Tutorial for the R package chngpt

Youyi Fong

November 21, 2019

1 Types of threshold effects

1.1 Continuous threshold effects

The continuous threshold effects supported in the *chngpt* package (Figure 1.1) are

$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_- \qquad \text{(upper hinge)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_- + \beta_2 (x-e)_-^2 \qquad \text{(quad upper hinge)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_- + \beta_2 (x-e)_-^2 + \gamma x \qquad \text{(M21)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_- + \beta_2 (x-e)_-^2 + \beta_3 (x-e)_-^3 \qquad \text{(cubic upper hinge)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_- + \beta_2 (x-e)_-^2 + \beta_3 (x-e)_-^3 + \gamma x \qquad \text{(M31)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_+ \qquad \text{(hinge)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_+ + \beta_2 (x-e)_+^2 \qquad \text{(quad hinge)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_+ + \beta_2 (x-e)_+^2 + \gamma x \qquad \text{(M12)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_+ + \beta_2 (x-e)_+^2 + \beta_3 (x-e)_+^3 \qquad \text{(cubic hinge)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_+ + \beta_2 (x-e)_+^2 + \beta_3 (x-e)_+^3 + \gamma x \qquad \text{(M13)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_+ + \gamma x \qquad \text{(segmented)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e)_- + \beta_2 (x-e)_-^2 + \beta_3 (x-e)_+ + \beta_4 (x-e)_+^2 \qquad \text{(M22)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e) + \beta_2 (x-e)_-^2 + \beta_3 (x-e)_+^2 \qquad \text{(M22c)}$$
$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 (x-e) + \beta_2 (x-e)^2 + \beta_3 (x-e)_-^3 + \beta_4 (x-e)_+^3 \qquad \text{(M33c)}$$

where e denote the threshold parameter, x is the predictor with threshold effect, z denote a vector of additional predictors, and

$$(x-e)_+ = \begin{cases} x-e & \text{if} \quad x > e \\ 0 & \text{if} \quad \text{otherwise} \end{cases}$$
$$(x-e)_- = \begin{cases} 0 & \text{if} \quad x > e \\ x-e & \text{if} \quad \text{otherwise} \end{cases}.$$

Hinge and segmented models are studied in Fong et al. (2017b). Upper hinge models are studied in Elder and Fong (2019). (Quad/cubic)(hinge/upper hinge/M1/M2/M3a/M3b) are threshold models with quadratic and cubic effects that are under development.

