The Stata Journal (2004)
4, Number 2, pp. 103–114

Confidence intervals for kernel density estimation

Carlo V. Fiorio
London School of Economics and STICERD

Abstract. This article describes `asciker` and `bsciker`, two programs that enrich the possibility for density analysis using Stata. `asciker` and `bsciker` compute asymptotic and bootstrap confidence intervals for kernel density estimation, respectively, based on the theory of kernel density confidence intervals estimation developed in Hall (1992b) and Horowitz (2001). `asciker` and `bsciker` allow several options and are compatible with Stata 7 and Stata 8, using the appropriate graphics engine under both versions.

Keywords: st0000, kernel density asymptotic confidence intervals, kernel density bootstrap confidence intervals, asymptotic bias, asciker, bsciker, vkdensity

1 Overview

Nonparametric density estimation using Stata can be performed with the official program `kdensity`; see [R] **kdensity**. Some extensions have been provided in Salgado-Ugarte, Shimizu, and Taniuchi (1993); Salgado-Ugarte and Pérez-Hernández (2003); and Van Kerm (2003), which are mostly oriented to develop variable bandwidth kernel density estimation. However, little attention has been paid to performing inference on kernel density estimation. One exception is the recent `akdensity` program presented in Van Kerm (2003) that allows one to compute variability bands as an approximation to confidence intervals. The present article describes `asciker` and `bsciker`, two programs that enrich the possibility for density analysis using Stata. `asciker` and `bsciker` compute asymptotic and bootstrap confidence intervals for kernel density estimation, respectively, based on the theory of kernel density confidence intervals estimation developed in Hall (1992b) and Horowitz (2001). `asciker` and `bsciker` allow several options and are compatible with Stata 7 and Stata 8, using the appropriate graphics engine under both versions.

2 Performing inference on pointwise density estimation

The kernel methodology aims to estimate the density f of a random variable, X, from a random sample $X_i, i = 1, 2, \ldots, n$ without assuming that f belongs to a known family of functions. The (fixed width) kernel density estimation basically slides a window of given width along the data range, counting and properly weighting the observations that fall into the window. Formally, the kernel estimator of f is

$$\widehat{f}_n(x) = \frac{1}{nh_n} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h_n}\right)$$

where K is a kernel function with given properties; $h_n, n = 1, 2, \ldots, n$ is a positive sequence of bandwidths, which depends on the number of the observations in the sample[1]. The density f is assumed to have $r \geq 2$ continuous derivatives in the neighborhood of x (Silverman 1986; Hall 1992a). The bandwidth h_n is sometimes referred to as the *smoothing parameter* since a larger bandwidth makes the estimate smoother and vice versa.

To perform inference on a density function, $f(x)$, we need an asymptotically pivotal statistic, provided suitable estimators for the variance, $\sigma_n(x)$, and the bias, $b_n(x)$, are available.

It can be shown that if nh^{2r+1} is bounded and $n \to \infty$:

$$Z_n(x) \equiv \frac{\widehat{f}_n - f(x) - b_n(x)}{\sigma_n(x)} = \frac{\widehat{f}_n(x) - E\{\widehat{f}_n(x)\}}{\sigma_n(x)} \xrightarrow{d} N(0,1) \tag{1}$$

Hence, (1) could be used to perform inference on the true density, provided the bias and variance of $\widehat{f}_n(x)$ were known.

Whenever we compute a density estimate, $\widehat{f}_n(x)$, we wish it to approximate the true density, $f(x)$, as rapidly as possible. The fastest possible rate of convergence of $\widehat{f}_n(x)$ to $f(x)$ is obtained by setting the bandwidth proportional to a particular power of the sample size ($h_n \propto n^{-1/(2r+1)}$). With such a bandwidth, (a) the difference between the true and the estimated density is never bigger than $n^{-r/(2r+1)}$ ($\widehat{f}_n - f(x) = O_p[n^{-r/(2r+1)}]$); (b) the bias becomes negligible as the sample size increases ($b_n(x) \propto n^{-r/(2r+1)}$); and (c) the variance collapses to zero as sample size becomes larger ($\sigma_n(x) \propto n^{-r/(2r+1)}$) (Horowitz 2001).

