Package 'vrcp'

January 8, 2016

Type Package

Title Change Point Estimation for Regression with Varying Segments and Heteroscedastic Variances

Description Estimation of varying regression segments and a change point in 2-segment regression models with heteroscedastic variances, and with or without a smoothness constraint at the change point.

Version 0.1.1

Author Jiayang Sun, Yulei Wang, and Junheng Ma

Maintainer Yulei Wang <yxw390@case.edu>

License GPL-2 | GPL-3

Imports splines

Depends ggplot2

LazyData true

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2016-01-08 00:27:43

R topics documented:

vrcp *Estimation for Varying Regression Segments and Change Point in Heteroscedastic Data*

Description

Estimation of two segments and a change point in 2-segment regression models with varying variances and varying types of regression segments, with or without a smoothness constraint at the change point.

Usage

```
vrcp(dataset, lo, hi, smooth = c("c0", "c1", "u"), segment1 = c("L", "Q",
  "Log", "Exp", "NLExp"), segment2 = c("L", "Q", "Log", "Exp", "NLExp"),
  variance = c("Common", "Diff"), spline = c("FALSE", "TRUE"), start)
```

Arguments

dataset	either a data frame, or a matrix, containing 2 columns, where the first column contains covariate values and the second column contains response values.
lo, hi	lower and upper bounds for the x value of the change point, to be set by a user.
smooth	smoothness constraint of the regression function at the change point. Default constraint is "c0," continuous at the change point. "c1" indicates that the first derivative at the change point is continuous, while "u" indicates that there is constraint at the change point.
segment1, segment2	regression model used to compute parameters in segment1(or segment2) with additive Gaussian errors. Currently allowable models are: L: Linear model. y ~ a0 + a1 * x Q: Quadratic model. y ~ a0 + a1 * x + a2 * x^2 Exp: Linearizable Exponential model. y ~ a0 + a1 * exp(x) Log: Linearizable Logarithm model. y ~ a0 + a1 * log(x) NLExp: Nonlinearizable Exponential model. y ~ a0 + a1 * exp(a2 * (x - k)), where k is the change point in x. These 5 models lead to the following 15 allowable combinations of varying types of segments, reflecting reasonable models we have seen from Degredation Science: "L"-"L", "L"-"Q", "L"-"Exp", "L"-"Log", "L"-"NLExp", "Q"-"L", "Q"-"Q", "Q"-"Exp", "Q"-"Log", and "Q"-"NLExp", "Exp"-"L", "Exp"-"Exp", "Log"-"L", "Log"-"Log", and "NLExp"-"L".
variance	variance type of the data set, either "Common" that requires the variances at two segments to be the same, or "Diff" that does NOT require them to be the same.

`spline` a "TRUE" or "FALSE" logical argument, where "TRUE" shows a B-spline fit with the knot at the change point, as an extra option only available for segment combinations of "L"-"L" and "Q"-"Q". The default is "FALSE".

`start` a named numeric vector or a logic status "FALSE". Default is "start=FALSE", which will compute initial values automatically based on data. "start=a vector" specifies initial values of vector parameters, typically in the case with a nonlinearizable segment as specified below, where a0, a1, a2 are regression parameters for segment1, and b0, b1, b2 are regression parameters for segment2, etc:

"L"-"NLExp": 3 initial values for (a0, a1; b2) may be specified.

"Q"-"NLExp": 4 initial values for (a0, a1, a2; b2) may be specified.

"NLExp"-"L": 3 initial values for (b0, b1; a2) may be specified.

Value

maxloglik: maximum log-likelihood value.

sigma2: estimated variance(s) for two segments

coe: coefficients for two regression segments, beta = (a0,a1,a2,b0,b1,b2). No a2, b2 output for linear/linearizable segment.

changepoint: change point in x value.

References

Stephen J. Ganocy and Jiayang Sun (2015), "Heteroscedastic Change Point Analysis and Application to Footprint Data", J of Data Science, v.13.

