Practical exercise 2 : estimation in partially linear models

Objective: Write a piece of R code implementing an estimation method in partially linear models, described in Chapter 2. You can illustrate its calculation using the Warsaw real estate data (this is obtained from `data(apartments)` after having loaded package `PBImisc`), or one of the real data examples in the package `np`, or a data set that you have simulated yourself.

Level 1: implement the least squares estimator $\widehat{\boldsymbol{\beta}}$ of the parametric component with a given kernel (*e.g.* naive) and a given bandwidth h.

Level 2: write a function, whose arguments should be a sample of data, a vector $\boldsymbol{b}$ and a point z, which returns the kernel estimator of the regression function $\mathbb{E}(Y - \boldsymbol{X}^\top \boldsymbol{b} | Z = z)$ at the point z, with this same kernel and bandwidth.

Level 3: combine your codes in Levels 1 and 2 to write a function, whose arguments should be a sample of data and a point z, which returns the kernel estimator of the regression function $g(z) = \mathbb{E}(Y - \boldsymbol{X}^\top \boldsymbol{\beta} | Z = z)$ at the point z, with this same kernel and bandwidth.

Level 4: upgrade your code in Level 3, first by including as arguments in this function the kernel function (within a reasonable list of kernels) and the bandwidth h, and then by allowing a second, different choice of bandwidth for the estimation of the regression function g by cross-validation.

Level 5: speed up your code in Level 4.

Compare your results with those of existing packages. Do you see any problem with the implementation of the estimator? (instability, unreasonably large computation time...) Can you think of and implement a cross-validation method that will yield an optimal choice of h both for the estimation of the parametric component $\boldsymbol{\beta}$ and the regression function g?