Since the true variance, $\sigma_n^2(x)$, is generally unknown in a nonparametric density problem, we need to find a studentized statistic that is asymptotically pivotal.

The variance of $\widehat{f}_n$ equals

$$\begin{aligned}
\sigma_n^2(x) &= \frac{1}{n^2}\sum_{i=1}^{n} \mathrm{Var}\left\{\frac{1}{h_n}K\left(\frac{x-X_i}{h_n}\right)\right\} \\
&= \frac{1}{n}\frac{1}{h_n^2}EK\left(\frac{x-X_i}{h_n}\right)^2 - \frac{1}{n}\left\{\frac{1}{h_n}EK\left(\frac{x-X_i}{h_n}\right)\right\}^2 \\
&\simeq \frac{1}{n}\frac{1}{h_n^2}\int K\left(\frac{x-z}{h_n}\right)^2 f(z)dz - \frac{1}{n}f(x)^2 \qquad (2) \\
&\simeq \frac{f(x)}{nh_n}\int K(u)^2 du \qquad (3)
\end{aligned}$$

In particular, (2) is only an approximation of the previous line because it comes from a first-order Taylor approximation; (3) comes from a change of variable and the

[1] The subscript n is retained for variables that depend on the sample size.

fact that n^{-1} is of smaller order than $(nh_n)^{-1}$ when $h_n \to 0$ as $n \to \infty$. It should also be noted that (2) is the finite-sample variance of $\widehat{f}_n$, while (3) is the variance of the asymptotic distribution of $\widehat{f}_n$.

To perform inference on kernel density estimation, a sample analog of (2) instead of a sample analog of (3) is used since the asymptotic expansion required to obtain asymptotic refinements is simpler if σ_n^2 is estimated by a sample analog of the finite-sample variance (Horowitz 2001). A sample analog of the exact finite-sample variance of $\widehat{f}_n(x)$ is provided by (Hall 1992b, 678):

$$s_n^2(x) = \frac{1}{(nh_n)^2} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h_n}\right)^2 - \frac{\widehat{f}_n(x)^2}{n} \tag{4}$$

Hence, a studentized form of $Z_n(x)$ for asymptotic confidence interval is defined by

$$t_n(x) = \frac{\widehat{f}_n(x) - E\{\widehat{f}_n(x)\}}{s_n(x)} \tag{5}$$

However, it is important to notice that t_n is the asymptotic t statistic for testing the hypothesis or forming the confidence interval for $E\{\widehat{f}_n(x)\}$ but cannot be used to test the hypothesis and build confidence intervals for $f(x)$, unless $b_n(x)$ is negligibly small. A bias that is not asymptotically converging to zero causes the asymptotic distribution of t_n, with $f(x)$ replacing $E\{\widehat{f}_n(x)\}$ and not to be centered at 0 (Horowitz 2001).

The asymptotic bias is a characteristic of nonparametric estimators, such as the kernel density estimation, and it cannot be overcome with the use of the bootstrap. Let $X_i^*, i = 1, 2, \ldots, n$ be the bootstrap sample obtained sampling the data X_i with replacement. Then, the bootstrap estimator of f is

$$\widehat{f}_n^* = \frac{1}{nh_n} \sum_{i=1}^{n} K\left(\frac{x - X_i^*}{h_n}\right) \tag{6}$$

and the bootstrap analog of $s_n^2(x)$ is

$$s_n^{2*}(x) = \frac{1}{nh_n^2} \sum_{i=1}^{n} K\left(\frac{x - X_i^*}{h_n}\right)^2 - \frac{\widehat{f}_n^*(x)^2}{n}$$

