Presenting groups: 9 and 10, Date: 16.6.2021

Exercise 1:

Consider the following data generating process with $n = 1000$ observations and $p = 50$ covariates. Initially assume $\mathbf{X} \sim \mathcal{N}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ where $\boldsymbol{\mu} = (0, \ldots, 0)^\top$ and $\boldsymbol{\Sigma}$ is the covariance matrix with the variances on the diagonal (values chosen by you) and zeros on the off-diagonal. The true coefficients range from 0.1 to 0.5 (you can sample values from this range or use equispaced values on that interval) and the errors are drawn from a normal distribution $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, 1)$.

The aim of this exercise: compare ridge regression and lasso.

a) Write a function to calculate the ridge and lasso for a wide range of λ. You may take the exponential range from 10^{-2} to 10^2, for example

b) Calculate test MSE for both methods on a test dataset with the same number of observations. Plot the results.

c) Find an optimal λ (the one that minimizes MSE) using k-fold cross-validation for several values of k.

d) Find an optimal value of λ using the test MSE.

e) Compare performance of ridge, lasso and OLS approaches using the values of λ picked in c) and d).

Exercise 2 (Simulation Study)**:**

Evaluate the difference in prediction performance of these methods in a simulation study by changing the dgp in the following way.

a) Increase the number of regressors.

b) Increase sparsity of the true coefficients.

c) Propose a manipulation of the dgp that illustrates the case when lasso outperforms the ridge regression and vice versa.

You do not have to program the `glmnet` functions yourself (although you may, of course). Some helpful packages, libraries and commands:

```
library(glmnet) ###required to use the command below

glmnet() ###performs ridge and lasso
cv.glmnet() ###automatically computes the cross-validation estimates for glmnet
```