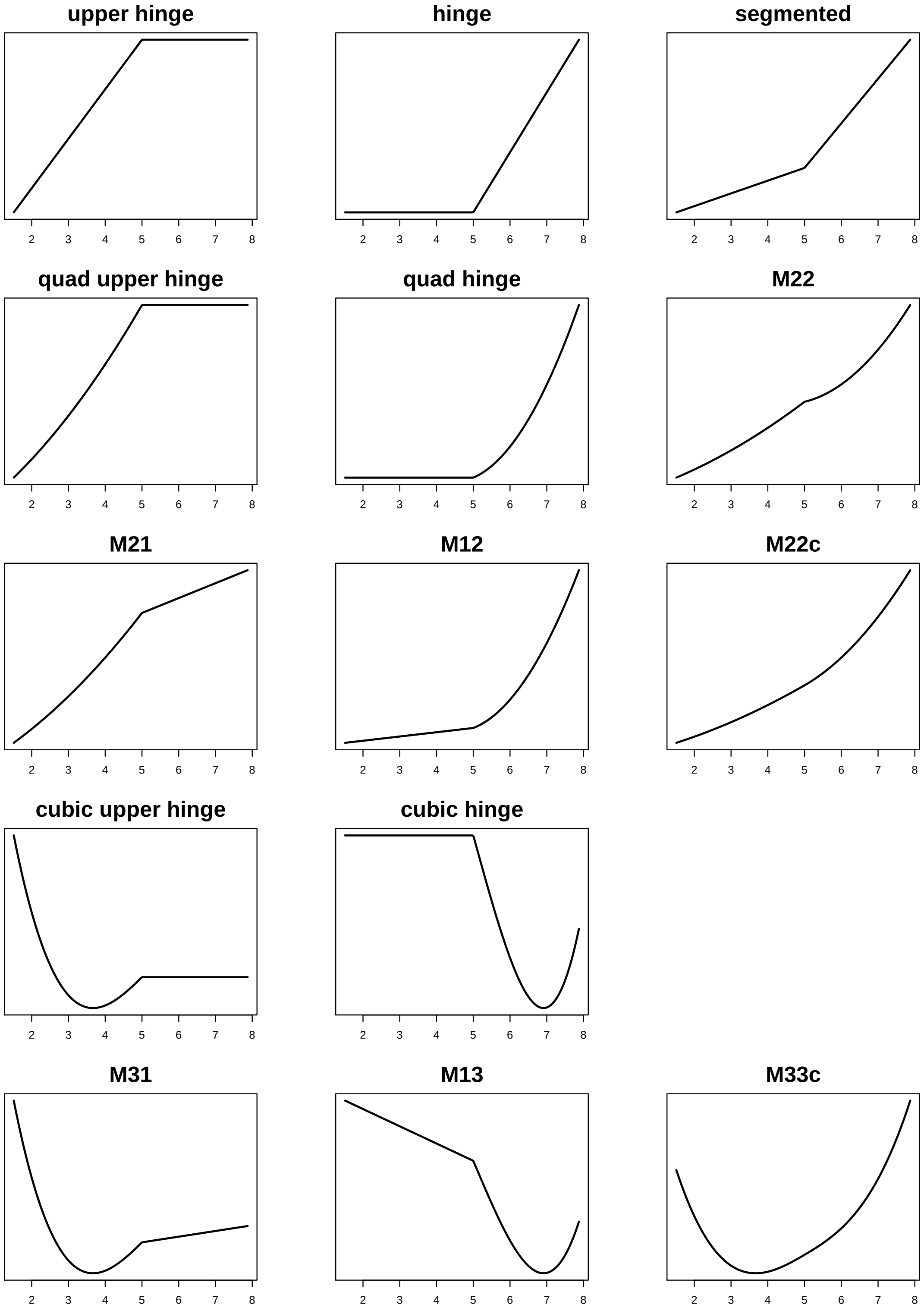

Figure 1.1: Types of continuous threshold effects supported in chngpt.

1.2 Discontinuous threshold effects

The discontinuous threshold effects supported in the *chngpt* package (Figure 1.2) are:

$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 I\left(x > e\right) \qquad \text{(step)}$$

$$\eta = \alpha_1 + \alpha_2^T z + \beta_1 \left(x - e\right)_+ + \gamma x + \beta_2 I\left(x > e\right), \qquad \text{(stegmented)}$$

where e denote the threshold parameter, x is the predictor with threshold effect, z denote a vector of additional predictors, and

$$I\left(x > e\right) = \begin{cases} 1 & \text{if} \quad x > e \\ 0 & \text{if} \quad \text{otherwise} \end{cases}$$

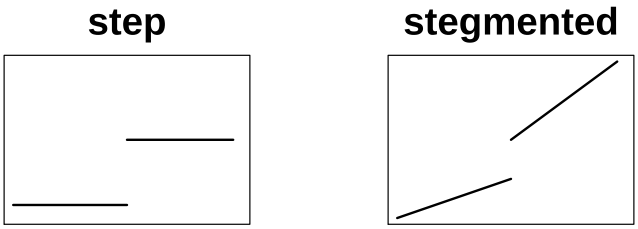

Figure 1.2: Types of discontinuous threshold effects supported in chngpt.

2 Examples

The examples below are organized by type of threshold effects and regression models. Before we get into specific examples, here are some notes that are of general interest:

- The fitted model has a component named best.fit, which is the glm or coxph fit at the estimated threshold parameter. This could be useful to know if one would like to extract information from model.

2.1 Continuous threshold linear regression

For continuous threshold linear regression, we have developed a grid search method for estimation that is super fast (Fong, 2018). Together with the observation that bootstrap confidence intervals have better coverage than robust analytical confidence intervals (Fong et al., 2017b) for continuous threshold linear models, we recommend setting `est.method="fastgrid"` and `var.type="bootstrap"` in the call to chngptm.