The bootstrap analog of t_n is

$$t_n^* = \frac{\widehat{f}_n^*(x) - \widehat{f}_n(x)}{s_n^*(x)} \tag{7}$$

From (6), we can see that $\widehat{f}_n^*(x)$ is an unbiased estimator of $\widehat{f}_n(x)$, though $\widehat{f}_n(x)$ is a biased estimator of $f(x)$. Hence, t_n^* is a bootstrap t-statistic for forming a confidence interval for $E\{\widehat{f}_n(x)\}$. For instance, for the symmetric two-sided confidence interval with coverage probability $1-\alpha$, the upper limit is given by

$$E\{\widehat{f}_n(x)\}^U = \widehat{f}_n(x) - s_n(x) \times u^*_{\alpha/2} \tag{8}$$

and the lower limit is given by

$$E\{\widehat{f}_n(x)\}^L = \widehat{f}_n(x) - s_n(x) \times u^*_{1-(\alpha/2)} \tag{9}$$

where, $u^*_{\alpha/2}$ is the bootstrap estimate of the quantile defined by $P(t^*_n \leq u^*_{\alpha/2}) = \alpha/2$ and $u^*_{1-(\alpha/2)}$ is the bootstrap estimate of the quantile defined by $P(t^*_n \leq u^*_{1-(\alpha/2)}) = 1-(\alpha/2)$ (Hall [1992b, 679] and Davidson and MacKinnon [2003, chapter 5]). However, t^*_n can be used to form a confidence interval for $f(x)$ only if the bias $b_n(x)$ is negligible.

2.1 Methods for controlling the asymptotic bias

There are two main methods for dealing with asymptotic bias:

- explicit bias removal
- undersmoothing

Regardless of the method used to remove asymptotic bias, forming a confidence interval requires using a bandwidth sequence that converges more rapidly than the one that maximizes the rate of convergence of a point estimator of $f(x)$. "Nonparametric point estimation and nonparametric interval estimation or testing of hypothesis are different tasks that require different degrees of smoothing" (Horowitz 2001, 3199). Hall (1992b) shows that undersmoothing performs better in terms of errors in the coverage probability and suggests setting $h_n \propto \gamma n^{1/(2r+1)}$, with $0 < \gamma < 1$. Horowitz (2001) suggests setting $h_n \propto n^{-\kappa}$, with $\kappa > 1/(2r+1)$. In other words, to compute the confidence interval of a nonparametric density estimation, we need to use a smaller bandwidth than the one chosen to compute the density estimation. Reducing the bandwidth (*undersmoothing*) will make the bias converge to zero more rapidly and the statistic $t_n(x)$ asymptotically centered at 0.

Hence, with undersmoothing, the bias is identically zero, and the statistic t_n in (5) becomes

$$t_n^{\mathrm{us}}(x) = \frac{\widehat{f}_n^{\mathrm{us}}(x) - f(x)}{s_n^{\mathrm{us}}(x)} \tag{10}$$

where "us" stands for undersmoothed estimate. Such a statistic can then be used to compute confidence intervals about the true distribution $f(x)$. The same reasoning applies to the bootstrap confidence intervals: with undersmoothing, the statistic (7) is an unbiased estimator of $\widehat{f}_n^{\mathrm{us}}(x)$, which itself is an unbiased estimator of $f(x)$. Hence, upper and lower limits in (8) and (9) are respectively replaced by

$$f(x)^U = \widehat{f}_n^{\mathrm{us}}(x) - s_n^{\mathrm{us}}(x) \times u^{*\mathrm{us}}_{\alpha/2}$$

$$f(x)^L = \widehat{f}_n^{\text{us}}(x) - s_n^{\text{us}}(x) \times u_{1-(\alpha/2)}^{*\text{us}}$$

where $u_{\alpha/2}^{*\text{us}}$ and $u_{1-(\alpha/2)}^{*\text{us}}$ are computed as previously explained from $t_n^{*\text{us}}$, the undersmoothed version of t_n^*.

2.2 Asymptotic versus bootstrap confidence intervals

Although the asymptotic confidence interval can be computed using (10), Horowitz (2001) demonstrates that the bootstrap provides asymptotic refinements for tests of hypothesis and confidence intervals in nonparametric density estimation. With asymptotic critical values, the difference between the true and nominal rejection probabilities of a symmetrical t test is $O\{(nh_n)^{-1}\}$, provided that $nh_n^{r+1} \rightarrow 0$. If the latter condition is not verified, the error in rejection probability is greater than $O\{(nh_n)^{-1}\}$. With the bootstrap critical values, the difference between the true and the nominal rejection probabilities of the symmetrical t test is $o\{(nh_n)^{-1}\}$. Hence, with undersmoothing, the bootstrap provides asymptotic refinements for hypothesis tests and confidence intervals based on a kernel nonparametric density estimator.

