Functional data analysis: Descriptive tools

Write a report that contains the results of the computations that you are asked to carry out below, as well as the explanation of what you are doing. The main text (2 or 3 pages) should include pieces of source code and graphical and numerical output.
Upload your answers in a .pdf document (use LaTeX or R Markdown, for instance) to ATENEA, as well as the source code (*.R or *.Rmd, for instance). Your work must be reproducible.

Bike sharing data

In Bike Sharing Dataset, Washington D.C., USA (UCI Machine Learning Repository) you can find information on the bike-sharing rental service in Washington D.C., USA, corresponding to years 2011 and 2012. There are three files:

- `Readme.txt`: See there the definition of the variables in the other two files.
- `hour.csv`: Bike sharing counts aggregated on hourly basis. Records: 17379 hours
- `day.csv`: Bike sharing counts aggregated on daily basis. Records: 731 days

1. Transforming `cnt` data into functional data

1. Transform the 17379 values of `cnt` in `hour.csv` into a 731×24 matrix X with `NA` at hours with no data in `hour.csv`.
2. Impute these `NA` values with a 0. Do you think this imputation has sense? Call still X to the resulting matrix
3. Transform the previous data matrix X (with 0s instead `NA`s) into a `fd` object (library `fda`) and call it `cnt.fd`.
4. Transform the previous data matrix X (with 0s instead `NA`s) into a `fdata` object (library `fda.usc`) without any smoothing and call it `cnt.fdata`. We also refer to this functional data set as *the raw data*.
5. Assume that, for any day i, $\texttt{cnt}_i(t) \sim \text{Poisson}(\lambda_i(t))$, $t \in [0, 24]$. For each day in the bike sharing data set, use `mgcv::gam` to estimate the corresponding intensity function $\lambda_i(t)$ (see the `medfly25` example in the R script `SAT12_medfly_smooth.R`). Then transform the 731 estimated intensity functions into a functional data set (of class `fdata`) called `lambda.fdata`.
6. Transform the logarithms of the 731 estimated intensity functions into a functional data set (of class `fdata`) called `log.lambda.fdata`.

2. Descriptive tools in library `fda`

1. Estimate the mean function and the standard deviation function for `cnt.fd`. Then do it again differentiating by working/non-working days.

2. Estimate the correlation function for `cnt.fd`. Then do it again differentiating by working/non-working days.

3. Transform `log.lambda.fdata` into an `fd` object (call it `log.lambda.fd`) and repeat the previous two points for `log.lambda.fd`.

3. Descriptive tools in library `fda.usc`

1. Estimate the mean function for `cnt.fdata`, `lambda.fdata` and `log.lambda.fdata`. Then do it again differentiating by working/non-working days.

2. Compute the median and the trimmed mean based on Fraiman and Muniz depth for `cnt.fdata`, `lambda.fdata` and `log.lambda.fdata`. Compare the results with the mean functions.

3. Detect outliers in `cnt.fdata`, `lambda.fdata` and `log.lambda.fdata`.