Statistical Learning **Homework assignment 2**
Grün **SS 2024**

2 Model assessment and selection

Exercise 1:

We will use the AIC for model selection and compare the AIC values obtained to in-sample error estimates based on test data.

(a) Generate a simulated data set as follows:

```
> set.seed(1)
> x <- rnorm(100)
> y <- x - 2*x^2 + rnorm(100)
```

(b) Calculate the AIC values when fitting the following four models using least squares:

i. $Y = \beta_0 + \beta_1 X + \epsilon$
ii. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \epsilon$
iii. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \epsilon$
iv. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \beta_4 X^4 + \epsilon$

(c) Determine in-sample error estimates by drawing suitable test data from the data generating process using twice the negative log-likelihood as loss function and compare the values to the AIC values obtained.

Exercise 2:

We will perform leave-one-out cross-validation (LOOCV), k-fold cross-validation (kCV) and empirical bootstrapping on a simulated data set.

(a) Generate a simulated data set as follows:

```
> set.seed(1)
> x <- rnorm(100)
> y <- x - 2*x^2 + rnorm(100)
```

(b) Set a random seed, and then compute the LOOCV, kCV and empirical bootstrap errors based on the mean squared error loss that result from fitting the following four models using least squares:

i. $Y = \beta_0 + \beta_1 X + \epsilon$
ii. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \epsilon$
iii. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \epsilon$
iv. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \beta_4 X^4 + \epsilon$

(c) Repeat (b) using another random seed, and report your results. Are your results the same as what you got in (b)? Why?

(d) Which of the models in (b) had the smallest LOOCV, kCV and empirical bootstrap error? Is this what you expected? Explain your answer.

(e) Comment on the statistical significance of the coefficient estimates that result from fitting each of the models in (b) using least squares. Do these results agree with the conclusions drawn based on the cross-validation and bootstrap results?

Exercise 3:

In the following we use the wage data set available as data object `Schooling` in package **Ecdat**.

- Omit observations with missing values and the variable `wage76`. Mutate the variable `mar76` into a binary variable which is `TRUE` for value `"married"` and `FALSE` otherwise.
- Fit regularized linear regression models with `lwage76` as dependent variable using only linear effects for the covariates. Vary the α parameter for the elastic from 0 to 1 in step sizes of 0.2. Generate the argument `foldid` only once and specify it when calling `cv.glmnet()` to ensure that the same folds are used.
- Perform 10-fold cross-validation considering the MSE for a range of penalty values of λ and fixed value of α and visualize the results (e.g., using the default plot method for objects returned by `cv.glmnet` from package **glmnet**) for the different values of α.
- Select the best value for λ for a fixed value of α using either the value which minimizes the cross-validation loss or the $1-$ SE rule and compare the selected models (e.g., based on complexity, predicted values and classification performance). Compare the performance across different values of α.
- Inspect the best solution for $\alpha = 1$ using the $1-$ SE rule and assess the variables selected and the estimated coefficients as well as the correlation between the predicted and the observed values (as a measure of goodness-of-fit).

Exercise 4:

In the following use the South African heart disease data available as data object `SAheart` in package **ElemStatLearn**.

- Fit a logistic regression model with Lasso penalty using only linear effects for the covariates.
- Perform 20-fold cross-validation considering the deviance loss for a range of penalty values and visualize the results (e.g., using the default plot method for objects returned by `cv.glmnet` from package **glmnet**).
- Select the penalty value using either the value which minimizes the cross-validation loss or the $1-$ SE rule and compare the selected models (e.g., based on complexity, predicted values and classification performance).

  (*Hint*: Functions `coef()` and `predict()` can be used with objects returned by `cv.glmnet` and have an argument `s` which can be specified as `"lambda.min"` and `"lambda.1se"` to select the λ value where the loss is minimized or within one standard error.)

Exercise 5:

The dataset `phoneme` in package **ElemStatLearn** contains data from an acoustic-phonetic continuous speech corpus. There are five classes contained in the dataset. The covariates are log-periodograms of length 256. In the following two-group classification is performed using only the classes `"aa"` and `"ao"`.

- Visualize the data by plotting the covariate values on the y-axis and the index on the x-axis using line plots. Use different colors for the two classes.
- Select 1000 samples as training data and use the remaining ones as test data.

- Fit a logistic regression model to the training data using all covariates and determine the misclassification rate and the average log-likelihood value on the training and test data.
- The complexity of this model can be reduced by restricting the regression coefficients to vary only smoothly over the covariates, i.e., regression coefficients for close covariates are similar.

  This can be achieved using splines. For example the following transformation creates a 12-dimensional model matrix X^* based on natural cubic splines which can be used to fit the logistic regression model instead of the 256-dimensional X:

  ```
  > library("splines")
  > H <- ns(1:256, df = 12)
  > X.star <- X %*% H
  ```

  Fit a logistic regression model to the training data using X^* as model matrix and determine the misclassification rate and the average log-likelihood value on the training and test data.
- Vary the degrees of freedom in the spline basis expansion using 2 to the power of 1 to 8, i.e., 2 to 256. Calculate the misclassifiation rate and the mean log-likelihood on the training and test data for each of the fitted models.
- Compare the misclassification rates and mean log-likelihoods based on training and test data sets visually for the different degrees of freedom. Interpret the results.