3 Implementation notes

Both `asciker` and `bsciker` are packaged in two modules. They both make use of `vkdensity`, an enhanced version of [R] **kdensity**. In addition to the latter condition, `vkdensity` estimates the variance of the kernel as in (4) and it allows undersmoothing, as well as oversmoothing. The undersmoothing performed is as described in Horowitz (2001), with $h_n \propto n^{-\kappa}$, $\kappa > 1/(2r+1)$. `vkdensity` also allows the choice between three different optimal bandwidth estimators: Scott (1992), Härdle (1991), and Silverman (1986) (see also `bandw` in Salgado-Ugarte et al. 1995b); as well as the possibility of a user-defined bandwidth.

`asciker` is conceptually similar to `akdensity` developed in Van Kerm (2003). Its main improvement is that it allows for computation of the actual confidence interval, not only variability bands, reducing the relevance of the bias by undersmoothing.

The structure of `bsciker` is more complex. `bsciker` develops in three steps:

1. It generates B bootstrap samples (random sample with replacement) from the original dataset.
2. It computes the kernel density and its variance for each bootstrap dataset using `vkdensity` *with undersmoothing*.
3. It merges results from previous steps, computes the pivotal statistic, and computes the relevant bootstrap critical values to form upper and lower bounds of the kernel estimation confidence interval.

A methodological issue arises here concerning the correct degree of undersmoothing, which is connected with the choice of the optimal bandwidth. Many “optimal” bandwidth estimators assume that the underlying distribution is normal, although in many cases, nonparametric density estimation is precisely adopted because the underlying distribution may depart significantly from normality. Using `asciker` and `bsciker`, we assume that the search for the correct bandwidth has been performed beforehand[2]. If we believe that the optimal bandwidth is reasonably determined using one of the optimal bandwidth algorithms implemented in `asciker` and `bsciker`, the commands can be applied straightforwardly, and only the degree of undersmoothing remains to be chosen.

3.1 Bootstrapping weighted samples

`bsciker` can also be used with `aweight` and `fweight`. Let's assume that the dataset is made of N observations and each observation has a weight attached to it. `bsciker` first expands the dataset so that all observations have a weight equal 1 and then extracts a bootstrap sample of dimension N from it. For instance, if all the weights were 1 or 2, then `bsciker` will create an unweighted dataset where observations with weight 2 are included twice. `bsciker` allows such weights to be noninteger[3], however more complex sampling weights are not implemented. Some caution should be used for very large datasets and very large frequency weights so that the maximum number of observations that Stata allows is not exceeded.

4 Syntax

The syntax of `asciker` partly mimics the syntax of the official [R] **kdensity**, which uses fixed kernel estimation methods:

```
asciker varname [weight] [if exp] [in range] [, nograph
   generate(newvarx newvard newvarb) at(varx) usmooth(#) [epan|gauss]
   [scott|hardle|silver] mbandw(#) n(#) percent(#) gr7 graph_options]
```

Most options for `asciker` are the same as those for [R] **kdensity**. The specific options are the following:

`generate`(*newvarx newvard newvarb*) creates four new variables: *newvarx* will contain the points of estimation; *newvard* will contain the density estimation; and *newvarb_u* and *newvarb_l* will contain the upper and lower bound confidence interval variable.

[2] For a discussion about the choice of bandwidth, see Silverman (1986) among others.

[3] In particular, `bsciker` renormalizes the weights so that the smallest value is 1. Hence, it replaces these weights with their rounded-to-integer versions. Although rough, such an approximation is often reasonable for most datasets. I was first suggested this solution by William Gould with reference to a similar problem. Of course, I bear all the responsibility for the implementation of this solution in `bsciker`. For more details, see the Statalist archive at
www.hsph.harvard.edu/cgi-bin/lwgate/STATALIST/archives/statalist.0201/Subject/article-39.html.

`usmooth(#)` chooses the degree of undersmoothing for confidence interval estimation, i.e., the parameter κ as in section 3. The default value is 1/4. Increasing this number will result in a smaller bandwidth, i.e., a more variable and less biased estimation, and vice versa.

