

Package ‘interflex’

March 6, 2017

Type Package

Title Multiplicative Interaction Models Diagnostics and Visualization

Version 1.0.3

Date 2017-3-5

Author Jens Hainmueller, Jonathan Mummolo and Yiqing Xu

Maintainer Yiqing Xu <yiqingxu@ucsd.edu>

Description Performs diagnostic tests of multiplicative interaction models and plots non-linear marginal effects of a treatment on an outcome across different values of a moderator.

License GPL-2

Imports Rcpp (>= 0.12.3), ggplot2 (>= 2.1.0), sandwich (>= 2.3-4),
pcse (>= 1.9), lmtest (>= 0.9-34), Lmoments (>= 1.2-3),
doParallel (>= 1.0.10), foreach (>= 1.4.3), mgcv(>= 1.8-16)

LinkingTo Rcpp, RcppArmadillo

NeedsCompilation yes

Repository CRAN

Date/Publication 2017-03-06 00:20:58

R topics documented:

interflex-package	2
inter.binning	2
inter.gam	6
inter.kernel	7
inter.raw	10
interflex-internal	12
s1	12
s2	12
s3	13
s4	13

Index	14
--------------	-----------

interflex-package

Multiplicative Interaction Models Diagnostics and Visualization

Description

Producing Flexible Marginal Effect Estimates with Multiplicative Interaction Models

Details

This package performs diagnostics and visualizations of multiplicative interaction models. Besides conventional linear interaction models, it provides two additional estimation strategies—linear regression based on pre-specified bins and locally linear regressions based on Gaussian kernels—to flexibly estimate the conditional marginal effect of a treatment variable on an outcome variable across different values of a moderating variable. These approaches relax the linear interaction effect assumption and safeguard against excessive extrapolation.

Author(s)

Jens Hainmueller; Jonathan Mummolo; Yiqing Xu (Maintainer).
Please report bugs to <yiqingxu@ucsd.edu>.

References

Jens Hainmueller; Jonathan Mummolo; Yiqing Xu. 2016. "How Much Should We Trust Estimates from Multiplicative Interaction Models? Simple Tools to Improve Empirical Practice." Available at SSRN: https://papers.ssrn.com/abstract_id=2739221.

See Also

[inter.raw](#), [inter.binning](#), [inter.gam](#), and [inter.kernel](#)

inter.binning

The Binning Estimator

Description

Implementing the binning estimator, a generalization of the multiplicative interaction model and conducting various diagnostic tests

Usage

```
inter.binning(Y, D, X, Z = NULL, FE = NULL, weights = NULL,
             data, na.rm = FALSE, nbins = 3, cutoffs = NULL,
             vartype = "homoscedastic",
             cl = NULL, time = NULL, pairwise = TRUE, figure = TRUE, main = NULL,
             Ylabel = NULL, Dlabel = NULL, Xlabel = NULL,
             xlim = NULL, ylim = NULL, interval = NULL, Xdistr = "histogram", wald = TRUE)
```

Arguments

Y	a string, name of the outcome variable.
D	a string, name of the treatment variable.
X	a string, name of the moderating variable.
Z	a vector of variable names of control variables.
weights	a string, name of the weighting variable.
FE	a vector of variable names of fixed effects indicators.
data	a data.frame that stores Y, D, X, and other variables.
na.rm	a logical flag indicating whether to list-wise delete data. The algorithm will report error if missing data exist.
nbins	an integer that determines the number of bins of the moderator. The sample will be divided based on integral multiples of 100/nbins percentile. nbins will be automatically subtracted by 1 if more than one multiples of the percentile have the same value (for example, if the moderator has over 70% zeros, both the 33 and 66 percentiles are zero). The default is 3. Ignored when cutoffs is supplied.
cutoffs	a vector of numbers that determines how subgroups are divided based on the moderator X. When this option is supplied, nbins will be ignored. The smallest number of the first interval and the largest number of the last interval do not need to be specified. Numbers smaller than the minimum or larger than the maximum of X will be ignored. nbins equals to the length of this vector plus 1. Ignored if the treatment is dichotomous.
vartype	a string that controls the variance-covariance estimator. Can be one of the following: "homoscedastic"; "robust", "cluster", and "pcse" (panel corrected standard errors). The default is "homoscedastic".
cl	a string specifying the name of clustering variable for clustered standard errors. vartype must be either "cluster" or "pcse". vartype will be automatically set to "cluster" if cl is supplied and vartype is not "pcse".
time	a string specifying the name of time variable for panel corrected standard errors. Ignored when vartype is not "pcse".
pairwise	a logical flag indicating whether to switch on the pairwise option when estimating panel corrected standard errors. Ignored when vartype is not "pcse". The default is TRUE.
Ylabel	a string that controls the label of Y in the plot.
Dlabel	a string that controls the label of D in the plot.
Xlabel	a string that controls the label of X in the plot.
main	a string that control the title of the plot.
xlim	a two-element numeric vector that controls the range of the x-axis to be shown in the plot.
ylim	a two-element numeric vector that controls the range of the y-axis to be shown in the plot (with small adjustments to improve aesthetics).
Xdistr	a string indicating the way the distribution of the moderator will be plotted, either "histogram" (or "hist") or "density". The default is "histogram".

