Homework: Cross-Validation

4

a) Load the `Wage` dataset from the package `ISLR` as follows:

```
library(ISLR)
data(Wage)

x <- model.matrix(wage ~ ., Wage)
y <- Wage$wage
```

The additional call to `model.matrix()` is necessary in order to transform the factors into suitable dummy variables.

Then split the dataset into a training (90 %) and a test set (10 %). Afterwards, estimate a linear model on the training data that predicts the wage given all other variables. Calculate the the mean squared error (MSE) between the predicted values and the actual values on the test set.

How does the seed value for the random number affect the outcome? For this purpose, use initial seeds of 57 and 283.

3

b) Use the code from the previous exercise and implement the Leave-One-Out Cross-Validation (LOOCV) manually. What is its mean squared error now? What is the advantage of the LOOCV approach?

Note: This may take a while to compute.

3

c) Once again, use the code from the previous exercise and implement k-fold Cross-Validation. Calculate the mean squared error with $k = 10$ and $k = 5$ folds. Do they differ severely?

Hint: there is a function `split` that divides a dataset into groups.

3

d) Use the `Auto` dataset from the library `ISLR` and predict the `mpg` variable using `horsepower` using a polynomial model. Repeat the analysis for $M = 10$ different random splits. Plot the different MSE curves across degrees $p = 1, \ldots, 10$ of the polynomial.

Homework: Model Tuning

3

a) Load the `GermanCredit` dataset from the `caret` package. Use this package also to train a generalized additive (GAM) model with splines (`method="gamSpline"`). The model should predict the credit class given all other variables. Train the model using the values from 1 to 10 for the degrees of freedom (`df`) of the GAM model.

Furthermore, the `trainControl()` function accepts an argument `repeat` which repeats the tuning process multiple time as means to get a bootstrapped prediction. It then automatically returns the mean value of the predictions. Use 3 repetitions and, as a prerequisite, also change its method parameter (`method="boot"`).

Plot how the tuning parameters affect the model performance. What is the best tuning parameter?

Homework: Bootstrapping

3

a) Load the `Carseats` dataset from the library `ISLR`. Compute the mean of the `Sales` variable. In a next step, we want to understand the potential distribution of the mean. For this purpose, create 20 boostrapped replicates of the data. Then apply the mean function to each boostrapped replicate. What is the range of the boostrapped mean values?

3

b) Repeat the above analysis by using `boot()`. In addition, what are the confidence intervals and how does the distribution look like?

3

c) Given is the following data for which we want to determine confidence intervals.

```
set.seed(0)
x <- rexp(200)
```

- First of all, compute the 5 % confidence intervals assuming a normal distribution with unknown variance. For this purpose, one can use the formula

$$\mu \pm t_{\alpha/2} \frac{\sigma}{\sqrt{n}}$$

for observations $x_1, \ldots, x_n$ with mean μ and standard deviation σ given a α % confidence interval. The variable $t_{\alpha/2}$ denotes the $100\,(1-\alpha/2)$-th percentile of the Student-t distribution with $n-1$ degrees of freedom. Its values can be found with `qt(1 - alpha/2, df=n-1)`. Thereby, one can quantify the uncertainty of a point estimate of the population mean.
- Why shouldn't we rely upon the above formula for the given data?
- Instead, compute the confidence intervals using bootstrapping?

3

d) Given is above data for which we want to determine confidence intervals. How does the mean estimate from the bootstrap and the width of the 95 % change when increasing the number of bootstrap replications B? Test values $B = 50, 100, 500, 1000, 5000, 10000$.

3

e) Use the dataset `GermanCredit` in order to estimate a logit model. The credit class should be explained in terms of age and gender. Bootstrap the coefficients.

3

f) The boostrap resampling picks some values more than once, and others not at all. We want to understand now the average behavior. How many observations of a dataset with $N = 400$ entries are not selected in a bootstrapped sample? How does it change if the dataset contains only $N = 10$ entries?

In a next step, we want to understand the asymptotic behavior. Derive a mathematical function that gives the fraction of non-selected entries of a bootstrapped replicate given the size of the data set. How many unique observations are in a bootstrapped sample when $N \to \infty$?

3

g) When datasets are subject to clusters and groups, it is often necessary to ensure that these subgroups appear with same frequency. Consider for instance the example where we want to estimate the average height of persons.

Our original dataset might be imbalanced, for instance, containing more observations of males than of females. However, we want to estimate the average height independent of gender. Hence, resampling needs to pick male and female observations with same likelihood, no matter what their true frequency in the dataset is.

The above process is named *stratified resampling*.

In this question, load the following dataset and estimate the mean height. How does the results change when incorporating stratified resampling?

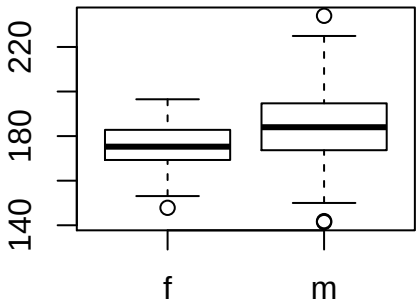
```
set.seed(0)

n_male   <- 630
n_female  <- 170
d <- data.frame(height=c(rnorm(n_male, 185, 15), rnorm(n_female, 175, 10)),
                gender=factor(rep(c("m", "f"), c(n_male, n_female))))

mean(d$height)

## [1] 182.7314

boxplot(height ~ gender, d)
```



3

h) Modify your code from the previous exercise. Use now the `bayesboot` package to calculate the mean of each group instead. T

Furthermore, use the `as.bayesboot` function to calculate the posterior of the differences to receive a `bayesboot` object, which can be plotted using the `plot` function.

3

i) We now use three small datasets which we resample to infer the distribution of the mean:

```r
set.seed(0)

height2 <- rnorm(2, 180, 10)
height4 <- rnorm(4, 180, 10)
height8 <- rnorm(8, 180, 10)
```

Visualize the distributions as histograms using the non-parametric and the Bayesian bootstrap. How do they differ?

Lastly, find and read materials that explain the Bayesian bootstrap. How is it different from the frequentest variant of the Bayesian that we used before? For instance, study the following blog post: `http://sumsar.net/blog/2015/04/the-non-parametric-bootstrap-as-a-bayesian-model/`.

3

j) Proof the mathematical solution for the risk minimization of a portfolio. That is

$$\alpha = \frac{\sigma_Y^2 - \sigma_{XY}}{\sigma_X^2 + \sigma_Y^2 - 2\sigma_{XY}},$$

which minimizes the expression $\text{Var}(\alpha X + (1-\alpha)Y)$.

3

k) When the dataset is small, one sometimes utilizes a two-staged cross-validation. Implement this process for the `Carseats` dataset from the `caret` package in order to predict the sales. Use $k = 10$ splits and a linear model.

For simplicity, we remove the variables that are factors:

```r
data(Carseats)

x <- subset(Carseats, select=-c(Sales, ShelveLoc, Urban, US))
y <- Carseats$Sales
```