Stata
Statistics/Data analysis

Title

lpdensity —— Local Polynomial Density Estimation and Inference.

Syntax

lpdensity *Var* [*if*] [*in*] [,
grid(*Var***)** **bw(***Var* or *#***)** **p(***#***)** **q(***#***)** **v(***#***)** **kernel(***KernelFn***)** **scale(***#***)**
nomasspoints
bwselect(*BwMethod***)** **nlocalmin(***#***)** **nuniquemin(***#***)** **noregularize** **nostdvar**
cweights(*Var***)** **pweights(***Var***)**
genvars(*NewVarName***)**
rgrid(*Var***)** **rindex(***Var***)** **level(***#***)** **ciuniform** **cisimul(***#***)** **separator(***#***)**
plot
estype(*ESOpts***)** **esline_options(***ESLineOpts***)** **espoint_options(***ESPointOpts***)**
citype(*CIOpts***)** **ciregion_options(***CIRegionOpts***)** **ciline_options(***CILineOpts***)**
ciebar_options(*CIEbarOpts***)**
histogram **hiplot_options(***HistOpts***)**
graph_options(*GraphOpts***)**
]

Description

lpdensity implements the local polynomial regression based density (and derivatives) estimator proposed in Cattaneo, Jansson and Ma (2020). Robust bias-corrected inference, both pointwise (confidence intervals) and uniform (confidence bands) are also implemented following the results in Cattaneo, Jansson and Ma (2020) and Cattaneo, Jansson and Ma (2021a). See Cattaneo, Jansson and Ma (2021b) for more implementation details and illustrations.

Companion command: lpbwdensity for bandwidth selection.

Companion R functions are also available here.

Related Stata and R packages are available in the following website:

https://nppackages.github.io/

Options

Estimation

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 local polynomial order for constructing point estimates. Default is **p(2)** (local quadratic regression).

q(*#*) specifies the local polynomial order for constructing confidence intervals/bands (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.

kernel(*KernelFn*) specifies the kernel function used to construct the local-polynomial estimator(s).
triangular   *K(u) = (1 - |u|) * (|u|<=1)*. This is the default option.
epanechnikov *K(u) = 0.75 * (1 - u^2) * (|u|<=1)*.
uniform   *K(u) = 0.5 * (|u|<=1)*.

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 useful when only a subsample is employed for estimation.

nomasspoints will not adjust point estimates or standard errors even if there are mass points in the data.

Bandwidth Selection

bwselect(*BwMethod*) specifies method for data-driven bandwidth selection. This option will be ignored if **bw(***Var***)** is provided.
- **mse-dpi** mean squared error optimal bandwidth for each grid point. This is the default option.
- **imse-dpi** integrated mean squared error optimal bandwidth which is common for all grid points.
- **mse-rot** rule-of-thumb bandwidth based on a Gaussian reference model.
- **imse-rot** integrated rule-of-thumb bandwidth based on a Gaussian reference model which is common for all grid points.

nlocalmin(*#*) specifies the minimum number of observations in each local neighborhood. This option will be ignored if set to 0, or if **noregularize** is used. The default value is **20+p(***#***)+1**.

nuniquemin(*#*) specifies the minimum number of unique observations in each local neighborhood. This option will be ignored if set to 0, or if **noregularize** is used. The default value is **20+p(***#***)+1**.

noregularize suppresses local sample size checking.

nostdvar will not standardize the data for bandwidth selection. Note that this may lead to unstable performance of the numerical optimization procedure.

Weights

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

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

Storing and displaying results

genvars(*NewVarName*) specifies if new varaibles should be generated to store estimation results. If *NewVarName* is provided, the following new varaibles will be generated:
- *NewVarName_grid* grid points,
- *NewVarName_bw* bandwidth,
- *NewVarName_nh* local/effective sample sizes,
- *NewVarName_f_p* and *NewVarName_se_p* point estimates with polynomial order **p(***#***)** and the corresponding standard errors,
- *NewVarName_f_q* and *NewVarName_se_q* point estimates with polynomial order **q(***#***)** and the corresponding standard errors, only available if different from **p(***#***)**,
- *NewVarName_CI_l* and *NewVarName_CI_r* confidence intervals/bands.

rgrid(*var*) specifies a set of grid points to display the results. When omitted, this will be the same as **grid(***Var***)**.

rindex(*var*) specifies a set of indices to display the results. This option will be ignored if **rgrid(***Var***)** is provided.

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

ciuiform computes a uniform confidence band instead of pointwise confidence intervals.

cisimul(#) specifies the number of simulations used to construct critical values. Default is **cisimul(2000)**. This option will be ignored unless **ciuniform** is provided.

separator(#) draw a seperation line after every # variables; default is **separator(5)**.