`scott`, `hardle`, and `silver` allow you to choose between three different optimal bandwidths, i.e., those proposed by Scott (1992), Härdle (1991), and Silverman (1986). See also `bandw` (Salgado-Ugarte, Shimizu, and Taniuchi 1995b; 1995a; 1993).

`mbandw(#)` specifies the bandwidth manually.

`percent(#)` specifies the coverage probability of the confidence interval. The default value is set at 95 (meaning 95% coverage probability), but it can be changed at will.

`gr7` computes the graph using Stata 7 graph facilities (not as nice but sometimes quicker).

The syntax of `bsciker` is

`bsciker` *varname* [*weight*] [`if` *exp*] [`in` *range*] [`, nograph`
`generate(`*newvarx newvard newvarb*`) at(`*varx*`) usmooth(#)` [`epan`|`gauss`]
[`scott`|`hardle`|`silver`] `mbandw(#) n(#) bsrepl(#) seed(#) bsppts(#)`
[`up(#) lp(#)`|`percent(#)`] `gr7` *graph_options*]

The main differences with respect to the syntax of `asciker` are

`bsrepl(#)` specifies the number of bootstrap replications to compute. The default value is 99, which is good for a first investigation. However, for a final estimation and small datasets, it often should be much larger than 99. For instance, in section 5, we suggest using 999 bootstrap replications. Of course, computation speed will dramatically reduce with such a number of bootstrap replications.

`bsppts(#)` specifies the percentage of estimation points at which to compute the confidence interval. This option is particularly useful for estimation with many data points, which would make the confidence interval very lengthy. The default value is 100%.

`seed(#)` sets the seed of the bootstrap resampling for replication purposes.

`up(#)` and `lp(#)` allow one to specify the upper and lower percentiles to be computed for the confidence interval. The default value are set to 97.5% and 2.5%, so as to obtain a 95% confidence interval, but these values can be changed at will. If `up()`, `lp()`, and `percent()` are specified, `up()` and `lp()` are ignored.

`percent(#)` specifies the coverage probability of the confidence interval. The default value is set at 95 (meaning 95% coverage probability), but it can be changed at will. If `up()`, `lp()`, and `percent()` are specified, `up()` and `lp()` are ignored.

The syntax of `vkdensity` is

`vkdensity` *varname* [*weight*] [`if` *exp*] [`in` *range*] [`,` `nograph`
`generate(`*newvarx newvard newvarv*`)` `at(`*varx*`)` `usmooth(`#`)` [`epan`|`gauss`]
[`scott`|`hardle`|`silver`] `mbandw(`#`)` `n(`#`)` *graph_options*]

There is only one command that differs from `asciker`:

`generate`(*newvarx newvard newvarv*), where *newvarx* contains the points of estimation, *newvard* the pointwise density estimate, and *newvarv* the variance of the pointwise estimate.

5 Examples

As a simple illustration, we simulated some random samples from an *ad hoc* bimodal distribution, a mixture of two normal distributions with equal variance and different mean:

$$\Pi = \frac{9}{20}N(0,1/2) + \frac{11}{20}N(2,1/2)$$

We simulated two samples of different size from Π: $n = 50$ and $n = 1000$. We named the simulated samples x_50 and and x_1000, respectively. We run `asciker` on x_50 as follows:

```
. asciker x_50, generate(x2a y2a b2a) usmooth(.25) nograph n(50) silver
Note: this program requires installation of vkdensity.ado!
significance level: 5%
bandwidth choice (Silverman)=   0.52963
(oversmoothed) bandwidth choice (Silverman)=   0.43554
```

The oversmoothing parameter, κ, is set to 1/4. The program reminds us that `vkdensity.ado` is necessary and shows the significance level of the confidence interval and the estimated optimal bandwidth. The confidence interval is set at 95%.