2.1.1 Example 1. The MTCT dataset, segmented model

To estimate a threshold linear regression model with a segmented-type change point for the relationship between *V3_BioV3B* and *NAb_score* in the *MTCT* dataset, we call

```
fit=chngptm (formula.1=V3_BioV3B~1, formula.2=~NAb_score, dat.mtct.2,
  type="segmented", family="gaussian",
  est.method="fastgrid", var.type="bootstrap", save.boot=TRUE)
```

- `formula.2` and `formula.1`: threshold variable and the rest of the model
- `type`: type of threshold model to fit
- `est.method` defaults to *fastgrid* and is recommended
- `var.type`: *bootstrap* method is recommended here
- `save.boot`: saves bootstrap samples for plotting bootstrap distributions

Calling `summary(fit)`, we get

```
Change point model type:  segmented

Coefficients:
                          est     p.value*    (lower    upper)
(Intercept)         -22.33152 1.593423e-08 -30.07675 -14.58628
NAb_score            67.23925 2.212981e-14  49.98398  84.49452
(NAb_score-chngpt)+ -64.83129 3.692679e-14 -81.61413 -48.04845

Threshold:
      est    (lower    upper)
0.4653923 0.4535000 0.4772845
```

Note that we there is an asterisk next to p.value. This is because bootstrap procedures to generate confidence intervals do not readily lead to p values. The presented p values are approximations, obtained assuming that the bootstrap sampling distributions are normal.

To get an estimate of the slope after threshold, we call

```
 est=lincomb(fit, comb=c(0,1,1), alpha=0.05); print(est)
```

and get

```
                        95%          95%
 2.40795883 -0.06780353  4.88372120
```

Calling `plot(fit, which=1)` and `plot(fit, which=3)`, we get the two plots on the left-hand side of Figure 2.1. Changing `est.method` to `smoothapprox` in the model fit led us to the two plots on the right-hand side.

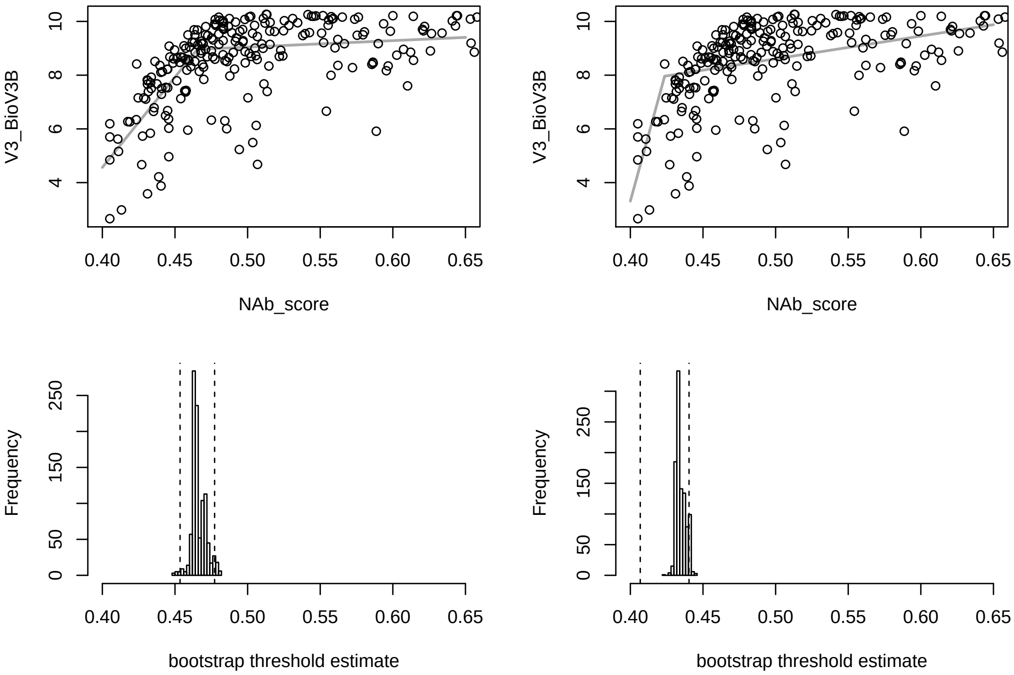

Figure 2.1: This is a replicate of Fong (2018) Figure 1. Left: results by fast grid search; right: results by smooth approximation search. Top: scatterplots with fitted models (gray lines); bottom: bootstrap distributions of the threshold estimate from 10^3 replicates. The dashed lines correspond to the 95% symmetric bootstrap confidence interval.

2.1.2 Example 2. The trees dataset, segmented model

To estimate a threshold linear regression model with a segmented-type change point in *Girth* for the *trees* dataset, we call

```
fit=chngptm(formula.1=Volume∼1, formula.2=∼Girth, data=trees,
type="segmented", family="gaussian",
 var.type="bootstrap", weights=NULL)
```

- `formula.2` and `formula.1`: threshold variable and the rest of the model
- `type`: type of threshold model to fit
- `var.type`: *bootstrap* method is recommended for confidence interval
- `weights` can be supplied

Calling `summary(fit)`, we get

```
Change point model type:  segmented

Coefficients:
                      est     p.value*     (lower     upper)
(Intercept)     -24.614440 1.985482e-04 -37.580354 -11.648527
Girth             3.993966 9.288973e-11   2.785558   5.202373
(Girth-chngpt)+   4.266618 8.261144e-04   1.765770   6.767467

Threshold:
   est (lower upper)
  16.0   12.9   19.1
```

Calling `plot(fit)`, we get Figure 2.2.