Examples

```
library(ggplot2)

# Test the vrcp() using simulated data sets

# Example 1: L-L model with "c0", continuity at change point and common variance
# Simulate the data
x1<-seq(0,2,by=0.05)
x2<-seq(2.05,5,by=0.05)

# The true regression functions
yt1 <- 2+0.5*x1
yt2 <- -1+2*x2

# Add noises
y1<-yt1+rnorm(length(x1),0,0.25)
y2<- yt2+rnorm(length(x2),0,0.25)
z<-data.frame(c(x1,x2),c(y1,y2))
names(z)=c("x","y")

# z is the simulated data in data frame. Let's visualize it
plot(z)
```

```
# It looks like a L-L regression with a change point between 1.5 and 2.5
# Fit with vrcp with L-L segments and "c0" constraint
ans <- vrcp(z,1.5,2.5,"c0","L","L","Common", spline = "TRUE") # Fit with common variance
ans

# The fitted L-L regression and spline are superimposed on the data
# Let's compare it with the true regression
x<-z$x
yt<-c(yt1,yt2)
ans$plot + ggplot2::geom_line(aes(x = x, y = yt), color = c("blue"), size=1) +
ggtitle("LL-c0-com model: Estimates vs. true model (in blue)")

ans <- vrcp(z,1.5,2.5,"c0","L","L","Diff",spline = "TRUE") # Fit with different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt), color = "blue", size=1) +
ggtitle("LL-c0-diff model: Estimates vs. true model (in blue)") # compare


## Not run:
# Example 2: L-Log model with "c1" change point and common variance.
# Simulate the data
x1<-seq(0.05,2.05,by=0.05)
x2<-seq(2.1,5.05,by=0.05)

# The true regression functions
yt1 <- 3+1*x1
yt2 <- 3.61+2*log(x2)

# Add noises
y1<- yt1+rnorm(length(x1),0,0.5)
y2<- yt2+rnorm(length(x2),0,0.5)
z<-data.frame(c(x1,x2),c(y1,y2))
names(z)<-c("x","y")

# z is the simulated data in data frame. Let's visualize it
plot(z)

# It looks like a L-Log regression with a change point between 1.9 and 2.2
# Fit with vrcp with specification of L-Log segments and "c1" options with
# and without common variance restriction
ans <- vrcp(z,1.9,2.2,"c1","L","Log","Common")
ans

# The fitted L-Log regression is superimposed on the data
# Let's compare it with the true regression
x<-z$x
yt<-c(yt1,yt2)
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LLog-c1-com model: Estimate (in magenta) vs. true model") # Fit with common variance

ans <- vrcp(z,1.9,2.5,"c1","L","Log","Diff")
ans
```

```
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LLog-c1-diff model: Estimate (in magenta) vs. true model") # Fit with different variance

# both fits look good

# Check what would look like with misspecification of smoothness at change point
ans <- vrcp(z,1.9,2.2,"c0","L","Log","Common") # Fit with common variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LLog-c0-com fit to LLog-c1-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,1.5,2.5,"c0","L","Log","Diff") # Fit with different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LLog-c0-diff fit to LLog-c1-com model: Estimate (in magenta) vs. true model")

# both look lack of fit, especially at the change point.
# Hence, the correct specification of model is important


# Example 3: Log-L - Simulated data set is "c1", smooth.
# Simulate the data
x1<-seq(2,4,by=0.05)
x2<-seq(4,7,by=0.05)

# The true regression functions
yt1 <- 1.6+0.5*log(x1)
yt2 <- 1.89+0.1*x2

# Add noises
y1<- yt1+rnorm(length(x1),0,0.1)
y2<- yt2+rnorm(length(x2),0,0.1)
z<-data.frame(c(x1,x2),c(y1,y2))
names(z)=c("x","y")

# z is the simulated data in data frame. Let's visualize it
plot(z)

# It looks like a Log-L regression with a change point between 3.9 and 4.5
# Fit with vrcp with specification of Log-L segments and "c1" options with
# and without common variance restriction
ans <- vrcp(z,3.9,4.5,"c1","Log","L","Common") # Fit with common variance
ans

# The fitted Log-L regression is superimposed on the data
# Let's compare it with the true regression
x<-z$x
yt<-c(yt1,yt2)
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
```

```
ggtitle("LogL-c1-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,3.8,4.2,"c1","Log","L","Diff") # Fit with different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LogL-c1-diff model: Estimate (in magenta) vs. true model")