We then run `bsciker` on the same sample, setting the number of bootstrap replications to 999, as follows:

```
. bsciker x_50, generate(x2b y2b b2b) usmooth(.25) bsrepl(999) n(50) silver
Note: this program requires installation of vkdensity.ado!
Lower percentile:2.5%
Upper percentile:97.5%
bandwidth choice (Silverman)=   0.52963
(oversmoothed) bandwidth choice (Silverman)=   0.43554
Bootstrap samples are being generated
...   (output omitted)   ...
vkdensity on bootstrap samples being computed: be patient, please
...   (output omitted)   ...
Merging all vkdensity on bootstrapped samples
...   (output omitted)   ...
Generating statistic tstar
Computing percentiles for cstar
  (output omitted)
```

The graphs are then combined and overlapped with the true distribution to obtain figure 1. Analogous steps were undertaken for the sample `x_1000`, leading to figure 2. The main results from these and analogous simulations performed are that kernel density estimations must be handled with great care when sample size is relatively small, as confidence intervals can be very wide (figure 1). However, even with a relatively larger sample size, confidence intervals can be highly informative. As for the comparison between asymptotic and bootstrap confidence intervals, the former tends to be smoother and narrower, though the answer both provide is in many cases rather similar. The main drawback of `bsciker` is that it takes much longer to compute than `asciker`, especially the larger the dataset and the number of bootstrap simulations required.

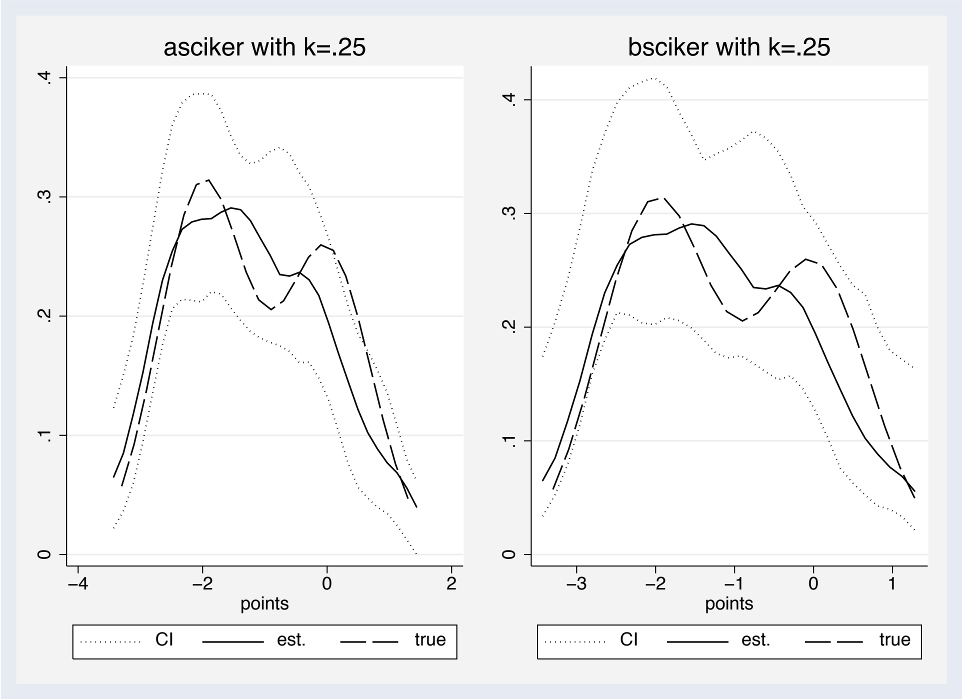

Figure 1: 95% asymptotic and bootstrap confidence intervals for `kdensity`. $n = 50$

(*Continued on next page*)