To test whether there is a change point (Fong et al., 2015), we call

```
test=chngpt.test(formula.null=Volume∼1, formula.chngpt=∼Girth, trees,
type="segmented", family="gaussian")
```

When printed, we get

```
        Maximum of Likelihood Ratio Statistics

data:  trees
Maximal statistic = 17.694, change point = 15.388, p-value = 0.00014
alternative hypothesis: two-sided
```

The first line gives the type of test carried out, and it is maximal likelihood ratio test here, which is the default. In addition, a plot function can be called on the test object to show the score or likelihood ratio statistic as a function of candidate change points.

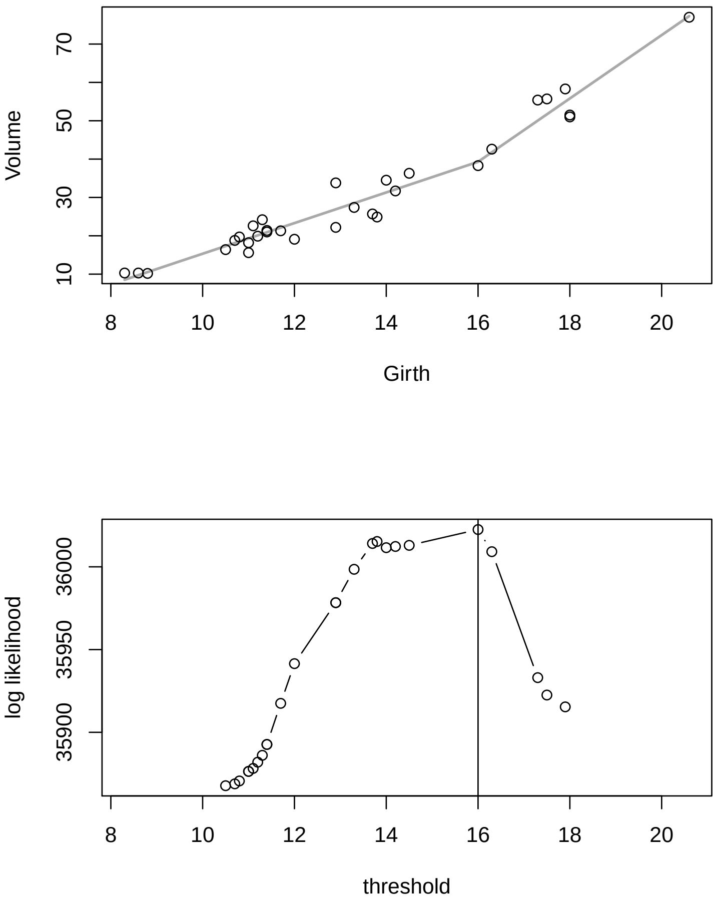

Figure 2.2: (a) Scatterplot of timber volume vs girth. The gray line shows the fitted segmented model. (b) Log likelihood of the submodel versus threshold parameter.

2.1.3 Example 3. The vapor pressure dataset, hinge quadratic model

To estimate a hinge quadratic linear regression model in *temperature* for the *pressure* dataset, we call

```
fit=chngptm(formula.1=pressure∼1, formula.2=∼temperature, data=pressure,

type="hingequad", family="gaussian", var.type="bootstrap")
```

Calling `summary(fit)`, we get

```
Change point model threshold.type:  hingequad

Coefficients:
                                est  p.value*       (lower     upper)
(Intercept)             8.278463507 0.4733673 -14.35129837 30.9082254
```

```
(temperature-chngpt)+      0.007124705 0.9944183  -2.00325636  1.9890069
I((temperature-chngpt)+^2) 0.039305656 0.3644561  -0.04564143  0.1242527

Threshold:
   est (lower upper)
   220   -680    240
```

2.2 Continuous threshold logistic regression

For continuous threshold logistic regression, a fast grid search method for estimation is not yet available. In addition, we have observed that bootstrap confidence intervals have similar coverage as robust analytical confidence intervals (Fong et al., 2017b). Thus, we recommend either `var.type="bootstrap"` or `var.type="robust"` in the call to chngptm. Note that when it is set to *robust*, an auxiliary fit needs to be supplied, which is generally a smooth parametric model with enough but not too many degrees of freedom.