<code>figure</code>	a logical flag indicating whether to draw the graph. The default is TRUE.
<code>interval</code>	manually set the interval for the graph.
<code>wald</code>	wald test.

Details

inter.binning implements the binning estimator. There are three steps to implement the estimator. First, we discretize the moderator X into several bins and create a dummy variable for each bin. The default is 3 bins, respectively corresponding to the three terciles. Second, we pick an evaluation point within each bin, where we want to estimate the conditional marginal effect of D on Y . Typically, we choose the median value of X in each bin. Third, we estimate a model that includes interactions between the bin dummies and the treatment indicator, the bin dummies and the moderator X minus the evaluation points, as well as the triple interactions. The last two terms are to capture the effect of D on Y within each bin.

The binning estimator has several key advantages over the standard multiplicative interaction model. First, the binning estimator is much more flexible as it jointly fits the interaction components of the standard model to each bin separately. The model does not impose the linear interaction effect (LIE) assumption. Instead, the conditional marginal effects can vary freely across the three bins and therefore can take on any non-linear or non-monotonic pattern that might describe the heterogeneity in the effect of D on Y across low, medium, or high levels of X .

Second, since the bins are constructed based on the support of X , the binning ensures that the conditional marginal effects are estimated at typical values of the moderator and do not rely on excessive extrapolation or interpolation.

Third, the binning estimator is easy to implement using any regression software and the standard errors for the conditional marginal effects are directly estimated by the regression so there are no need to compute linear combinations of coefficients to recover the conditional marginal effects.

Fourth, the binning estimator actually provides a generalization that nests the standard multiplicative interaction model as a special case. It can therefore serve as a formal test on the validity of a global LIE assumption imposed by the standard model. So in the special case when the standard multiplicative interaction model is correct and therefore the global LIE assumption holds, then—as the sample size grows—the marginal effect estimates from the binning estimator converge in probability on the unbiased marginal effect estimates from the standard multiplicative interaction model.

In the case of a binary treatment, **inter.binning** displays at the bottom of the figure a stacked histogram that shows the distribution of the moderator X . In this histogram the total height of the stacked bars refers to the distribution of the moderator in the pooled sample and the red and white shaded bars refer to the distribution of the moderator in the treatment and control groups, respectively. Adding such a histogram makes it easy to judge the degree to which there is common support in the data. In the case of a continuous treatment, it displays a histogram at the bottom that simply shows the distribution of X in the entire sample.

Value

<code>est.binning</code>	stores the binning estimates
<code>binary.treatment</code>	whether the treatment variable is binary

bin.size	the percentage of observations within each bin
treat.variation	the variance of the treatment variable within each bin
X.LKurtosis	the L-kurtosis measure of the moderator
correctOrder	whether the three binning estimates are in the correct order (i.e. monotonically increasing or decreasing)
p.twosided	p values of pairwise t-tests of the binning estimates when there are 2 or 3 bins.
p.wald	p-value of a Wald test. The NULL hypothesis is that the linear interaction model and the binning model are statistically equivalent.
graph	stores the graphic output, a ggplot2 object.

Author(s)

Jens Hainmueller; Jonathan Mummolo; Yiqing Xu (Maintainer)

References

Jens Hainmueller; Jonathan Mummolo; Yiqing Xu. 2016. "How Much Should We Trust Estimates from Multiplicative Interaction Models? Simple Tools to Improve Empirical Practice." Available at SSRN: https://papers.ssrn.com/abstract_id=2739221.

