Package 'monfuncreg'

March 15, 2018

Type Package

Title Monotone Nonparametric Regression for Functional/Longitudinal Data

Version 1.0

Date 2018-03-14

Author Ziqi Chen, Hongtu Zhu

Maintainer Ziqi Chen <chenzq453@gmail.com>

Description
Monotone Nonparametric Regression for Mean Function in Functional/Longitudinal Models.

License GPL (>= 2)

LinkingTo Rcpp, RcppArmadillo

R topics documented:

`monfuncreg` *Monotone Nonparametric Regression for Functional/Longitudinal Data*

Description

monfuncreg provides a increasing monotone estimator of the mean regression function in functional/longitudinal models.

Usage

```
monfuncreg(x, y, NN, N, hr, hd, weight="OBS", t)
```

Arguments

x	vector containing the x-values (design points) of a sample, scaled x to [0,1] $x = (x_{11}, ..., x_{1m_1}, ..., x_{1m_n})$.
y	vector containing the y-values (response) of a sample, $y = (y_{11}, ..., y_{1m_1}, ..., y_{1m_n})$.
NN	the number of observations for each subject, $NN = (m_1, ..., m_n)$.
N	the number of evaluation points of the unconstrained nonparametric regression estimator.
hr	bandwith of kernel K_r of the regression estimation step.
hd	bandwith of kernel K_d of the density estimation step.
weight	"OBS" or "SUBJ".
t	vector of points where the monotone estimation is computed, which is on [0,1].

Details

Monotone Nonparametric Regression for Functional/Longitudinal Data, by Chen, Zhu, Gao and Fu, 2018

Value

`monfuncreg` returns a list of values

mon1$value	the points, for which the monotone function values will be estimated
mon1$estimate	the monotone estimate at mon1$value

Author(s)

Ziqi Chen, Hongtu Zhu

Maintainer: Ziqi Chen <chenzq453@gmail.com>

References

Monotone Nonparametric Regression for Functional/Longitudinal Data, by Chen, Zhu, Gao and Fu, 2018

Examples

```
library(monfuncreg)
shuju=read.csv("Origdata1.csv",head=F,quote="")
YY=list()
TT=list()
jishu=0
for(i in 1:length(shuju[,1])){
  if((shuju[i,1]==1)&(is.na(shuju[i,13])==FALSE)){
    jishu1=1
    jishu=jishu+1
    YY[[jishu]]=numeric(0)
    TT[[jishu]]=numeric(0)
    YY[[jishu]][jishu1]=shuju[i,13]
    TT[[jishu]][jishu1]=shuju[i,5]
  }
  if(i>1){ if((shuju[i,1]>shuju[i-1,1])&(is.na(shuju[i,13])==FALSE)){
```

```
    jishu1=jishu1+1
    YY[[jishu]][jishu1]=shuju[i,13]
    TT[[jishu]][jishu1]=shuju[i,5]
  }

  }

}

NN=numeric(0)
for(i in 1:562){
  NN[i]=length(YY[[i]])
}

TT2=numeric(sum(NN))
YY2=numeric(sum(NN))

shu1=0
for(i in 1:562){
  TT2[(shu1+1):(shu1+NN[i])]=TT[[i]]
  YY2[(shu1+1):(shu1+NN[i])]=YY[[i]]
  shu1=shu1+NN[i]
}

TT1=(TT2-min(TT2))/(max(TT2)-min(TT2))
YY1=-(YY2-mean(YY2))/sd(YY2)
n=length(NN)

quant1=quantile(TT1,probs=seq(0,1,0.25))
hr=1.06*(n*mean(NN))^(-1/5)*min(sd(TT1),(quant1[4]-quant1[2])/1.34)
quant1=quantile(YY1,probs=seq(0,1,0.25))
hd=1.06*(n*mean(NN))^(-0.3)*min(sd(YY1),(quant1[4]-quant1[2])/1.34)
t1=seq(0.01,0.99,by=0.001)
N=1000

weight="OBS"
jieguo1=monfuncreg(TT1, YY1, NN,  hr, hd, N, weight,t1)
weight="SUBJ"
jieguo2=monfuncreg(TT1, YY1, NN,  hr, hd, N, weight,t1)

matplot(jieguo1$variable,cbind(-jieguo1$estimate,-jieguo2$estimate),type="l",
lty=1,lwd=2,xlab="Scaled Age", ylab="Standardized Volume of Grey Matter")
```

Index