# results look similar, not bad.

# Fit with Log-L segments and "c0" options with and without common variance restriction
ans <- vrcp(z,3.5,4.5,"c0","Log","L","Common") # Common variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LogL-c0-com fit to LLog-c1-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,3.5,4.5,"c0","Log","L","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LogL-c0-diff fit to LLog-c1-com model: Estimate (in magenta) vs. true model")

# Little worse than the one with c1 constraint

# Fit with Log-L segments and u" options with and without common variance restriction
ans <- vrcp(z,3.5,4.5,"u","Log","L","Common") # Common variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LLog-u-com fit to LLog-c1-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,3.5,4.5,"u","Log","L","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LLog-u-diff fit to LLog-c1-diff model: Estimate (in magenta) vs. true model")

# Clearly shows lack of fit at the change point.
# Again, the correct specification of the model, or use of available information is important.


# Example 4: QL-c1-com model, fitted by Q-L and Exp-L models,
# with and without a common variance constraint, respectively.

# Simulate Q-L data
x1<-seq(0,2,by=0.05)
x2<-seq(2,5,by=0.05)

# The true regression functions
yt1 <- 2+2*x1+2*x1^2
yt2 <- -6+10*x2
```

```
# Add noises
y1<- yt1+rnorm(length(x1),0,3)
y2<- yt2+rnorm(length(x2),0,3)
z<-data.frame(c(x1,x2),c(y1,y2))
names(z)=c("x","y")

# z is the simulated data in data frame. Let's visualize it
plot(z)

# It looks like a Q-L regression with a change point between 1.8 and 2.5
# Fit with vrcp with specification of Q-L segments and "c1" options with
# and without common variance restriction
ans <- vrcp(z,1.8,2.5,"c1","Q","L","Common")  # Common variance
ans

# The fitted Q-L regression is superimposed on the data
# Let's compare it with the true regression
x<-z$x
yt<-c(yt1,yt2)
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("QL-c1-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,1.8,2.5,"c1","Q","L","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("QL-c1-diff model: Estimate (in magenta) vs. true model")

# Fit with vrcp with specification of Exp-L segments and "c1" options with
# and without common variance restriction

ans <- vrcp(z,1.5,2.5,"c1","Exp","L","Common") # Common variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("ExpL-c1-com fit to QL-c1-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,1.5,2.5,"c1","Exp","L","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("ExpL-c1-diff fit to QL-c1-com model: Estimate (in magenta) vs. true model")

# Fit with vrcp with specification of Exp-L segments and "c0" options with
# and without common variance restriction

ans <- vrcp(z,1.5,2.5,"c0","Exp","L","Common") # Common variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("ExpL-c0-com fit to QL-c1-com model: Estimate (in magenta) vs. true model")
```

```
ans <- vrcp(z,1.5,2.5,"c0","Exp","L","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("ExpL-c0-com fit to QL-c1-com model: Estimate (in magenta) vs. true model")

# Fit with vrcp with specification of Exp-L segments and "u" options with
# and without common variance restriction

ans <- vrcp(z,1.5,2.5,"u","Exp","L","Common") # Common variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("ExpL-u-com fit to QL-c1-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,1.5,2.5,"u","Exp","L","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("ExpL-u-com fit to QL-c1-com model: Estimate (in magenta) vs. true model")

# Exp-L fits surprisingly well in this case.


# Example 5: Exp-Exp with "c0" change point and common variance. - No option of smoothness
# Simulate the data
x1<-seq(0,2,by=0.05)
x2<-seq(2.05,5,by=0.05)

# The true regression functions
yt1 <- 0.916+2*exp(x1)
yt2 <- 12+0.5*exp(x2)

# Add noises
y1<-yt1+rnorm(length(x1),0,5)
