Package ‘npregderiv’

July 8, 2020

Title Nonparametric Estimation of the Derivatives of a Regression Function

Version 1.0

Depends R (>= 3.1.0)

Description Estimating the first and second derivatives of a regression function by the method of Wang and Lin (2015) <http://www.jmlr.org/papers/v16/wang15b.html>.

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

NeedsCompilation no

Author Olga Savchuk [aut, cre],
Alex Volinsky [aut]

Maintainer Olga Savchuk <olga.y.savchuk@gmail.com>

Repository CRAN

Date/Publication 2020-07-08 08:30:10 UTC

R topics documented:

`npregderiv package`	*The package "Nonparametric Estimation of the Derivatives of a Regression Function" implements the method of Wang and Lin (2015) of estimating the first and second derivatives of a regression function from the original data in the case of an increasing evenly spaced design. The derivative estimates are computed at the design points. The computations are based on the difference quotients. The package also includes the experimental data set on the deflection measurements of a round substrate before and after the thin film deposition. See more details regarding the experiment and data analysis in the article of Savchuk and Volinsky (2020).*

Description

Estimating the first and second derivatives by the method of Wang and Lin (2015) requires choosing the soothing parameter k that shows how many observations from each side of an estimation point are used. Notice that the method of Wang and Lin (2015) requires an estimation point to be one of the design points. If a single value of k is used for all estimation points, it is impossible to compute a derivative estimate for the first and the last k data points that constitute the left and right boundary regions, respectively. Generally, an adaptive value of k may be used (see Wang and Lin (2015)). Even though the article of Wang and Lin (2015) contains the expressions for the asymptotically optimal k, those results are of no practical importance since they involve the higher order derivatives of the regression function. Wang and Lin (2015) implement their method for the values for k ranging from $0.02n$ to $0.2n$, where n is the sample size. The larger the value of k, the smoother the resulting estimate. However, too large k may lead to smoothing away certain important features of the function's derivatives. We found that the value of $k = 0.1n$ worked reasonably well for estimating the first and second derivatives of the regression functions in many practical cases.

Details

The function `reg_1derivWL` is developed to estimate the first derivative of the underlying regression function from the original data. The function `reg_2derivWL` is intended to estimate the function's second derivative. The latter function was used for estimating the second derivatives of the deflection functions in the experiment described and analyzed in the article of Savchuk and Volinsky (2020). The deflection's second derivative is used to estimate the substrate's curvature and, ultimately, the residual stress of the thin film deposited on the substrate.

The data set `wafer40` analyzed in the article of Savchuk and Volinsky (2020) is included into the package. It contains the deflection measurements for one of the wafers (wafer 40) used in the experiment. The measurements are taken before and after the thin film deposition in two radial directions on the round substrate and are marked as "0 degrees" and "90 degrees" depending on the scan orientation. The design points are separated by 0.000005 m. The resulting sample size in each case is n=7755. All deflection and position measurements are recorded in meters.

References

Wang, W.W., Lin, L. (2015) http://www.jmlr.org/papers/v16/wang15b.html "Derivative Estimation Based on Difference Sequence via Locally Weighted Least Squares Regression", *Journal of Machine Learning Research*, 16, 2617-2641.

Savchuk, O., Volinsky, A.A. (2020). "Nonparametric estimation of SiC film residual stress from the wafer profile".

`reg_1derivWL` *Estimating the first derivative of a regression function by the method of Wang and Lin (2015)*

Description

The design data $xdat$ must be increasing and evenly spaced. The first derivative estimate is computed at u that must be one of the design points $xdat$ from the interior region. The interior region excludes the first k points with the smallest $xdat$ values (left boundary) and the last k points with the largest $xdat$ values (right boundary).

Usage

```
reg_1derivWL(xdat, ydat, k, u)
```

Arguments

`xdat`	numerical vector of the increasing and evenly spaced design points
`ydat`	numerical vector of the corresponding data points
`k`	integer value of the smoothing parameter
`u`	an estimation point (scalar) that must be one of the $xdat$ points from the interior region

Details

The method's smoothing parameter k shows how many data points are used from each side of an estimation point u to compute the first derivative estimate at that point. Choose $k < n/4$, where n is the sample size. The value of $k = 0.1n$ can be taken as a starting point.

Value

The computed first derivative estimate (scalar) at the point u.

References

Wang, W.W., Lin, L. (2015) http://www.jmlr.org/papers/v16/wang15b.html "Derivative Estimation Based on Difference Sequence via Locally Weighted Least Squares Regression", *Journal of Machine Learning Research*, 16, 2617-2641.