To estimate a logistic regression model with a hinge-type change point in *NAb_ SF162L* for the MTCT dataset, we call

```
library(splines)
fit=chngptm(formula.1=y∼birth, formula.2=∼NAb_SF162LS, dat.mtct,
type="hinge", family="binomial",
est.method="smoothapprox", var.type="robust",
aux.fit=glm(y∼birth + ns(NAb_SF162LS,3), dat.mtct, family="binomial"))
```

- `formula.2` and `formula.1`: threshold variable and the rest of the model
- `type`: type of threshold model to fit
- `est.method`: *smoothapprox* is recommended
- `var.type`: *robust* is recommended for confidence interval
- `aux.fit`: required for *robust* variance estimation

Calling `summary(fit)`, we get

```
Change point model type:  hinge

Coefficients:
                             OR     p.value    (lower    upper)
(Intercept)           0.7026523 0.341429662 0.3388366 1.4571044
birthVaginal          1.2397649 0.523159883 0.6393632 2.4039809
(NAb_SF162LS-chngpt)+ 0.6712371 0.001332547 0.5270730 0.8548327

Threshold:
   26.3%   (lower   upper)
7.373374 5.472271 8.186464
```

To test whether there is a change point (Fong et al., 2015), we call

```
    test=chngpt.test(formula.null=y∼birth, formula.chngpt=∼NAb_SF162LS, dat.mtct,
type="hinge", family="binomial", main.method="score")
```

When printed, we get

```
        Maximum of Score Statistics

data:  dat.mtct
Maximal statistic = 3.3209, change point = 7.0347, p-value = 0.00284
alternative hypothesis: two-sided
```

The first line gives the type of test carried out, and it may be maximal likelihood ratio test. In addition, a plot function can be called on the test object to show the score or likelihood ratio statistic as a function of candidate change points.

2.2.1 cbind

The *chngptm* function supports the use of cbind in the formula, as the *glm* function does. For example,

```
dat.2=sim.chngpt("thresholded", "step", n=200, seed=1, beta=1, alpha=-1,
  x.distr="norm", e.=4, family="binomial")
dat.2$success=rbinom(nrow(dat.2), 10, 1/(1 + exp(-dat.2$eta)))
dat.2$failure=10-dat.2$success
fit.2a=chngptm(formula.1=cbind(success,failure)~z, formula.2=~x,
  family="binomial", dat.2, type="step")
```

2.3 Continuous threshold Poisson regression

Only grid search method and bootstrap confidence intervals are supported, so getting the model fit with confidence intervals could take some time.

```
counts <- c(18,17,15,20,10,20,25,13,12)
outcome <- as.integer(gl(3,1,9))
treatment <- gl(3,3)
print(d.AD <- data.frame(treatment, outcome, counts))
fit.4=chngptm(formula.1=counts ~treatment, formula.2=~outcome, data=d.AD,
  family="poisson", type="segmented", var.type="bootstrap")
summary(fit.4)
```

2.4 Discontinuous threshold GLM

Confidence interval for discontinuous threshold regression models can be constructed by m-out-of-n bootstrap. (The m.out.of.n argument is set to TRUE inside chngptm for the step and stegmented models.) For example,

```
fit=chngptm(formula.1=mpg~hp, formula.2=~drat, mtcars, type="step",
 family="gaussian", var.type="bootstrap", ci.bootstrap.size=100)
summary(fit)
```

The result:

```
Change point model threshold.type:  step

Coefficients:
                        est     p.value*      (lower      upper)
(Intercept)     27.29298302 3.412465e-20 21.48163370 33.10433234
hp              -0.05692654 2.675802e-05 -0.08349583 -0.03035726
I(drat>chngpt)   5.24824935 8.914610e-03  1.31503967  9.18145903

Threshold:
   est (lower upper)
  4.08   3.07   5.09
```

2.5 Threshold Cox regression

The *chngpt* package also provides some support for estimation of threshold Cox regression models. What is missing, though, is confidence intervals for parameter estimates and hypothesis testing methods. See the help page on *chngpt* for an example.