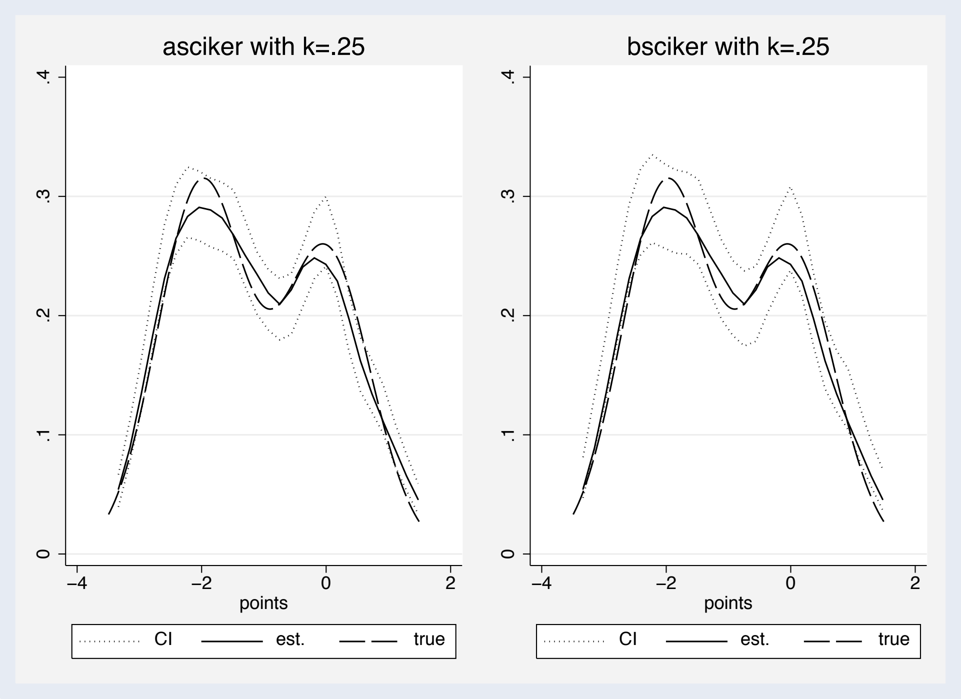

Figure 2: 95% asymptotic and bootstrap confidence intervals for `kdensity`. $n = 1000$

As a further illustration of `bsciker` and `asciker`, I use the coral trout length data presented in [R] **kdensity**, Salgado-Ugarte et al. (1993), and Van Kerm (2003). The data consist of 316 length observations of coral trout (in mm) and can be downloaded from the Stata Press web site. As figure 3 shows, confidence intervals are also useful in considering the hypothesis of multimodality of the underlying distribution. As discussed in Van Kerm (2003), fixed-bandwidth kernel density estimation of these data tends to oversmooth the estimation of the underlying distribution. The solution suggested thereby is to use adaptive kernel density estimation; such a method allows us to clearly detect the two main modes of the distribution. Estimating the confidence intervals of the fixed bandwidth kernel density, we can provide additional evidence to the hypothesis of two main modes at around 350 and 420. We can also put forward the hypothesis that the distribution presents two additional minor modes at around 250 and 500. Clearly, such a hypothesis needs to be tested since the pointwise confidence intervals are nothing but a measure of how uncertain the estimation is at each estimation point. For tests of the number of modes, see, for instance, Silverman (1986, 146)

```
. use http://www.stata-press.com/data/r7/trocolen.dta, clear
. asciker length, generate(x2a y2a b2a) usmooth(.25) nograph n(50)
  (output omitted)
. graph twoway line b2a_u y2a b2a_l x2a, scheme(sj)
. graph save length_a, replace
. bsciker length, generate(x2b y2b b2b) usmooth(.25) bsrepl(999) nograph n(50)
  (output omitted)
. graph twoway line b2b_u y2b b2b_l x2b, scheme(sj)
. graph save length_b, replace
. graph combine length_a.gph length_b.gph
```