Plotting

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

estype(*ESOpts*) specifies the plotting style of point estimates.
- **line** a curve. This is the default option.
- **points** individual points.
- **both** both of the above.
- **none** will not plot point estimates.

esline_options(*ESlineOpts*) specifies additional **twoway line** options for plotting point estimates.

espoint_options(*ESPointOpts*) specifies additional **twoway scatter** options for plotting point estimates.

citype(*CIOpts*) specifies the plotting style of confidence intervals/bands.
- **region** shaded region. This is the default option.
- **line** upper and lower bounds.
- **ebar** error bars.
- **all** all of the above.
- **none** will not plot confidence intervals/bands.

ciregion_options(*CIRegionOpts*) specifies additional **twoway rarea** options for plotting confidence intervals/regions.

ciline_options(*CILineOpts*) specifies additional **twoway rline** options for plotting confidence intervals/regions.

ciebr_options(*CIEbarOpts*) specifies additional **twoway rcap** options for plotting confidence intervals/regions.

histgram if specified, a histogram will be included in the background.

hiplot_options(*HistOpts*) specifies additional **twoway histogram** options for the histogram.

graph_options(*GraphOpts*) specifies additional options for plotting, such as legends and labels.

Remarks

Bias correction is only used for the construction of confidence intervals/bands, but not for point estimation. The point estimates, denoted by f_p, are constructed using local polynomial estimates of order **p(#)**, while the centering of the confidence intervals/bands, denoted by f_q, are constructed using local polynomial estimates of order **q(#)**. The confidence intervals/bands take the form: [f_q - cv * SE(f_q) , f_q + cv * SE(f_q)], where cv denotes the appropriate critical value and SE(f_q) denotes an standard error estimate for the centering of the confidence interval/band. As a result, the confidence intervals/bands may not be centered at the point estimates because they have been bias-corrected. Setting **q(#)** and **p(#)** to be equal results on centered at the point estimate confidence intervals/bands, but requires undersmoothing for valid inference (i.e., (I)MSE-optimal bandwdith for the density point estimator cannot be used). Hence the bandwidth would need to be specified manually when **q(#)** = **p(#)**, and the point estimates will not be (I)MSE optimal. See Cattaneo, Jansson and Ma (2020, 2021a) for details, and also Calonico, Cattaneo, and Farrell (2018, 2020) for robust bias correction methods.

Sometimes the density point estimates may lie outside of the confidence intervals/bands, which can happen if the underlying distribution exhibits high curvature at some evaluation point(s). One possible solution in this case is to increase the polynomial order **p(*#*)** or to employ a smaller bandwidth.

Examples

Generate artifitial data:

```
. set obs 2000
. set seed 42
. gen lpd_data = rnormal()
```

Density estimation at empirical quantiles:

```
. lpdensity lpd_data
```

Density estimation at empirical quantiles with the IMSE-optimal bandwidth:

```
. lpdensity lpd_data, bwselect(imse-dpi)
```

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)
```

Report uniform confidence bands (instead of pointwise confidence intervals):

```
. lpdensity lpd_data, ciuniform
. lpdensity lpd_data, ciuniform level(99)
```

Save estimation results to new variables:

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

Density plot:

```
. lpdensity lpd_data, plot
. lpdensity lpd_data, plot histogram
. lpdensity lpd_data, plot histogram ciuniform level(90)
```

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

Calonico, S., M. D. Cattaneo, and M. H. Farrell. 2018. On the Effect of Bias Estimation on Coverage Accuracy in Nonparametric Inference. *Journal of the American Statistical Association* 113(522): 767-779.

Calonico, S., M. D. Cattaneo, and M. H. Farrell. 2020. Coverage Error Optimal Confidence Intervals for Local Polynomial Regression. Working paper.

Cattaneo, M. D., Michael Jansson, and Xinwei Ma. 2020. Simple Local Polynomial Density Estimators. *Journal of the American Statistical Association* 115(531): 1449-1455.

Cattaneo, M. D., Michael Jansson, and Xinwei Ma. 2021a. Local Regression Distribution Estimators. *Journal of Econometrics*, forthcoming.

Cattaneo, M. D., Michael Jansson, and Xinwei Ma. 2021b. lpdensity: Local Polynomial Density Estimation and Inference. *Journal of Statistical Software*, forthcoming.

Authors

Matias D. Cattaneo, Princeton University, Princeton, NJ. cattaneo@princeton.edu.

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

Xinwei Ma, University of California San Diego, La Jolla, CA. x1ma@ucsd.edu.