Title

logdensity — Univariate density estimation

Syntax

logdensity [*varname*] [*if*] [*in*] **, x(namelist) h(#)** [*more options*]

options	Description
Main	
x(*namelist***)**	points at which to estimate the logarithm of the density and its derivative(s); can be a variable name or a row vector; missing entries are silently dropped
h(*#***)**	bandwidth to be used
kernel**(***namelist***)**	kernel function; default is *epanechnikov*
degree**(***#***)**	order of the local polynomial approximation; default is *1* (local linear)
minx**(***#***)**	left end of support of *varname*; default is minimum value of *varname* - *h*, effectively negative infinity
maxx**(***#***)**	right end of support of *varnae*; default is maximum value of *varname* + *h*, effectively infinity
g(*string***)**	function used to estimate derivative(s) of the log-density; see below for description of default behavior
dg(*string***)**	derivative of *g*; see below for description of default behavior

Description

logdensity estimates the (natural) logarithm of the density of the variable in *varname* using a local polynomial approximation. This command is an implementation of the estimator developed in Pinkse, J. and Schurter, K. (2020) "Estimates of derivatives of (log) densities and related objects" available on arXiv. An advantage of this approach relative to alternative kernel-based estimators for the density is that the estimated density, defined as exp() of the log-density, is continuous and nonnegative whenever the kernel function is nonnegative and continuous, even when the density is estimated near the boundary of the support of the data. See the paper for details.

Estimates are stored in the matrix e(L). The number of nonmissing observations used to estimate the log-density and its derivative(s) is returned in the scalar e(N) and the bandwidth is stored in the scalar e(h).

This command uses **integrate** to perform numerical integration. Please run ssc install integrate, replace to install the latest version of **integrate**. You may also need to run integrate, installmata to finish the installation (and make sure the directory in which the command attempts to save integrate.mlib exists and is writeable).

Options

Main

x(*namelist***)** is a required *namelist* of values of *varname* at which estimates of the logarithm of the density and its derivative(s) will be returned. The option can be specified as a variable name or a row matrix. Any missing entries of the variable or matrix will be silently dropped before computing the estimates, i.e. the stored matrix e(L) will have as many rows as there are nonmissing observations/elements of x.

h(*#***)** is the bandwidth that will be used in the estimation.

kernel**(***namelist***)** the kernel function that will be used to estimate the log-density. The default is the *epanechnikov* kernel. The other options are *gaussian*, *quartic*, *triweight*, *triangle*, *uniform*, and *cosinus*.

degree**(***#***)** specifies the order of the local polynomial approximation to the log-density that the estimation will use. The default is to use a local linear approximation (**degree(1)**).

minx(*#*), **ma**xx(*#*) specify the support of the variable *varname*. The default assumes the support is the entire real line. If the support is bounded, these options must be changed in order for the estimates to be consistent at the boundaries.

g(*string***)**, **dg(***string***)** specify the function and its derivative that will be used to estimate the derivative(s) of the log-density. The function g must return a vector of length *degree* and it must be zero outside of the compact interval [-zl,zr], where zl = min((x-minx)/h,1) and zr = min((maxx-x)/h,1). The function should be differentiable and dg should return its derivative. The default value of g is a function of the function whose *j*-th element is equal to g(u,zl,zr,S)=(u+zl)^j * (u-zr) for j=1,...,S and dg is the corresponding derivative with respect to u. The user can replace these defaults by defining Mata functions called g2 and dg2 that accept the same arguments (u,zl,zr,S). See the above paper on arXiv for a discussion of alternative choices.

Remarks

This estimation method is relatively customizable compared to alternative kernel-based density estimators. The current version of this Stata command accepts arbitrary choices of the function g, but does not directly support alternative choices of the kernel function. Users can modify the Mata function *kernel_f* and the list of available kernels in the Stata program if they wish to explore these alternatives. All Mata and Stata code is in logdensity.ado.

Examples

```
. preserve
. drop _all
. matrix x = (0,0.1,0.2)
. set obs 1000
. gen A = rchi2(2)
. logdensity A, x(x) h(0.5) minx(0)
. logdensity A, x(x) h(0.5) minx(0) degree(2)
. mata
function g2(u,zl,zr,S)
{
power = (1::S)'
output = (((colnonmissing(power))'*(u:+zl)'):^power')':*((u:-zr):^2)
return(output)
}
mata mosave g2(), replace
```

```
function dg2(u,zl,zr,S)
{
power = (1::S)'
output =
    (((colnonmissing(power:-1))'*(u:+zl)'):^(power:-1)')':*((u:-zr):^2)
    > :*power +
    (((colnonmissing(power:-1))'*(u:+zl)'):^(power)')':*(2*(u:-zr))
return(output)
}

mata mosave dg2(), replace
end
. logdensity A, x(x) h(0.5) minx(0) g(g2) dg(dg2)
. restore
```

Stored results

Scalars

e(N)	The number of nonmissing observations used in estimation
e(h)	The bandwidth used in estimation

Matrices

e(L)	The estimated log-density and its derivatives

Contact

Karl Schurter kschurter@psu.edu

References

Pinkse, J. and Schurter, K. 2020. Estimates of derivatives of (log) densities and related objects.