2.6 Models with interaction terms

In the following example we fit a model with an interaction term.

```
fit=chngptm(formula.1=mpg ~hp, formula.2=~hp*drat, mtcars, type="segmented",
 family="gaussian", var.type="bootstrap", ci.bootstrap.size=100)
summary(fit)
```

The model being fitted is

$$\eta = \beta_1 + \beta_2 z + \beta_3 x + \beta_4 (x - e)_+ + \beta_5 z x + \beta_6 z (x - e)_+$$

The result:

```
Change point model threshold.type:  segmented

Coefficients:
                          est  p.value*       (lower      upper)
(Intercept)        71.0423961 0.5949231 -190.8328276 332.9176199
hp                 -0.5714405 0.4809352   -2.1605786   1.0176976
drat              -14.3708279 0.7431579 -100.3306292  71.5889735
(drat-chngpt)+     21.6073593 0.6806816  -81.3015235 124.5162420
hp:drat             0.1658607 0.5333802   -0.3560702   0.6877916
hp:(drat-chngpt)+  -0.1970979 0.5620552   -0.8633923   0.4691965
```

```
Threshold:
   est (lower upper)
  3.23   2.35   4.11
```

In the following example we fit a model with two interaction terms.

```
fit=chngptm(formula.1=mpg~hp+wt, formula.2=~hp*drat+wt*drat, mtcars, type="step",
 family="gaussian", var.type="bootstrap", ci.bootstrap.size=100)
summary(fit)
```

The model being fitted is

$$\eta = \beta_1 + \beta_2 z_1 + \beta_3 z_2 + \beta_4 I\left(x > e\right) + \beta_5 z_1 I\left(x > e\right) + \beta_6 z_2 I\left(x > e\right)$$

The result:

```
Change point model threshold.type:  step

Coefficients:
                          est      p.value*      (lower     upper)
(Intercept)       30.83332346 1.458455e-06  18.2870806 43.3795663
hp                -0.02389962 7.233212e-01  -0.1562164  0.1084172
wt                -2.58756410 1.867228e-01  -6.4287268  1.2535986
I(drat>chngpt)    11.69827186 7.188926e-01 -52.0030753 75.3996190
hp:I(drat>chngpt) -0.00894615 9.652991e-01  -0.4119918  0.3940995
wt:I(drat>chngpt) -3.22148003 8.902722e-01 -48.9891600 42.5461999

Threshold:
   est (lower upper)
 3.730  3.237  4.223
```

3 Further considerations

3.1 Model choice

The choice of threshold effects is typically through a combination of domain knowledge and modeling. One modeling approach is to first examine the relationship using local polynomial regression.

The *hinge* and *upper hinge* models are both special cases of the *segmented* model with the slope before threshold fixed at 0. If the hinge model is reasonable, it is preferred over the segmented model because the model can be estimated with substantially higher precision (Fong et al., 2017b).

3.2 Estimation and inference methods

There are three types of search methods for finding the MLE (maximum likelihood estimator). Users generally do not need to worry about setting the argument, which is *est.method*, since the function chooses the most appropriate one by default. In the order of development, the three search methods are grid, smooth approximation, and fastgrid. The grid method is the most general and the slowest; it is recommended when other methods are not available. The smooth approximation method (Fong et al., 2017a) involves approximating the likelihood function with a differentiable function to allow gradient-based search; it is available for linear and logistic regression and mostly recommended for logistic regression only. Fastgrid (Fong, 2018; Elder and Fong, 2019) is a new method that is super fast and gives exact solutions; it is only available for certain threshold linear regression models.

Robust confidence interval methods are described in Fong et al. (2017b).

Hypothesis testing methods are described in Fong et al. (2015) and Fong et al. (2017a).

Acknowledgement

We are grateful for questions and comments from researchers around the world who are interested in using *chngpt*, which have led to great improvement to the package.

References

Elder, A. and Fong, Y. (2019), “Estimation and Inference for Upper Hinge Regression Models,” *Environmental and Ecological Statistics*, in press.

Fong, Y. (2018), “Fast Bootstrap Confidence Intervals for Continuous Threshold Linear Regression,” *Journal of Computational and Graphical Statistics*, in press.

Fong, Y., Di, C. and Permar, S. (2015), “Change point testing in logistic regression models with interaction term,” *Statistics in medicine*, 34, 1483–1494.

Fong, Y., Huang, Y., Gilbert, P. and Permar, S. (2017a), “chngpt: threshold regression model estimation and inference,” *BMC Bioinformatics*, 18, 454–460.

Fong, Y., Chong, D., Huang, Y. and Gilbert, P. (2017b), “Model-robust Inference for Continuous Threshold Regression Models,” *Biometrics*, 73, 452–462.