SDA 2023 — Assignment 3

For these exercises you can use the functions `CV` and `h_opt` (in the file "functions_Ch4.txt") and the function `bootstrap` (in the file "functions_Ch5.txt") on the Canvas page. Investigate these functions before you use them. Anyhow, here is some preliminary information on the functions `h_opt` and `CV`:

`h_opt` uses formula (4.3) in the syllabus; it is based on the sample standard deviation as an estimator of the standard deviation. Also, the normal location scale family has been used to find a hopefully reasonable value for the involved integral $\int (f'')^2(t)dt$.

Note: for any location scale family $\{G_{\mu,\sigma} : \mu \in \mathbb{R}, \sigma > 0\}$ w.r.t. the distribution function G with density g, we have $g_{\mu,\sigma}(t) = \frac{d}{dt}G_{\mu,\sigma}(t) = \frac{d}{dt}G(\frac{t-\mu}{\sigma}) = g(\frac{t-\mu}{\sigma})/\sigma$. Thus, one can show that $\int (g''_{\mu,\sigma})^2(t)dt = \int (g'')^2(t)dt/\sigma^5$. The following table is a list of values of $\int (g''_{\mu,\sigma})^2(t)dt$ for different unimodal densities g with variance 1. You can used these if you wish to replace the default choice for the normal distribution.

density	$g(t)$	$\int (g'')^2(t)dt$
standard normal	$(2\pi)^{-1/2}\exp(-t^2/2)$	$\frac{3}{8}\pi^{-1/2}$
logistic (with scale parameter $s = \sqrt{3}/\pi$)	$\frac{\exp(-t/s)}{s(1+\exp(-t/s))^2}$	$\pi^5 \frac{13}{3^{7/2}\cdot 35}$
Double exponential (with scale parameter $b = 1/\sqrt{2}$)	$(2b)^{-1}\exp(-\frac{\lvert t-\mu\rvert}{b})$	$\sqrt{2}$
exponential (with rate parameter $\lambda = 1$)	$\lambda\exp(-\lambda t)1\{t > 0\}$	0.5

The function `CV` computes for a given bandwidth, sample, and kernel the cross-validation criterion $\hat{R}(\hat{f})$. Thus, in order to find the minimizing bandwidth value, you should apply `CV` to multiple bandwidths and then select the one that led to the smallest value of $\hat{R}(\hat{f})$. In R this can be achieved via, e.g.

```
cv_crit <- sapply(h_vec, CV, sample=sample, kernel="gauss")
h_min <- h_vec[which(cv_crit == min(cv_crit))]
```

where `h_vec` is a vector of bandwidths for each of which the cross-validation criterion shall be computed.

Kernel density estimators in R can be obtained by using the function `density`. Use the `help`-function to find out how to use this function.

Make a concise report of *all* your answers in *one single PDF file*, with only *relevant R* code *in an appendix*. It is important to make clear in your answers how you have solved the questions. Graphs should look neat (label the axes, give titles, use correct dimensions etc.). Multiple graphs can be put into one figure using the command `par(mfrow=c(k,l))`, see `help(par)`. Sometimes there might be additional information on what exactly has to be handed in.
Read the file AssignmentFormat.pdf on Canvas carefully.

Exercise 3.1 (Class) The file `sample31.txt` contains a sample of $n = 200$ observations. Try different values of the bandwidth h to see the influence of h on the kernel density estimator for the density underlying the sample.

Exercise 3.2 The file `sample32.txt` contains a sample of $n = 150$ observations. Pick a kernel function and a bandwidth, and use a kernel density estimator to estimate the density based on the sample.
Note: don't just use trial and error but proceed systematically, i.e. motivate your choices.

Exercise 3.3 The file `sample33.txt` contains a sample of $n = 100$ *positive* observations. Find a suitable kernel density estimate based on this sample.

Hint: it seems appropriate to assign no mass to $\hat{f}(x)$ for $x < 0$. Take a look at Lecture 4 to find out how this can be achieved in a reasonable way. Use just one of the available approaches.

Hint: if you would like to use the log*-transformation and first find a suitable kernel density estimate $\hat{f}_y$ based on the* log*-transformed sample* **y***, i.e. $y_1 = \log(x_1), ..., y_n = \log(x_n)$, you can obtain the density estimate $\hat{f}_x$ for the original sample based on*

```
yrange <- seq(min(y), max(y), length.out=512)
lines(exp(yrange), density(y, ..., from=min(yrange) , to=max(yrange))$y/exp(yrange))
```

This is due to $F_y(t) = F_x(\exp(t))$ for the cumulative distribution functions of the y- and x-samples, respectively. Thus, $f_y(t) = f_x(\exp(t)) \cdot \exp(t)$ for their densities.