y2<-yt2+rnorm(length(x2),0,5)
z<-data.frame(c(x1,x2),c(y1,y2))
names(z)=c("x","y")

# z is the simulated data in data frame. Let's visualize it
plot(z)

# It looks like a Exp-Exp regression with a change point between 1.5 and 2.5
# Fit with vrcp with specification of Exp-Exp segments and "c0" options with
# and without common variance restriction

ans <- vrcp(z,1.5,2.5,"c0","Exp","Exp","Common") # Common variance ## simulation of smooth
ans

# The fitted Exp-Exp regression is superimposed on the data
# Let's compare it with the true regression
x<-z$x
yt<-c(yt1,yt2)
```

```
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("ExpExp-c0-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,1.5,2.5,"c0","Exp","Exp","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("ExpExp-c0-diff model: Estimate (in magenta) vs. true model")

# Fit with vrcp with specification of Exp-Exp segments and "u" options with
# and without common variance restriction

ans <- vrcp(z,1.5,2.2,"u","Exp","Exp","Common") # Common variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("ExpExp-u-com fit to ExpExp-c0-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,1.5,2.5,"u","Exp","Exp","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("ExpExp-u-diff fit to ExpExp-c0-com model: Estimate (in magenta) vs. true model")

# Unconstraint fits okay, considering information of continuity was not used.


# Example 6: Log-Log with "c0" change point and common variance. - No option of smoothness
# Simulate the data
x1<-seq(2,4,by=0.05)
x2<-seq(4.05,7,by=0.05)

# The true regression functions
yt1 <- 2 - 2*log(x1)
yt2 <- 13.1 - 10*log(x2)

# Add noises
y1<- yt1+rnorm(length(x1),0,0.5)
y2<- yt2+rnorm(length(x2),0,0.5)
z<-data.frame(c(x1,x2),c(y1,y2))
names(z)=c("x","y")

# z is the simulated data in data frame. Let's visualize it
plot(z)

# It looks like a Log-Log regression with a change point between 3.5 and 4.5
# Fit with vrcp with specification of Log-Log segments and "c0" options with
# and without common variance restriction

ans <- vrcp(z,3.5,4.5,"c0","Log","Log","Common") # Common variance
ans
```

```
# The fitted Log-Log regression is superimposed on the data
# Let's compare it with the true regression
x<-z$x
yt<-c(yt1,yt2)
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LogLog-c0-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,3.5,4.5,"c0","Log","Log","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LogLog-c0-diff model: Estimate (in magenta) vs. true model")

# Fit with vrcp with specification of Log-Log segments and "u" options with
# and without common variance restriction

ans <- vrcp(z,3.7,4.5,"u","Log","Log","Common") # Common variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LogLog-u-com fit to LogLog-c0-com model: Estimate (in magenta) vs. true model")


ans <- vrcp(z,3.7,4.5,"u","Log","Log","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LogLog-u-diff fit to LogLog-c0-com model: Estimate (in magenta) vs. true model")


# Example 7: Q-Exp with "c1" change point and common variance.
# Simulate the data
x1<-seq(0,2,by=0.05)
x2<-seq(2,5,by=0.05)

# The true regression functions
yt1 <-  .2+.2*x1+.5*x1^2
yt2 <- .3832+.3*exp(x2)

# Add noises
y1<- yt1+rnorm(length(x1),0,3)
y2<- yt2+rnorm(length(x2),0,3)
z<-data.frame(c(x1,x2),c(y1,y2))
names(z)=c("x","y")

# z is the simulated data in data frame. Let's visualize it
plot(z)

# It looks like a Q-Exp regression with a change point between 1.5 and 2.2
# Fit with vrcp with specification of Q-Exp segments and "c1" options with
# and without common variance restriction
```

```
ans <- vrcp(z,1.5,2.2,"c1","Q","Exp","Common") # Common variance
ans

# The fitted Q-Exp regression is superimposed on the data
# Let's compare it with the true regression
x<-z$x
yt<-c(yt1,yt2)
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("QExp-c0-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,1.5,2.2,"c1","Q","Exp","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("QExp-c0-diff model: Estimate (in magenta) vs. true model")


