Statistical and Machine Learning **Homework assignment 3**
Grün **WS 2023/24**

3 Model assessment and selection

Set a random seed in your R script before doing any analysis involving drawing (pseudo-)random numbers.

Exercise 1:

In the following we assume that we have a fixed design for the inputs, i.e., $\boldsymbol{X} \in \mathbb{R}^{N \times p}$ is deterministic.

The dependent variable is assumed to result from the following data generating process

$$\boldsymbol{y} = \boldsymbol{X}\beta + \boldsymbol{\epsilon},$$

where $\boldsymbol{y} \in \mathbb{R}^N$, $\beta \in \mathbb{R}^p$ and

$$\mathrm{E}[\boldsymbol{\epsilon}] = \boldsymbol{0}, \qquad \mathrm{Var}[\boldsymbol{\epsilon}] = \sigma^2 \boldsymbol{I}_N,$$

with $\boldsymbol{I}_N$ the identity matrix of dimension N.

For a given training data set $\mathcal{T}$ of size N assume that the OLS estimate for the regression coefficients is given by

$$\hat{\beta}_{\mathcal{T}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{y}_{\mathcal{T}}.$$

Determine the following:

- Determine the expected training error:

$$\mathrm{E}_{\mathcal{T}} \left[\frac{1}{N} (\boldsymbol{y}_{\mathcal{T}} - \boldsymbol{X}\hat{\beta}_{\mathcal{T}})^\top (\boldsymbol{y}_{\mathcal{T}} - \boldsymbol{X}\hat{\beta}_{\mathcal{T}}) \right].$$

- Determine the expected in-sample error:

$$\mathrm{E}_{\mathcal{T}} \left[\mathrm{E}_{\boldsymbol{y}} \left[\frac{1}{N} (\boldsymbol{y} - \boldsymbol{X}\hat{\beta}_{\mathcal{T}})^\top (\boldsymbol{y} - \boldsymbol{X}\hat{\beta}_{\mathcal{T}}) | \mathcal{T} \right] \right].$$

- Determine the difference between the expected training error and the expected in-sample error.

Exercise 2:

Assume the following data generating process:

$$y = x + x^2 + \epsilon,$$

with $\epsilon \sim N(0, \sigma_\epsilon^2)$ and $x \sim N(0, \sigma_x^2)$ and x and ϵ independent.

- Determine analytically the test error using the squared error loss given parameter estimates $\hat{\boldsymbol{\beta}}$.
- Assume $\sigma_\epsilon^2 = \sigma_x^2 = 1$. Draw a sample of size $N = 40$ as training data and determine the test error using the squared error loss using the analytical formula as well as simulation when estimating the regression coefficients $\boldsymbol{\beta}$ using OLS.

Exercise 3:

Consider the in-sample prediction error $\mathrm{Err}_{\mathrm{in}}$ and the training error $\overline{\mathrm{err}}$ in the case of squared-error loss:

$$\mathrm{Err}_{\mathrm{in}} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{E}_{Y^0}[(Y_i^0 - \hat{f}(x_i))^2]$$

$$\overline{\mathrm{err}} = \frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{f}(x_i))^2.$$

Establish that the average optimism in the training error is

$$\frac{2}{N}\sum_{i=1}^{N} \mathrm{Cov}(\hat{y}_i, y_i).$$

Exercise 4:

Assume $\boldsymbol{y}$ arises from an additive-error model $Y = f(X) + \epsilon$ with $\mathrm{Var}(\epsilon) = \sigma_\epsilon^2$ and $\hat{\boldsymbol{y}} = \boldsymbol{S}\boldsymbol{y}$.

Show that

$$\sum_{i=1}^{N} \mathrm{Cov}(\hat{y}_i, y_i) = \mathrm{trace}(\boldsymbol{S})\sigma_\epsilon^2.$$

Exercise 5:

Assume for N observations $\boldsymbol{y} = (y_1, \ldots, y_N)$ the following model: They are drawn i.i.d. from a Poisson distribution with parameter λ. Further assume that the prior distribution for λ is an improper prior which is proportional to a constant on the positive reals.

- Determine an approximation of the marginal likelihood based on the Laplace approximation given by

$$p(\boldsymbol{y}|\mathcal{M}) \approx \exp(\ell(\boldsymbol{y}|\hat{\lambda}))\sqrt{\frac{2\pi}{\mathcal{J}(\hat{\lambda})}},$$

  where $\ell(\boldsymbol{y}|\lambda)$ is the log-likelihood of the data assuming that the observations are i.i.d. data from a Poisson distribution with parameter λ, $\hat{\lambda}$ is the maximum likelihood estimate and $\mathcal{J}(\lambda)$ is the observed information matrix, i.e., the second derivative of the log-likelihood function evaluated at λ.

- Determine -2 times the logarithm of the approximation and compare the result to the Bayesian information criterion for this model.

Exercise 6:

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) Create a scatterplot of X against Y. Comment on what you find.

(c) 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. $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \beta_4 X^4 + \epsilon$

(d) Determine in-sample error estimates using twice the negative log-likelihood as loss function. Obtain an estimate by drawing 100 test data sets from the data generating process where the x values are the same as in the training data and averaging over the negative log-likelihood values when inserting the values of the test data set and the predicted values. Compare these in-sample error estimates to the AIC values obtained.

Exercise 7:

We will perform leave-one-out cross-validation (LOOCV) and k-fold cross-validation (kCV) 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 and kCV 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 and kCV error? Is this what you expected? Explain your answer.

Exercise 8:

We perform a simulation study to assess how good the performance of the LASSO is for variable selection:

- The following data generating process is used:
  - Draw a 100-dimensional vector from a standard multivariate normal distribution.
  - Determine the dependent variable with $\epsilon \sim N(0, 0.1)$ (i.e., $\sigma^2 = 0.1$) by:

$$y = \sum_{i=1}^{10} x_i + \epsilon.$$

- Draw 100 data sets of size 1000. Split each data set into a training data set containing the first 100 observations and a test data set containing the remaining 900 observations. For each of the 100 repetitions use `glmnet` from the **glmnet** package to fit the LASSO model for different values of λ to the training data set and select the λ value where predictive performance is best on the test data set.

- Determine the proportion of correctly included coefficients from all relevant ones (true positive rate) and the proportion of wrongly included coefficients from all irrelevant ones (false positive rate) for each of the 100 data sets and visualize the distribution of the two rates.

Exercise 9:

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 $1 - \text{SE}$ 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 10:

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. Determine also the AIC values for each fitted model.
- Compare the misclassification rates and mean log-likelihoods based on training and test data sets and the AIC values visually for the different degrees of freedom. Interpret the results.