Exercise 3.4 The file `list34.RData` contains a list `list34` with the following entries:

- a vector `sample34` which is a sample of $n = 120$ numbers;
- a vector `true.density.x` which is a sample of size 512 with the x-values for the true density;
- a vector `true.density.y` which is a sample of size 512 with the corresponding y-values of the true density.

Find two kernel density estimates based on `sample34`: for the first, use the bandwidth obtained from the function `h_opt`, for the second, use the bandwidth obtained from the cross-validation criterion. Compare these estimates to the true density function. Compare the density estimate with the true density and argue which kernel density estimate seems preferable.

Note: in reality, one of course doesn't know the true density. But in this case, the "true" density was found quite accurately by simulation and it can therefore be considered "known".

Hints: first explore the functions `h_opt` & `CV`.
Search for the minimizer of the cross-validation criterion on the interval $[0.0005, 0.008]$.

Hand in for Exercises 3.2–3.4: answers to the questions, your estimation strategy, and relevant plots that helped you to find the final kernel density estimates. Motivate your choices of kernel functions, bandwidths, and transformations, if any are used.
General information on the .RData file (for Exercise 3.5) to be submitted to the dummy assignment "RData A3" on Canvas:
Create in R a list `mylist` that contains the required entries as specified in the exercise(s) which are marked as "(partially) .RData file hand-in". You should store your list in an .RData file by using the R command `save(mylist, file="[PATH]/myfile3_[GROUP_NO].RData")`, where `[PATH]` stands for the path on your computer where you wish to save the .RData file and `[GROUP_NO]` stands for the number of the assignment group you have chosen on Canvas). For example, for if you're in group 91, use `"[PATH]/myfile3_91.RData"`.
Exercise 3.5 requires you to set a seed depending on your group number: `20230303 + [GROUP_NO]`. For instance, for if you're in group 91, use the seed 20230303+91 = 20230394.
In any case, `mylist` must have the entry `stud_no`: a vector that contains the student numbers of you and your group partner.

The functions `bootstrap` (in `functions_Ch5.txt`), `rt, ecdf(x)(y)` could be useful for the following exercise, where `x` stands for a sample and `y` stands for an evaluation argument.

Exercise 3.5 (partially .RData file hand-in) One sample drawn from a t-distribution with unknown degrees of freedom $k > 0$ is stored in the file `t-sample.txt`. With the help of this sample, we would like to estimate the distribution of the statistic $T = \hat{F}(1)$, which is the empirical cumulative distribution function evaluated at 1.

a. Compute $\hat{F}(1)$ based on the given t-sample.

b. Set the seed to `20230303 + [GROUP_NO]`. Then use the empirical bootstrap method applied to the t-sample to generate $B = 2000$ bootstrap estimates of the statistic T. Store these in a vector `ecdf1_empBS` in your R environment.
Tip: in combination with the bootstrap, it could be useful to use the following function: `function(x) ecdf(x)(1)`.

c. Repeat the steps of part b. (including setting the seed once again) but with the parametric bootstrap instead of the empirical bootstrap. Use $\hat{k} = 2s^2/(s^2 - 1)$ as an estimator of the degrees of freedom k, where s^2 denotes the sample variance. Denote the vector which contains the obtained parametrically bootstrapped statistics by `ecdf1_parBS`.

d. Plot two separate histograms of the bootstrap samples obtained in b. and c.
Compare them to another histogram for the true distribution of T. One can obtain this in the following way:
Set the seed to `20230303 + [GROUP_NO]`. Then generate 2000 independent samples of size 50 from the t-distribution with 5 degrees of freedom (which is the true underlying distribution by the way!), and compute the value of T for each sample; then store those 2000 realizations of T in the vector `ecdf1_realizations` and plot their histogram. Next to this, also plot the histograms of `ecdf1_empBS` and `ecdf1_parBS`, and compare them with the histogram of `ecdf1_realizations`.
Based on these comparisons, which bootstrap method seems preferable in the present context? Motivate your answer.

e. Use the empirical and the parametric bootstrap samples to find estimates of the standard deviation of T. Compare these two estimates with an approximation of the true standard deviationof that statistic, which could be obtained as the sample standard deviation of the realizations from d.

<u>**Hand in:**</u>
For the the main report: from part d.: relevant plots, descriptions, and motivated answers.
Stored in your .RData file: from parts a.-e. the following entries of your list `mylist` in R:

a.: `ecdf1_sample`: the value of T based on the t-sample,

b.: `ecdf1_empBS`: the vector of 2000 empirically bootstrapped statistic T,

c.: `ecdf1_parBS`: the vector of 2000 parametrically bootstrapped statistic T,

d.: `ecdf1_realizations`: the vector of 2000 realized values of T,

e.: `sd_empBS`: the standard deviation estimate based on `ecdf1_empBS`,
`sd_parBS`: the standard deviation estimate based on `ecdf1_parBS`,
`sd_realizations`: the standard deviation estimate based on `ecdf1_realizations`.