See Also

reg_2derivWL

Examples

```
# EXAMPLE 1: Toy example
xdata=1:100
ydata=xdata^2
reg_1derivWL(xdata,ydata,10,30)

# EXAMPLE 2 (simulated data).
m=function(x)
  sin(2*pi*x)+log(6+5*x)
m1=function(x)                                            # first derivative of m
  2*pi*cos(2*pi*x)+5/(6+5*x)
N=300                        # sample size
xi=2*(1:N/N)-1               # equidistant design points
sigma=0.25                   # noise level used in the article
yi=m(xi)+rnorm(N,sd=sigma)   # generating data
K=0.1*N                      # selected value of the smoothing parameter k
x_inter=xi[(K+1):(N-K)]      # interior estimation region
N_inter=length(x_inter)    # number of points where the second derivative estimate is computed
M1=array(0,N_inter)         # Vector of estimated second derivatives at the points x_inter
for (i in 1:N_inter)
  M1[i]=reg_1derivWL(xi,yi,K,x_inter[i])
op=par(no.readonly=TRUE)
par(mfrow=c(1,2))
plot(xi,yi,pch=20,cex=1.7,main="Regression function and generated data",xlab="",ylab="",
     cex.axis=1.8,cex.main=1.5)
lines(xi,m(xi),'l',col="red",lwd=2)
title(xlab="argument",ylab="regression function",line=2.5,cex.lab=1.8)
legend(0.65,-0.4,lty="solid",col="red",lwd=2,legend="true regression function",bty="n",
       cex=1.45,seg.len=0.5)
plot(x_inter,M1,'l',lwd=2,main="Actual and estimated first derivative",xlab="",ylab="",
     cex.axis=1.8,cex.main=1.5)
lines(x_inter,m1(x_inter),'l',lwd=2,col="red")
title(xlab="argument",ylab="second derivative",line=2.5,cex.lab=1.8)
legend(0.6,-10,lty=c("solid","solid"),col=c("red","black"),lwd=c(2,2),
      legend=c("true first derivative", "estimate"),bty="n",cex=1.45,seg.len=0.5)
par(op)


# EXAMPLE 3: Estimating the first derivative of the deflection function for the data on
# deflection after film deposition scanned at 0 degrees.
# See Savchuk O., and Volinsky, A. (2020) for the experiment's description.
xdesign=wafer40$x_dat
ydata=wafer40$y_after_0
n_data=length(xdesign)                        # sample size (original)
n_new=1034                               # reducing the number of points where the second
                                              # derivative is estimated.
K=ceiling(0.1*n_data)                         # value of the smoothing parameter
index=(K+1)+0:(n_new-1)*6                 # cutting about 10% of points from each side and
```

```
                                            # reducing the number of points where the second
                                                 # derivative is estimated
    x_inter=xdesign[index]                       # the values of argument where the second
                                                 # derivative is to be estimated
    y_inter=ydata[index]
    Der1=array(0,n_new)
    for (i in 1:n_new)
      Der1[i]=reg_1derivWL(xdesign,ydata,K,x_inter[i])
    op=par(no.readonly=TRUE)
    par(mfrow=c(1,2))
    plot(xdesign*1000,ydata*10^6,pch=20,main="Deflection AFTER film deposition. Angle: 0 degrees.",
         xlab="",ylab="",cex.axis=1.8,cex.main=1.5)
    title(ylab=expression(paste("deflection, ",mu,"m")),xlab="position, mm",line=2.25,cex.lab=1.8)
    plot(x_inter*1000,Der1,'l',lwd=2,
         main="1-st derivative AFTER film deposition. Angle: 0 degrees.",
         xlab="",ylab="",cex.axis=1.8, cex.main=1.5)
    title(ylab="first derivative of deflection", xlab="position, mm", line=2.5, cex.lab=1.8)
    par(op)
```

`reg_2derivWL` *Estimating the second derivative of a regression function by the method of Wang and Lin (2015)*

Description

The design data $xdat$ must be increasing and evenly spaced. The second derivative estimate is computed at u that must be one of the design points $xdat$ from the interior region. The interior region excludes the first k points with the smallest $xdat$ values (left boundary) and the last k points with the largest $xdat$ values (right boundary).

Usage

```
reg_2derivWL(xdat, ydat, k, u)
```

Arguments

xdat	numerical vector of the increasing and evenly spaced design points
ydat	numerical vector of the corresponding data points
k	integer value of the smoothing parameter
u	an estimation point (scalar) that must be one of the $xdat$ points from the interior region

Details

The method's smoothing parameter k shows how many data points are used from each side of an estimation point u to compute the second derivative estimate at that point. Choose $k < n/4$, where n is the sample size. The value of $k = 0.1n$ can be taken as a starting point.

Value

The computed second derivative estimate (scalar) at the point u.

References

Wang, W.W., Lin, L. (2015) http://www.jmlr.org/papers/v16/wang15b.html "Derivative Estimation Based on Difference Sequence via Locally Weighted Least Squares Regression", *Journal of Machine Learning Research*, 16, 2617-2641.

Savchuk, O., Volinsky, A.A. (2020). "Nonparametric estimation of SiC film residual stress from the wafer profile".