# Example 8: Q-Log with "c1" change point and common variance.
# Simulate the data
x1<-seq(0.05,2.05,by=0.05)
x2<-seq(2.1,5.05,by=0.05)

# The true regression functions
yt1 <- 2+1*x1+5*x1^2
yt2 <- 0+35*log(x2)

# Add noises
y1<-yt1+rnorm(length(x1),0,4)
y2<-yt2+rnorm(length(x2),0,4)
z<-data.frame(c(x1,x2),c(y1,y2))
names(z)=c("x","y")

# z is the simulated data in data frame. Let's visualize it
plot(z)

# It looks like a Q-Log regression with a change point between 1.5 and 2.5
# Fit with vrcp with specification of Q-Log segments and "c1" options with
# and without common variance restriction
ans <- vrcp(z,1.5,2.5,"c1","Q","Log","Common") # Common variance
ans

# The fitted Q-Log regression is superimposed on the data
# Let's compare it with the true regression
x<-z$x
yt<-c(yt1,yt2)
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("QLog-c1-com model: Estimate (in magenta) vs. true model")

ans <- vrcp(z,1.5,2.5,"c1","Q","Log","Diff") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
```

```
scale_colour_grey(name = "Model") +
ggtitle("QLog-c1-diff model: Estimate (in magenta) vs. true model")


# Example 9: Q-Q with "c1" change point and common variance.
# Simulate the data
x1<-seq(0.05,2,by=0.05)
x2<-seq(2.05,5,by=0.05)

# The true regression functions
yt1 <- 2+10*x1-5*x1^2
yt2 <- 30-21*x2+3.5*x2^2

# Add noises
y1<-yt1+rnorm(length(x1),0,2)
y2<-yt2+rnorm(length(x2),0,2)
z<-data.frame(c(x1,x2),c(y1,y2))
names(z)=c("x","y")

# z is the simulated data in data frame. Let's visualize it
plot(z)

# It looks like a Q-Q regression with a change point between 1.5 and 2.5
# Fit with vrcp with specification of Q-Q segments and "c1" options with
# and without common variance restriction
ans <- vrcp(z,1.5,2.5,"c1","Q","Q","Common",spline="TRUE") # Common variance
ans

# The fitted Q-Q regression and spline are superimposed on the data
# Let's compare it with the true regression
x<-z$x
yt<-c(yt1,yt2)
ans$plot + ggplot2::geom_line(aes(x = x, y = yt), color = "blue", size=1) +
ggtitle("QQ-c1-com model: Estimates vs. true model (in blue)")

ans <- vrcp(z,1.5,2.5,"c1","Q","Q","Diff",spline="TRUE") # Different variance
ans
ans$plot + ggplot2::geom_line(aes(x = x, y = yt), color = "blue", size=1) +
ggtitle("QQ-c1-diff model: Estimates vs. true model (in blue)")

# vrcp fits better than splines fitting.


# Example 10: L-NLExp with "c1" change point and common variance.
# Simulate the data
x1<- seq(0,2,by=0.05)
x2<- seq(2.05,5,by=0.05)

# The true regression functions
yt1 <- 10-0.8*x1
yt2 <- 8.4 - (-0.78/0.5) + (-0.78/0.5)*(exp(0.5*(x2-2)))

# Add noises
```

```
y1<- yt1+rnorm(length(x1),0,0.5)
y2 <- yt2+rnorm(length(x2),0,0.5)
z<-data.frame(c(x1,x2),c(y1,y2))
names(z)=c("x","y")

# z is the simulated data in data frame. Let's visualize it
plot(z)

# It looks like a L-NLExp regression with a change point between 1.8 and 2.05
# Fit with smooth L-NLExp, common or different variances

tryCatch(vrcp(z,1.8,2.05,"c1","L","NLExp","Common",start="FALSE"), error=function(e)
{return("Try different starting values. If this still fails, try a different nonlinear model
that might be more suitable to data.")})

ans <- vrcp(z,1.8,2.05,"c1","L","NLExp","Common",start="FALSE")
ans

# The fitted L-NLExp regression is superimposed on the data
# Let's compare it with the true regression
x<-z$x
yt<-c(yt1,yt2)
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LNLExp-c1-com model: Estimate (in magenta) vs. true model")

tryCatch(vrcp(z,1.8,2.05,"c1","L","NLExp","Diff",start="FALSE"), error=function(e){
return("Try different starting values. If this still fails, try a different nonlinear model
that might be more suitable to data.")})
ans <- vrcp(z,1.8,2.05,"c1","L","NLExp","Diff",start="FALSE")
ans$plot + ggplot2::geom_line(aes(x = x, y = yt, colour = c("true")), size=1) +
scale_colour_grey(name = "Model") +
ggtitle("LNLExp-c1-diff model: Estimate (in magenta) vs. true model")

## End(Not run)
```

Index