scdensity: a program for self-consistent density estimation

Joerg Luedicke

Yale University & University of Florida

Stata Conference, San Diego, CA – July 26-27, 2012
Outline

1. Non-parametric density estimation
Outline

1. Non-parametric density estimation
2. Self-consistent method
Outline

1. Non-parametric density estimation
2. Self-consistent method
3. scdensity: the program
Outline

1. Non-parametric density estimation
2. Self-consistent method
3. scdensity: the program
4. Monte-Carlo simulations
1. Non-parametric density estimation

2. Self-consistent method

3. scdensity: the program

4. Monte-Carlo simulations

5. Conclusion

Outline

1. Non-parametric density estimation
2. Self-consistent method
3. scdensity: the program
4. Monte-Carlo simulations
5. Conclusion
Outline

1. Non-parametric density estimation
2. Self-consistent method
3. scdensity: the program
4. Monte-Carlo simulations
5. Conclusion
6. Outlook
Outline

1. Non-parametric density estimation
2. Self-consistent method
3. scdensity: the program
4. Monte-Carlo simulations
5. Conclusion
6. Outlook
7. References
1. Non-parametric density estimation
2. Self-consistent method
3. scdensity: the program
4. Monte-Carlo simulations
5. Conclusion
6. Outlook
7. References
8. Appendix
Non-parametric density estimation

- **Histogram**
  - Probably most commonly used method for estimating a probability density function
Non-parametric density estimation

- **Histogram**
  - Probably most commonly used method for estimating a probability density function
  - Origin and binwidth need to be determined a-priori
Non-parametric density estimation

- **Histogram**
  - Probably most commonly used method for estimating a probability density function
  - Origin and binwidth need to be determined a-priori

- **Kernel density estimation**
  - Another very popular method for density estimation
Non-parametric density estimation

- **Histogram**
  - Probably most commonly used method for estimating a probability density function
  - Origin and binwidth need to be determined a-priori

- **Kernel density estimation**
  - Another very popular method for density estimation
  - Requires the choice of the kernel function (less important)
Non-parametric density estimation

- Histogram
  - Probably most commonly used method for estimating a probability density function
  - Origin and binwidth need to be determined a-priori

- Kernel density estimation
  - Another very popular method for density estimation
  - Requires the choice of the kernel function (less important)
  - And the smoothing parameter (aka bandwidth or window width)
Non-parametric density estimation

- **Histogram**
  - Probably most commonly used method for estimating a probability density function
  - Origin and binwidth need to be determined a-priori

- **Kernel density estimation**
  - Another very popular method for density estimation
  - Requires the choice of the kernel function (less important)
  - And the smoothing parameter (aka bandwidth or window width)
  - Smoothing parameters: trade-off between bias and variance
Histograms with different binwidths

Variable: bpsysiol from dataset sheas2.dta (webuse sheas2)

Self-consistent density estimation

Non-parametric density estimation

Self-consistent method

scdensity: the program

Monte-Carlo simulations

Conclusion

Outlook

References

Appendix
Kernel density estimates (Epanechnikov) with different bandwidths

Variable -systolic- from dataset manes2.dta (webuse manes2); N=10,351; the bandwidth in graph c) is derived by Stata's default bandwidth rule of thumb.
Remember the classical kernel density estimator:

\[ \hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{N} K\left( \frac{x - X_i}{h} \right) \] (1)
Self-Consistent Density Estimation

- Remember the classical kernel density estimator:
  \[ \hat{f}(x) = \frac{1}{n h} \sum_{i=1}^{N} K\left(\frac{x - X_i}{h}\right) \]  
  (1)

- The self-consistent estimate can be written as:
  \[ \hat{f}(x) = \frac{1}{n} \sum_{i=1}^{N} K(x - X_i) \]  
  (2)
Self-Consistent Density Estimation

- Remember the classical kernel density estimator:

\[ \hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{N} K\left(\frac{x - X_i}{h}\right) \]  (1)

- The self-consistent estimate can be written as:

\[ \hat{f}(x) = \frac{1}{n} \sum_{i=1}^{N} K(x - X_i) \]  (2)

- The basic idea of the self-consistent method is *not* to search for an optimal bandwidth, given an arbitrary kernel function...
Remember the classical kernel density estimator:

\[ \hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{N} K\left(\frac{x - X_i}{h}\right) \]  

(1)

The self-consistent estimate can be written as:

\[ \hat{f}(x) = \frac{1}{n} \sum_{i=1}^{N} K(x - X_i) \]  

(2)

The basic idea of the self-consistent method is *not* to search for an optimal bandwidth, given an arbitrary kernel function...