See Also

[inter.raw](#), [inter.gam](#), and [inter.kernel](#)

Examples

```
library(interflex)
data(interflex)
inter.binning(Y = "Y", D = "D", X = "X", Z = "Z1",
              data = s1, nbins = 3, vartype = "homoscedastic",
              Ylabel = "Y", Dlabel = "Tr", Xlabel="X")

## specifying cutoffs
inter.binning(Y = "Y", D = "D", X = "X", Z = "Z1",
              data = s2, cutoffs = c(1,2,4,5))

## with fixed effects
s4[, "wgt"] <- 1
inter.binning(Y = "Y", D = "D", X = "X", Z = "Z1",
              weights = "wgt", FE = c("group", "year"),
              data = s4, cl = "group", vartype = "cluster")
```

inter.gam

*Visualization of a Generalized Additive Model (GAM)***Description**

Estimating and plotting a GAM before estimating a multiplicative interaction model

Usage

```
inter.gam(Y, D, X, Z = NULL, weights = NULL, FE = NULL,
          data, SE = FALSE, k = 10, angle = c(30,100,-30,-120),
          Ylabel = NULL, Dlabel = NULL, Xlabel = NULL)
```

Arguments

Y	a string, name of the outcome variable.
D	a string, name of the treatment variable.
X	a string, name of the moderating variable.
Z	a vector of variable names of control variables.
weights	a string, name of the weighting variable.
FE	a vector of variable names of fixed effects indicators.
data	a data.frame that stores Y, D, X, and other variables.
SE	a logical variable indicating whether to show the surfaces one standard error above and below the predicted surface.
k	an integer indicating the dimension of the basis used to represent the smooth term. Smaller values of k produce smoother surfaces.
angle	a vector of numbers indicating viewing angles (allows 1 to 4 values).
Ylabel	a string that controls the label of Y in the plot.
Dlabel	a string that controls the label of D in the plot.
Xlabel	a string that controls the label of X in the plot.

Details

inter.gam provides a convenient way to visualize the relationships between the outcome, treatment, and moderator using GAM when both the treatment and moderator are continuous (based on [vis.gam](#) of the **mgcv** package).

Author(s)

Jens Hainmueller; Jonathan Mummolo; Yiqing Xu (Maintainer)

References

Jens Hainmueller; Jonathan Mummolo; Yiqing Xu. 2016. "How Much Should We Trust Estimates from Multiplicative Interaction Models? Simple Tools to Improve Empirical Practice." Available at SSRN: https://papers.ssrn.com/abstract_id=2739221.

See Also

[inter.raw](#), [inter.binning](#), and [inter.kernel](#)

Examples

```
library(interflex)
data(interflex)
inter.gam(Y="Y", D="D", X="X", Z=c("Z1"), data=s2)
```

inter.kernel

The Kernel Estimator

Description

Implementing the kernel estimator, based on a semi-parametric model that relaxes the linear interaction effect (LIE) assumption of conventional multiplicative interaction models and safeguards against excessive extrapolation

Usage

```
inter.kernel(Y, D, X, Z = NULL, weights = NULL, FE = NULL,
             data, na.rm = FALSE, CI = TRUE, conf.level = 0.95,
             cl = NULL, neval = 50, nboots = 200,
             parallel = TRUE, cores = 4, seed = 02139,
             bw = NULL, grid = 20, metric = "MSPE",
             Ylabel = NULL, Dlabel = NULL, Xlabel = NULL,
             main = NULL, xlim = NULL, ylim = NULL,
             Xdistr = "histogram", file = NULL)
```

Arguments

Y	a string, name of the outcome variable.
D	a string, name of the treatment variable.
X	a string, name of the moderating variable.
Z	a vector of variable names of control variables.
weights	a string, name of the weighting variable.
FE	a vector of variable names of fixed effects indicators.
data	a data.frame that stores Y, D, and X.
na.rm	a logical flag indicating whether to list-wise delete missing data. The algorithm will report an error if missing data exist.

