Extra exercise 3.2
Simulation experiment with OLS, best subset regression and ridge regression

This is exercise is mainly and extension of Extra exercise 3.1, where we include best subset regression and ridge regression in addition to OLS. We also change the simulation experiment slightly, and simulate new X-matrices for each simulation.

Consider the linear regression model

$$\begin{aligned} Y &= f(X) + \varepsilon \\ &= \beta_0 + \sum_{j=1}^{j=p} X_j \beta_j + \varepsilon, \end{aligned} \tag{1}$$

where $p = 15$,
$X = X_1, \ldots, {X_p}^T$,
$\beta_0 = 3$,
$\beta_j = 2$ for $j = 1, \ldots, 5$,
$\beta_j = 1$ for $j = 6, \ldots, 10$,
$\beta_j = 0$ for $j = 11, \ldots, 15$, and
$\text{Var}(\varepsilon) = 25$.

Furthermore, assume that

$$X = N_p(0, \Sigma), \tag{2}$$

i.e. multivariate normal with zero mean and covariance matrix Σ, where all the diagonal elements of Σ are 1 and all non-diagonal elements are 0.8. The Σ is also the correlation matrix with all correlations equal to 0.8.

a)

Generate $N = 20$ observations of the inputs from their multivariate normal distribution (2) and put them into a matrix $\mathbf{X}^{train}$. You can use the function `mvrnorm` from the R package `MASS`. For the given $\mathbf{X}^{train}$, simulate the corresponding output vector $\mathbf{y}^{train}$ from the model (1). Together, $\mathbf{y}^{train}$ and $\mathbf{X}^{train}$ constitute the training set.

Estimate the regression model by best subset regression, where the number of input variables used in the subset is selected by 10-fold cross validation. It can be useful to use the function `leaps` from the R package `leaps` for the best subset algorithm.

Furthermore, estimate the regression model by ridge regression, where the penalty parameter is selected by 10-fold cross validation. It can be useful to use the

function `lm.ridge` from the R package `MASS` for the ridge regression. Another possibility is to use the function `glmnet` from the R package `glmnet`. (This function has a build in cross validation procedure, but in this series of exercises (Extra 3.2, 3.3 and 3.4) it is more useful to program a separate cross validation procedure that is common for all methods that we will investigate.)

b)

Perform now the following simulation experiment with 1000 repetitions:

- Generate $N = 20$ observations of the inputs from their multivariate normal distribution (2) and put them into a matrix $\mathbf{X}^{train}$, and then generate $N = 1000$ observations and put them into a matrix $\mathbf{X}^{test}$
- For the given $\mathbf{X}^{train}$, simulate the corresponding output vector $\mathbf{y}^{train}$ from the model (1). Together, $\mathbf{y}^{train}$ and $\mathbf{X}^{train}$ constitute the training set.
- Simulate also an output vector $\mathbf{y}^{test}$ conditioned on $\mathbf{X}^{test}$. Together, these constitute a test set.
- Estimate the β vector by ordinary least squares (OLS), for instance by using the `lm` function in R. Furthermore, predict $\mathbf{y}^{test}$ by $\hat{f}(\mathbf{X}^{test}) = \mathbf{X}^{test}\hat{\beta}$.
- Estimate the β vector by best subset regression with number of inputs chosen be 10-fold cross validation, and ordinary least squares (OLS), and predict $\mathbf{y}^{test}$.
- Estimate the β vector by ridge regression with number of inputs chosen be 10-fold cross validation, and ordinary least squares (OLS), and predict $\mathbf{y}^{test}$.
- For each prediction method; compute the bias, variance and mean squared error of each of the β coefficients by averaging over the simulations, and compare with the theoretical values. Compute also the bias, variance and mean squared error (MSE) of $\hat{f}(\mathbf{X}^{test})$ averaged over the values of the inputs in the test set and over all simulations and finally compute the corresponding prediction error.

Repeat the simulation experiment, first with $N = 100$ and then with $N = 1000$ observations in the training set. You can then reduce the number of simulations to 500 and 100, respectively.