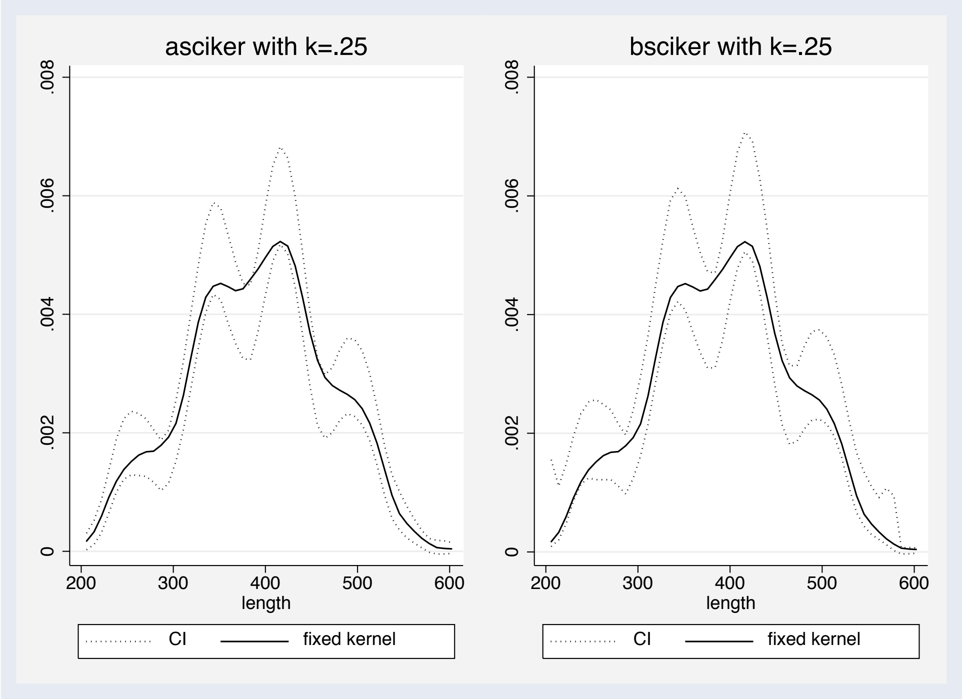

Figure 3: 95% asymptotic and bootstrap confidence intervals for coral-trout-length data

6 Acknowledgments

I developed the asciker and bsciker programs as part of my Ph.D. dissertation under the supervision of Frank Cowell and Vassilis Hajivassiliou, whose suggestions are gratefully acknowledged. I also benefited from the 9th UK Stata Users Group meeting held in London, 19–20 May 2003, and, in particular, from discussions with Philippe Van Kerm. I thank the many contributors of the Statalist and, in particular, William Gould, for the prompt and useful suggestions received. Last but not least, I thank Joe Newton and an anonymous referee for the very useful comments provided on previous versions of this paper.

7 References

Davidson, R. and J. G. MacKinnon. 2003. *Econometric Theory and Methods*. New York: Oxford University Press.

Hall, P. 1992a. *The Bootstrap and Edgeworth Expansion*. New York: Springer.

—. 1992b. Effect of bias estimation on coverage accuracy of bootstrap confidence intervals for a probability density. *Annals of Statistics* 20: 675–694.

Härdle, W. 1991. *Smoothing Techniques. With implementation in S*. New York: Springer.

Horowitz, J. L. 2001. The bootstrap. In *Handbook of Econometrics*, ed. J. J. Heckman and E. Leamer, vol. 5, 3159–3228. Amsterdam: North-Holland.

Salgado-Ugarte, I. H. and M. A. Pérez-Hernández. 2003. Exploring the use of variable bandwidth kernel density estimators. *Stata Journal* 3(2): 133–147.

Salgado-Ugarte, I. H., M. Shimizu, and T. Taniuchi. 1993. snp6: Exploring the shape of univariate data using kernel density estimators. *Stata Technical Bulletin* 16: 8–19. In *Stata Technical Bulletin Reprints*, vol. 3, 155−173. College Station, TX: Stata Press.

—. 1995a. snp6.1: ASH, WARPing, and kernel density estimation for univariate data. *Stata Technical Bulletin* 26: 23–31. In *Stata Technical Bulletin Reprints*, vol. 5, 161−172. College Station, TX: Stata Press.

—. 1995b. snp6.2: Practical rules for bandwidth selection in univariate density estimation. *Stata Technical Bulletin* 27: 5–19. In *Stata Technical Bulletin Reprints*, vol. 5, 172−190. College Station, TX: Stata Press.

Scott, D. W. 1992. *Multivariate Density Estimation*. New York: John Wiley & Sons.

Silverman, B. W. 1986. *Density Estimation for Statistics and Data Analysis*. London: Chapman & Hall.

Van Kerm, P. 2003. Adaptive kernel density estimation. *Stata Journal* 3(2): 148–156.

About the Author

Carlo V. Fiorio is Ph.D. student in Economics at the London School of Economics, UK. His main research interests are in the area of applied econometrics and public economics.