...but to find an optimal shape of the kernel, given the data.
Self-Consistent Density Estimation

- Remember the classical kernel density estimator:

\[
\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{N} K\left(\frac{x - X_i}{h}\right)
\]  

(1)

- The self-consistent estimate can be written as:

\[
\hat{f}(x) = \frac{1}{n} \sum_{i=1}^{N} K(x - X_i)
\]  

(2)

- The basic idea of the self-consistent method is *not* to search for an optimal bandwidth, given an arbitrary kernel function...
- ...but to find an optimal shape of the kernel, given the data.
- No parameters need to be fixed beforehand.
scdensity: the program

**Syntax**

```
scdensity varname [if] [in]  
[ , generate(newvar1 [newvar2])  
n(#) range(# #)  
nograph name(name [, replace]) ]
```
scdensity: the program

Syntax

```bash
scdensity varname [if] [in] [ , generate(newvar1 [newvar2]) n(#) range(# #) nograph name(name [, replace]) ]
```

- scdensity is available from SSC: `ssc install scdensity`
- `help scdensity` for further information
Monte Carlo simulations

- Experimental set-up
  - Four test densities.
Monte Carlo simulations

- Experimental set-up
  - Four test densities.
  - MISE as measure of estimation accuracy:
  - $MISE(\hat{f}) = E \int \{\hat{f}(x) - f(x)\}^2 dx$ [Silverman, 1998]
Monte Carlo simulations

- Experimental set-up
  - Four test densities.
  - MISE as measure of estimation accuracy:
    \[ MISE(\hat{f}) = E \int \{ \hat{f}(x) - f(x) \}^2 dx \] [Silverman, 1998]
  - Two kernel functions (Epanechnikov & Gaussian).
Monte Carlo simulations

- **Experimental set-up**
  - Four test densities.
  - MISE as measure of estimation accuracy:
    \[
    MISE(\hat{f}) = E \int \{ \hat{f}(x) - f(x) \}^2 dx \quad \text{[Silverman, 1998]}
    \]
  - Two kernel functions (Epanechnikov & Gaussian).
  - Three fixed bandwidth rules of thumb:
    1. \( h_o = 0.9 \min(\sigma, \text{IQ}/1.349)n^{-\left(\frac{1}{5}\right)} \)
    2. \( h_o = 1.06 \min(\sigma, \text{IQ}/1.349)n^{-\left(\frac{1}{5}\right)} \)
    3. \( h_o \geq 1.144\sigma n^{-\left(\frac{1}{5}\right)} \)
  - See [Silverman, 1998], [Haerdle et al., 2004], [Scott, 1992], respectively.
Monte Carlo simulations

- **Experimental set-up**
  - Four test densities.
  - MISE as measure of estimation accuracy:
    \[ MISE(\hat{f}) = E \int (\hat{f}(x) - f(x))^2 \, dx \] [Silverman, 1998]
  - Two kernel functions (Epanechnikov & Gaussian).
  - Three fixed bandwidth rules of thumb:
    1. \( h_o = 0.9 \min(\sigma, lQ/1.349)n^{-\left(\frac{1}{5}\right)} \)
    2. \( h_o = 1.06 \min(\sigma, lQ/1.349)n^{-\left(\frac{1}{5}\right)} \)
    3. \( h_o \geq 1.144\sigma n^{-\left(\frac{1}{5}\right)} \)
  - See [Silverman, 1998], [Haerdle et al., 2004], [Scott, 1992], respectively.
  - Variable bandwidth estimation (aka adaptive kernel).
Monte Carlo simulations

- Experimental set-up
  - Four test densities.
  - MISE as measure of estimation accuracy:
    \[ MISE(\hat{f}) = E \int \{\hat{f}(x) - f(x)\}^2 dx \] [Silverman, 1998]
  - Two kernel functions (Epanechnikov & Gaussian).
  - Three fixed bandwidth rules of thumb:
    1. \[ h_0 = 0.9 \min(\sigma, \text{IQ}/1.349)n^{-\left(\frac{1}{5}\right)} \]
    2. \[ h_0 = 1.06 \min(\sigma, \text{IQ}/1.349)n^{-\left(\frac{1}{5}\right)} \]
    3. \[ h_0 \geq 1.144\sigma n^{-\left(\frac{1}{5}\right)} \]
  - See [Silverman, 1998], [Haerdle et al., 2004], [Scott, 1992], respectively.
  - Variable bandwidth estimation (aka adaptive kernel).
  - The user written -kdens- (available from SSC [Jann, 2005], [Jann, 2007]) was used for kernel density estimation.
  - The user written -fmm- (SSC, [Deb, 2007]) was used for fitting maximum likelihood mixture models.
Results

- **Abbreviations:**
  - ML = maximum likelihood
  - SCD = self-consistent method
  - EPH2 = Epanechnikov kernel with bandwidth #2 from previous slide
  - GKH1 = Gaussian kernel with bandwidth #1 from previous slide
  - GKH2 = Gaussian kernel with bandwidth #2 from previous slide
  - GKH3 = Gaussian kernel with bandwidth #3 from previous slide
  - ADK = adaptive kernel (Epanechnikov)
Test density a):

\[
\phi(\mu, \sigma^2) = (2\pi)^{-\frac{1}{2}} \sigma^{-1} \exp\left\{-\frac{1}{2} \left( x - \mu \right)^2 / \sigma^2 \right\}
\]
Results for test density a)
Test density \( b \): \( f(x) = \frac{1}{2} \phi(0, 1) + \frac{1}{2} \phi(3, 1) \)
Results for test density b)
Test density $c$): $f(x) = \frac{1}{2} \phi(0, 1) + \frac{1}{2} \phi(5, 2^2)$
Results for test density c)

![Graph showing MISE vs. N for different methods]

Number of random draws: 500
Test density $d$):

\[ f(x) = \frac{1}{2} \phi(0, 1.2^2) + \frac{1}{4} \phi(4, 1.4^2) + \frac{1}{4} \phi(8, 0.6^2) \]
Results for test density d)
Given the test densities and kernel density estimators used in the simulations, the self-consistent method was the most accurate among the nonparametric estimators.
Given the test densities and kernel density estimators used in the simulations, the self-consistent method was the most accurate among the nonparametric estimators.