CI	a logical flag indicating whether the confidence intervals need to be shown. If CI = FALSE , the bootstrap procedure will not be conducted.
conf.level	a number that specifies the confidence interval. For example, 0.9 means a confidence interval of [5%, 95%]. The default is 0.95, i.e., [2.5%, 97.5%].
cl	a string specifying the name of clustering variable. If specified, the bootstrap procedure will be blocking on this variable. Ignored when CI = FALSE .
neval	an integer specifying the number of evaluation points. A large number would produce smoother marginal effects estimates but the algorithm will take more time to run. The default is 50.
nboots	an integer specifying the number of bootstrap runs. The default is 200.
parallel	a logical flag indicating whether parallel computing will be used in bootstrapping and/or cross-validation. Ignored if CI = FALSE and bw is specified.
cores	an integer indicating the number of cores to be used in parallel computing. If not specified, the algorithm will use the maximum number of logical cores of your computer (warning: this could prevent you from multi-tasking on your computer).
seed	an integer that sets the seed in random number generation. Ignored if CI = FALSE and bw is specified.
bw	a number that sets the bandwidth in Gaussian kernel reweighting.
grid	either an integer indicating the number of candidates in a grid search to find the optimal bandwidth, or a vector of positive numbers of candidate bandwidths. Ignored if bw is specified.
metric	Either "MSPE" (Mean Squared Prediction Error) or "MAPE" (Mean Absolute Prediction Error), criterion to be used in cross-validation to select bandwidth. The default is "MSPE".
Ylabel	a string that controls the label of Y in the plot.
Dlabel	a string that controls the label of D in the plot.
Xlabel	a string that controls the label of X in the plot.
main	a string that control the title of the plot.
xlim	a two-element numeric vector that controls the range of the x-axis to be shown in the plot.
ylim	a two-element numeric vector that controls the range of the y-axis to be shown in the plot (with small adjustments to improve aesthetics).
Xdistr	a string indicating the way the distribution of the moderator will be plotted, either "histogram" (or "hist") or "density". The default is "histogram".
file	save the output graph to the file.

Details

inter.kernel implements a kernel smoothing estimator of the marginal effect, which is an application of semi-parametric smooth varying-coefficient models. This approach is more complicated than the simple binning estimator, but provides a generalization that allows researchers to flexibly estimate the functional form of the marginal effect of D on Y across the values of X by estimating a series of local effects with a Gaussian kernel re-weighting scheme.

The kernel regression nests the standard linear interaction model as a special case. However, in the kernel regression the conditional effect of D on Y does not have to fall on a linear line as required by the linear interaction effect (LIE) assumption, but can vary freely across the range of X. In addition, if covariates Z are included in the model, the coefficients of those covariates are also allowed to vary freely across the range of X resulting in a very flexible estimator that also helps to guard against misspecification bias with respect to the covariates.

inter.kernel produces a smooth marginal effect of D on Y with respect to X. It estimates a series of locally linear regression using kernel re-weighting based on the distance between each value of X (the moderator) and each evaluation point. If the bandwidth is not supplied, it is selected using a standard 5-fold cross-validation procedure. Standard errors and confidence intervals can be computed using a bootstrap. It also adds a histogram (or density plot) at the bottom of the figure to help users assess the common support assumption based on the distribution of the moderator.

inter.kernel can accommodate linear fixed-effect models with multiple fixed-effect indicators. A fixed-effect model is estimated using a fast algorithm written in C++ at each evaluation point of the semi-parametric kernel estimation.

Value

bw	specified or cross-validated bandwidth.
est	marginal effects estimates (and confidence intervals if CI = TRUE)
CV.out	results from cross-validation if bw is not specified.
graph	stores the graphic output, a ggplot2 object.

Author(s)

Jens Hainmueller; Jonathan Mummolo; Yiqing Xu (Maintainer)

References

Jens Hainmueller; Jonathan Mummolo; Yiqing Xu. 2016. "How Much Should We Trust Estimates from Multiplicative Interaction Models? Simple Tools to Improve Empirical Practice." Available at SSRN: https://papers.ssrn.com/abstract_id=2739221.

See Also

[inter.raw](#), [inter.gam](#), and [inter.binning](#)

Examples

```
library(interflex)
data(interflex)
inter.kernel(Y = "Y", D = "D", X = "X", Z = "Z1", data = s3,
            nboots = 50, bw = 1.3, parallel = FALSE)
```

inter.raw

*Plotting Raw Data***Description**

Plotting raw data before estimating a multiplicative interaction model

Usage

```
inter.raw(Y, D, X, weights = NULL, data,
          nbins = 3, cutoffs = NULL, span = NULL,
          Ylabel = NULL, Dlabel = NULL, Xlabel = NULL, pos = NULL)
```

