f0_ols   = c()
f0_lasso = c()
f0_ridge = c()

for (s in 1:nsim) {

  X = MASS::mvrnorm(nobs, #Randomly drawn X variables
                          mu = rep(0, p),
                          Sigma = diag(p)
                          )

  y = X %*%beta + rnorm(nobs)  # Simulate y with normal noise

  # Fit OLS model
  fitOLS = lm(y~X)
```

[1] Problem based on an example by Stephen Hansen.

```
  f0_ols[s] = fitOLS$coefficients%*%c(1, x0)  # Prediction at x0

  # Fit LASSO regression
  fitLASSO = cv.glmnet(X, y, alpha = 1)
  f0_lasso[s] = t(coef(fitLASSO, s="lambda.1se"))%*%c(1, x0) %>% as.numeric()

  # Fit Ridge regression
  fitRidge = cv.glmnet(X, y, alpha = 0)
  f0_ridge[s] = t(coef(fitRidge, s="lambda.1se"))%*%c(1, x0) %>% as.numeric()

}
```

Questions

Before you answer the questions, make sure you understand the simulation code (For LASSO and Ridge regressions, the code makes predictions at a particular value of λ that is denoted as `lambda.1se`. We discuss next week why this is reasonable, for now, you can just take this as given.)

1. We will consider out-of-sample predictions at $x_0 = (2, 2, \ldots, 2)$. What value of $f(x_0)$ do you expect?
2. Run the code as is. Plot the distribution of predictions $\hat{f}(x_0)$ for the different models. What do you observe?
3. Compute bias, variance and mean squared error for the three different models at x_0, our out-of-sample observation.
4. How do bias and variance depend on the number of irrelevant regressors?
5. How do bias and variance depend on the number of observations?
6. The simulation above assumes that regressors are uncorrelated. How do results change when correlation between regressors is instead given by $\rho > 0$?

Hint: You can draw from a multivariate normal distribution with correlated variables by changing `Sigma` in the `mvrnorm` function. You can construct the covariance matrix with two lines of code:

```
rho = .5  # Assumed regressor correlation
sigma = matrix(rho, nrow = p, ncol = p)
diag(sigma) = 1
```