For one of the test densities \( f(x) = \frac{1}{2} \phi(0, 1) + \frac{1}{2} \phi(3, 1) \) ...

...the self-consistent method performed nearly as well as the (parametric) ML estimate

...without relying on any prior assumptions or parameter fixations.
Conclusion

- Given the test densities and kernel density estimators used in the simulations, the self-consistent method was the most accurate among the nonparametric estimators.
- For one of the test densities \( f(x) = \frac{1}{2} \phi(0, 1) + \frac{1}{2} \phi(3, 1) \)
- ...the self-consistent method performed nearly as well as the (parametric) ML estimate
- ...without relying on any prior assumptions or parameter fixations.
- The question remains: Is it of practical importance?
- Yes, it certainly can be of practical importance. The following figure shows an example:
Comparison of density estimates using real data

Self-consistent density estimation
Non-parametric density estimation
Self-consistent method
scdensity: the program
Monte-Carlo simulations
Conclusion
Outlook
References
Appendix

Variable height from the dataset nhanes2.data (~webuse nhanes2~); N=10,351; graph a): Epanechnikov kernel with bandwidth rule f1, Stata's default.
Program features

- Variance estimation, e.g. for confidence intervals/bands
- Weights
- Grid expansion
Outlook

■ Program features
  ■ Variance estimation, e.g. for confidence intervals/bands
  ■ Weights
  ■ Grid expansion

■ Non- and semiparametric models
  ■ Bivariate density estimation
  ■ Smoothing & regression
References (1)

Alberto Bernacchia & Simone Pigolotti (2011)
Self-consistent method for density estimation.

Partha Deb (2007)
kdens: FMM: Stata module to estimate finite mixture models
Statistical Software Components S456895, Boston College Department of Economics, revised 12 Feb 2012.

Wolfgang Haerdle et al.(2004)
Nonparametric and Semiparametric Models.
Springer, Berlin/Heidelberg.

Ben Jann (2005)
kdens: Stata module for univariate kernel density estimation, available from:
References (2)

- **Ben Jann (2007)**  
  Univariate kernel density estimation.  
  Online publication:  
  [http://fmwww.bc.edu/RePEc/bocode/k/kdens.pdf](http://fmwww.bc.edu/RePEc/bocode/k/kdens.pdf)

- **David W Scott (1992)**  
  Multivariate Density Estimation.  
  Wiley, New York et al.

- **Bernard W Silverman (1992)**  
  Density Estimation for Statistics and Data Analysis,  
  Chapman & Hall/CRC, Boca Raton et al.

- **Geoffrey S Watson & M R Leadbetter (1963)**  
  On the estimation of the probability density.  
I am thankful to Alberto Bernacchia for helpful discussions and sharing his R code.
Thank you!

joerg.luedicke@ufl.edu
Outline of the basic algorithm of the self-consistent estimator (1)

- Departure: an optimal convolution kernel can be derived for known densities [Watson & Leadbetter, 1963]
- The Fourier transform $K_{opt}(t)$ of the optimal kernel $K_{opt}(x)$ equals

$$K_{opt}(t) = \frac{N}{N - 1 + |\omega(t)|^{-2}}$$ (3)

- where $\omega(t)$ is the Fourier transform of the true density $f(x)$
- Then, the Fourier transform of the density estimate in equation (2) is

$$\hat{\omega}(t) = \Delta(t)K_{opt}(t) = \frac{N\Delta(t)}{N - 1 + |\omega(t)|^{-2}}$$ (4)
...where $\Delta(t)$ is the empirical characteristic function

$$\Delta(t) = \frac{1}{N} \sum_{i=1}^{N} \exp(itX_i) \quad (5)$$

$K_{opt}(t)$ is of course only known if the true density is known.

The self-consistent method now uses equation (4) for which the unknown term $\omega$ is replaced with an initial guess $\hat{\omega}_0$,

...which results in the estimate $\hat{\omega}_1$.

Then the improved estimate $\hat{\omega}_2$ is obtained by using a kernel which is optimal for $\hat{\omega}_1$, and so on.
Outline of the basic algorithm of the self-consistent estimator (3)

- This is iterated until a certain point in the sequence

\[ \hat{\omega}_{n+1} = \frac{N\Delta}{N - 1 + |\hat{\omega}_n|^{-2}} \]  

(6)

- ...is reached, for which

\[ \hat{\omega}_{sc} = \frac{N\Delta}{N - 1 + |\hat{\omega}_{sc}|^{-2}} \]  

(7)