Arguments

Y	a string, name of the outcome variable.
D	a string, name of the treatment variable.
X	a string, name of the moderating variable.
weights	a string, name of the weighting variable. It will influence both the linear and loess fits.
data	a data.frame that stores Y, D, and X.
nbins	an integer that determines the number of bins of the moderator (and hence the number of subplots) when the treatment is continuous. The sample will be divided based on integral multiples of 100/nbins percentile. nbins will be automatically subtracted by 1 if more than one multiples of the percentile have the same value (for example, if the moderator has over 70% zeros, both the 33 and 66 percentiles are zero). The default is 3. Ignored when cutoffs is supplied.
cutoffs	a vector of numbers that determines how subgroups are divided based on the moderator X. When this option is supplied, nbins will be ignored. The smallest number of the first interval and the largest number of the last interval do not need to be specified. Numbers smaller than the minimum or larger than the maximum of X will be ignored. nbins equals the length of this vector plus 1. Ignored if the treatment is dichotomous.
span	hyperparameter for the loess fit. Large numbers produce smoother lines (from ggplot2).
Ylabel	a string that controls the label of Y in the plot.
Dlabel	a string that controls the label of D in the plot.
Xlabel	a string that controls the label of X in the plot.
pos	the interval of Y.

Details

inter.raw provides a simple visual diagnostic, a scatterplot, to help researchers to detect potential problems with the linear interaction effect assumption and the lack of common support.

If the treatment D is binary, it plots outcome Y against the moderator X separately for the sample of treatment group observations ($D=1$) and the sample of control group observations ($D=0$). In each sample, it superimposes a linear regression line as well as loess fit in each group.

To check whether there is sufficient common support in the data, we can simply compare the distribution of X in both groups and examine the range of X values for which there are a sufficient number of data points in both groups. The box plots near the center of the figures display quantiles of the moderator at each level of the treatment. The dot in the center denotes the median, the end points of the thick bars denote the 25th and 75th percentiles, and the end points of the thin bars denote the 5th and 95th percentiles.

If the treatment and moderator are continuous, we adopt a simple binning approach. Specifically, **inter.raw** will split the sample into several subgroups based on the moderator. By default, it will split the sample into three roughly equal sized subgroups: low X (first tercile), medium X (second tercile), and high X (third tercile). For each of the subgroups, it plots Y against D while again adding the estimates for both the linear fit and the loess fit.

Value

graph stores the graphic output, a **ggplot2** object.

Author(s)

Jens Hainmueller; Jonathan Mummolo; Yiqing Xu (Maintainer)

References

Jens Hainmueller; Jonathan Mummolo; Yiqing Xu. 2016. "How Much Should We Trust Estimates from Multiplicative Interaction Models? Simple Tools to Improve Empirical Practice." Available at SSRN: https://papers.ssrn.com/abstract_id=2739221.

See Also

[inter.gam](#), [inter.binning](#), and [inter.kernel](#)

Examples

```
library(interflex)
data(interflex)
inter.raw(Y = "Y", D = "D", X = "X", data = s1,
         Ylabel = "Outcome", Dlabel = "Treatment", Xlabel="Moderator")
```

interflex-internal *Internal Interflex Functions*

Description

Internal Interflex functions

Details

These are not to be called by the user (or in some cases are just waiting for proper documentation to be written :).

s1 *s1*

Description

Simulated dataset 1: dichotomous treatment with linear marginal effects.

Usage

s1

Format

dataframe

s2 *s2*

Description

Simulated dataset 2: continuous treatment with linear marginal effects.

Usage

s2

Format

dataframe

s3

s3

Description

Simulated dataset 3: dichotomous treatment with nonlinear marginal effects.

Usage

s3

Format

dataframe

s4

s4

Description

Simulated dataset 4: dichotomous treatment with fixed effects..

Usage

s4

Format

dataframe

Index

*Topic **datasets**

- s1, [12](#)
- s2, [12](#)
- s3, [13](#)
- s4, [13](#)

*Topic **graphics**

- inter.binning, [2](#)
- inter.gam, [6](#)
- inter.kernel, [7](#)
- inter.raw, [10](#)

*Topic **ts**

- interflex-internal, [12](#)

coefs (interflex-internal), [12](#)

crossvalidate (interflex-internal), [12](#)

fastplm (interflex-internal), [12](#)

fastplm_predict (interflex-internal), [12](#)

inter.binning, [2](#), [2](#), [7](#), [9](#), [11](#)

inter.gam, [2](#), [5](#), [6](#), [9](#), [11](#)

inter.kernel, [2](#), [5](#), [7](#), [7](#), [11](#)

inter.raw, [2](#), [5](#), [7](#), [9](#), [10](#)

interflex (interflex-package), [2](#)

interflex-internal, [12](#)

interflex-package, [2](#)

interflex_fastplm (interflex-internal),
[12](#)

interflex_fastplm_predict
(interflex-internal), [12](#)

s1, [12](#)

s2, [12](#)

s3, [13](#)

s4, [13](#)

vcovCluster (interflex-internal), [12](#)

vis.gam, [6](#)