Statistics/Data Analysis

Title

lpdensity —— Local Polynomial Density Estimation and Inference.

Syntax

lpdensity *var* [*if*] [*in*] [, **grid(***var***) bw(***var* or *#***) p(***#***) q(***#***) v(***#***)**
bwselect(*BwMethod***) kernel(***KernelFn***) scale(***#***) level(***#***) cweights(***var***)**
pweights(*var***) genvars(***VarName***) separator(***#***) plot graph_options(***GraphOpts***)**]

Description

lpdensity implements the local polynomial regression based density (and derivatives) estimator proposed in Cattaneo, Jansson and Ma (2017a). This command can also be used to obtain smoothed estimates for cumulative distribution functions. See Cattaneo, Jansson and Ma (2017b) for more implementation details and illustrations.

Companion R functions are also available here.

Related Stata and R packages useful for nonparametric estimation and inference are described in the following website:

https://sites.google.com/site/nppackages/

Options

grid(*var*) specifies the grid on which density is estimated. When set to default, grid points will be chosen as 0.05−0.95 percentiles of the data, with 0.05 step size.

bw(*var* or *#*) specifies the bandwidth (either a variable containing bandwidth for each grid point or a single number) used for estimation. When omitted, bandwidth will be computed by method specified in **bwselect(***BwMethod***)**.

p(*#*) specifies the the order of the local−polynomial used to construct point estimates. Default is **p(2)** (local quadratic regression).

q(*#*) specifies the order of the local−polynomial used to construct pointwise confidence interval (a.k.a. the bias correction order). Default is **p(*#*)+1**. When specified the same as **p(*#*)**, no bias correction will be performed. Otherwise it should be strictly larger than **p(*#*)**.

v(*#*) specifies the derivative of distribution function to be estimated. **v(0)** for the distribution function, **v(1)** (default) for the density funtion, etc.

bwselect(*BwMethod*) specifies method for data−driven bandwidth selection. This option will be ignored if **bw(*var*)** is provided. Options are:
mse−dpi for mean squared error−optimal bandwidth selected for each grid point. This is the default option.
imse−dpi for integrated MSE−optimal bandwidth, common for all grid points.
mse−rot for rule−of−thumb bandwidth with Gaussian reference model.
imse−rot for integrated rule−of−thumb bandwidth with Gaussian reference model.

kernel(*KernelFn*) specifies the kernel function used to construct the local−polynomial estimator(s). Options are: **triangular**, **epanechnikov**, and **uniform**. Default is **triangular**.

scale(*#*) controls how estimates are scaled. For example, setting this parameter to 0.5 will scale down both the point estimates and standard errors by half. Default is **scale(1)**. This parameter is used if only part of the sample is used for estimation.

level(*#*) controls the level of the confidence interval, and should be between 0 and 100. Default is **level(95)**.

cweights(*var*) specifies weights used for counterfactual distribution construction.

pweights(*var*) specifies weights used in sampling. Should be nonnegative.

genvars(*VarName*) specifies if new varaibles should be generated to store estimation results. If *VarName* is provided, the following new varaibles will be generated:
VarName_grid (grid points),
VarName_bw (bandwidth),
VarName_nh (effective sample size),
VarName_f_p and *VarName_se_p* (point estimate with polynomial order **p(#)** and the corresponding standard error),
VarName_f_q and *VarName_se_q* (point estimate with polynomial order **q(#)** and the corresponding standard error, only available if different from **p(#)**),
VarName_CI_l and *VarName_CI_r* (confidence interval).

separator(*#*) draws separator line after every *#* variables; default is separator(5).

plot if specified, point estimates and confidence intervals will be plotted.

graph_options(*GraphOpts*) specifies options for plotting.

Examples

Generate artifitial data:

```
. set obs 1000
. set seed 42
. gen lpd_data = rnormal()
```

Density estimation at empirical quantiles:

```
. lpdensity lpd_data
```

Density estimation on a fixed grid (0.1, 0.2, ..., 1):

```
. gen lpd_grid = _n / 10 if _n <= 10
. lpdensity lpd_data, grid(lpd_grid)
```

Save estimation results to variables and plot:

```
. capture drop temp_*
. lpdensity lpd_data, genvars(temp) plot
```

Saved results

lpdensity saves the following in **e()**:

Scalars

e(N)	sample size
e(p)	option **p(#)**
e(q)	option **q(#)**
e(v)	option **v(#)**
e(scale)	option **scale(#)**
e(level)	option **level(#)**

Macros

e(bwselect)	option **bwselect(***BwMethod***)**
e(kernel)	option **kernel(***KernelFn***)**

Matrices

e(result)	estimation result

References

Cattaneo, M. D., Michael Jansson, and Xinwei Ma. 2017a. Simple Local Polynomial Density Estimators.
Working paper, University of Michigan.

Cattaneo, M. D., Michael Jansson, and Xinwei Ma. 2017b. lpdensity: Local Polynomial Density Estimation and Inference.
Working paper, University of Michigan.

Authors

Matias D. Cattaneo, University of Michigan, Ann Arbor, MI. cattaneo@umich.edu.

Michael Jansson, University of California at Berkeley, Berkeley, CA. mjansson@econ.berkeley.edu.

Xinwei Ma, University of Michigan, Ann Arbor, MI. xinweima@umich.edu.