See Also

reg_1derivWL, wafer40

Examples

```
# EXAMPLE 1 (simulated data).
# The regression function is taken from p. 2631 of the article of Wang and Lin (2015).
m=function(x)              # Regression function from the article of Wang and Lin (2015)
  32*exp(-8*(1-2*x)^2)*(1-2*x)
m2=function(x)              # second derivative of m
  -2048*exp(-8*(1-2*x)^2)*(128*x^3-192*x^2+90*x-13)
N=100                       # sample size
xi=1:N/N                    # equidistant design points
sigma=0.2                   # noise level used in the article
yi=m(xi)+rnorm(N,sd=sigma)  # generating data
K=0.1*N                     # selected value of the smoothing parameter k
x_inter=xi[(K+1):(N-K)]     # interior estimation region
N_inter=length(x_inter)    # number of points where the second derivative estimate is computed
M2=array(0,N_inter)        # Vector of estimated second derivatives at the points x_inter
for (i in 1:N_inter)
  M2[i]=reg_2derivWL(xi,yi,K,x_inter[i])
op=par(no.readonly=TRUE)
par(mfrow=c(1,2))
plot(xi,yi,pch=20,cex=1.7,main="Regression function and generated data",xlab="",ylab="",
     cex.axis=1.8,cex.main=1.5)
lines(xi,m(xi),'l',col="red",lwd=2)
title(xlab="argument",ylab="regression function",line=2.5,cex.lab=1.8)
legend(0.35,5,lty="solid",col="red",lwd=2,legend="true regression function",bty="n",
       cex=1.45,seg.len=0.5)
plot(x_inter,M2,'l',lwd=2,main="Actual and estimated second derivative",xlab="",ylab="",
     cex.axis=1.8,cex.main=1.5)
lines(x_inter,m2(x_inter),'l',lwd=2,col="red")
title(xlab="argument",ylab="second derivative",line=2.5,cex.lab=1.8)
legend(0.01,800,lty=c("solid","solid"),col=c("red","black"),lwd=c(2,2),
      legend=c("true second derivative", "estimate"),bty="n",cex=1.45,seg.len=0.5)
par(op)


# EXAMPLE 2: Estimating the second derivative of the deflection function for the data on
```

```
# deflection after film deposition scanned at 90 degrees.
# See Savchuk O., and Volinsky, A. (2020) for the experiment's description.
xdesign=wafer40$x_dat
ydata=wafer40$y_after_90
n_data=length(xdesign)      # sample size (original)
n_new=1034                  # reducing the number of points where the second
                            # derivative is estimated.
K=ceiling(0.1*n_data)       # value of the smoothing parameter
index=(K+1)+0:(n_new-1)*6   # cutting about 10% of points from each side and reducing the
                            # number of points where the second derivative is estimated
x_inter=xdesign[index]     # the values of argument where the second derivative is to be
                            # estimated
y_inter=ydata[index]
Der2=array(0,n_new)
for (i in 1:n_new)
  Der2[i]=reg_2derivWL(xdesign,ydata,K,x_inter[i])
op=par(no.readonly=TRUE)
par(mfrow=c(1,2))
plot(xdesign*1000,ydata*10^6,pch=20,main="Deflection AFTER film deposition. Angle: 90 degrees.",
     xlab="",ylab="",cex.axis=1.8,cex.main=1.5)
title(ylab=expression(paste("deflection, ", mu, "m")), xlab="position, mm", line=2.25,
      cex.lab=1.8)
plot(x_inter*1000,Der2,'l',lwd=2,
     main="2-nd derivative AFTER film deposition. Angle: 90 degrees.",xlab="",ylab="",
     cex.axis=1.8, cex.main=1.5)
title(ylab="second derivative of deflection, 1/m", xlab="position, mm", line=2.5,
      cex.lab=1.8)
par(op)
```

`wafer40` *The data set on the n=7755 substrate's deflection measurements before and after the thin film deposition in two radial directions.*

Description

The data frame on the deflection measurements is described and analyzed in the article of Savchuk and Volinsky (2020).

Usage

```
wafer40
```

Format

The data frame has five variables:

x_dat design points (separated by 0.000005 m);

y_before_0 deflections measurements (in m) before film deposition at 0 degrees;

y_after_0 deflections measurements (in m) after film deposition at 0 degrees;

y_before_90 deflections measurements (in m) before film deposition at 90 degrees;

y_after_90 deflections measurements (in m) after film deposition at 90 degrees.

Source

Savchuk, O., Volinsky, A.A. (2020). “Nonparametric estimation of SiC film residual stress from the wafer profile”.

See Also

reg_2derivWL.

Examples

```
# EXAMPLE: Plotting scatter plot of deflection in the case of 0 degrees before the film
# deposition. Note: position is displayed in mm, whereas the deflection is displayed in
# micrometers. No smoothing is used. The original n=7755 data points are plotted.
dev.new()
plot(wafer40$x_dat*1000,wafer40$y_before_0*1000000,pch=20,
     main="Deflection BEFORE film deposition. Angle: 0 degrees.",xlab="",ylab="",
     cex.axis=2,cex.main=2)
title(ylab=expression(paste("deflection, ", mu, "m")), xlab="position, mm",
      line=2.25, cex.lab=2)
```

Index