simulation_week7

Tingyu Zhu

11/12/2020

```
library(tidyverse)
library(here)
library(ggplot2)
library(purrr)
devtools::load_all()
set.seed(1222)
```

Inspection about the bandwidth. Compare the performance under different methods of selecting the bandwidth.

Simulation over different n & h & kernel

`sim_big_made` is used to simulate over different combinations of sample sizes, kernels and bandwidth. The out out is the MADEs under different combination of parameters. In this simulation, the bandwidths are randomly selected.

Generating simulation data.

```
# samples
x <- rt(5000, df=5)

# simulation parameters
ns <- c(100, 500)
kers  <- c("normal","epanech", "uniform", "biweight", "triweight")
hs <- data.frame(0.1,0.5,1)

grid <- seq(-5,15, 0.1)
# true densities
true_f <- dt(grid, df=5)
# simulation
n.sim <- 50
sim_rlt <- map(1:n.sim, ~sim_big_made(x, ns, kers, hs, grid,true_f))
```

Save results

`save_made` is used to:

a) save the raw data of `n.sim` time of replicas as "made_mat_big";

b) calculate the means and standard deviations of the `n.sim` time of replicas under different combination of parameters;

c) save the means and standard deviations as “made_comapre” for analysis and drawing plots.

```
r <- save_made(sim_rlt,ns,kers,hs,n.sim )

write_rds(r$made_mat_big , here("results", "week7-sim-1.rds"))
write_rds(r$made_comapre, here("results", "week7-sim-1-plot.rds"))
```

Types of bandwidth selectors

1. Normal reference bandwidth selector

The normal reference bandwidth selector is defined by:

$$\hat{h}_{opt} = \begin{cases} 1.06sn^{-1/5} \text{ for the Gaussian kernel} \\ 2.34sn^{-1/5} \text{ for the Epanechnikov kernel} \end{cases}$$

The normal reference bandwidth selector is only a simple rule of thumb. It is a good selector when the data are nearly Gaussian distributed. However, it can lead to over-smooth when the underlying distribution is asymmetric or multi-modal.

2. plug-in bandwidth selector

There are quite a few important techniques for selecting the bandwidth without the restrictions to the Gaussian. such as cross-validation (CV) and plug-in bandwidth selectors. Function `dpik()` in the package “KernSmooth” in R selects a bandwidth for estimating the kernel density estimation using the plug-in method.

```
library("KernSmooth")
```

```
## Warning: package 'KernSmooth' was built under R version 4.0.2
```

```
## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009
```

```
# dpik(x,kernel = "normal")
# dpik(x,kernel = "epanech")
```

Simulation

KDE with two types of kernels and three difference bandwidths. Here, the kernel functions are Gaussian kernel and the Epanech kernel. And the bandwith are chosed using the Normal reference bandwidth selector (labeled as “opt”), and the plug-in bandwidth selector (labeled as “plug”), under the corresponding kernels, and a randomly chosed bandwidth as the baseline (labled as “ran”).

We simulate with samples from three types of ditributions: t(5), exp(2), and gamma(2,0.5). We repeat the three types of distributions by changing two stpes: a) generating the samples using rt(),rexp(), and rgamma(); 2) corresponding true densities dt(),dexp(),and dgamma(). Then, save the results respectively.

```r
# samples from different distribution
nx <- 5000
x <- rt(nx, df=5)
#x <- rexp(nx, 2)
#x <- rgamma(nx, 2,0.5)

# simulation parameters
ns <- c(4000) # fix the sample size
kers  <- c("normal","epanech")
## different chioces of bandwidths
h_opt_Gau <- 1.06*sd(x)*nx^(-0.2)
h_opt_Epa <- 2.34*sd(x)*nx^(-0.2)# Normal reference bandwidth selector

h_plug_Gau <- dpik(x,kernel = "normal")
h_plug_Epa <- dpik(x,kernel = "epanech") # plug-in bandwidth selector

h_ran <- 0.5*nx^(-0.2) # random bandwith as the baseline

hs <- data.frame(h_opt_Gau,h_plug_Gau,h_opt_Epa, h_plug_Epa,h_ran)
(hs <- sort(hs))
```

```
##        h_ran h_plug_Gau h_opt_Gau h_plug_Epa h_opt_Epa
## 1 0.09102821  0.1937284 0.2479594  0.4288768 0.5473821
```

```r
grid <- seq(-5,15, 0.1)
# true densities
true_f <- dt(grid, df=5)
#true_f <- dexp(grid,2)
#true_f <- dgamma(grid, 2,0.5)

# simulation
n.sim <- 50
sim_rlt_bdwt <- map(1:n.sim, ~sim_big_made(x, ns, kers, hs, grid,true_f))
```

Save results

```r
r2 <- save_made(sim_rlt_bdwt,ns,kers,hs,n.sim )

write_rds(r2$made_mat_big , here("results", "week78-sim-t.rds"))
#write_rds(r2$made_mat_big , here("results", "week78-sim-exp.rds"))
#write_rds(r2$made_mat_big , here("results", "week78-sim-gamma.rds"))

write_rds(r2$made_comapre, here("results", "week78-sim-t-plot.rds"))
#write_rds(r2$made_comapre, here("results", "week78-sim-exp-plot.rds"))
#write_rds(r2$made_comapre, here("results", "week78-sim-gamma-plot.rds"))
```

Analysis

Results and plots are documented in `Summary_week7` in the `analysis` file.