Quantitative Economics for the Evaluation of the European Policy

Dipartimento di Economia e Management

Co-funded by the Erasmus+ Programme of the European Union

Project funded by

European Commission Erasmus + Programme –Jean Monnet Action

Project number 553280-EPP-1-2015-1-IT-EPPJMO-MODULE

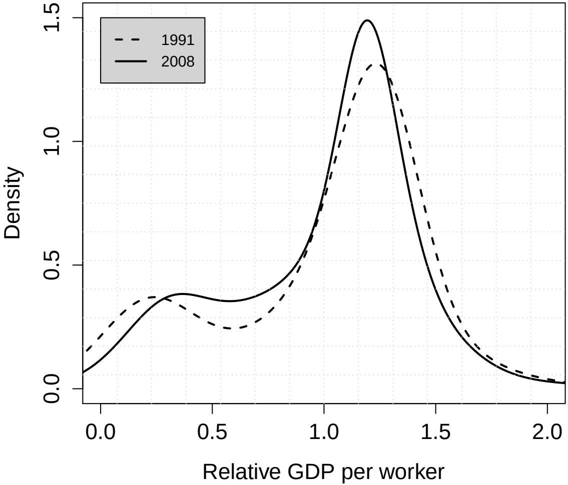
Irene Brunetti Davide Fiaschi Angela Parenti[1]

04/10/2016

[1] ireneb@ec.unipi.it, davide.fiaschi@unipi.it, and aparenti@ec.unipi.it.

Distribution of Regional GDP per Worker

	1991	2008
Gini	0.25	0.23
BIPOL	0.83	0.78



Estimate of The Density Function

Let be x a continuous random variable and f its probability density function (pdf).

The pdf characterizes the distribution of the random variable x since it tells "how x is distributed".

Moreover, from pdf it is possible to calculate the mean and the variance (it they exists) of x and the probability that x takes on values in a given interval.

Histogram

Histograms are nonparametric estimates of an *unknown density function*, $f(x)$, **without assuming any well-known functional form**. In order to build an histogram, you have to:

1. select an origin x_0 and divide the real line into "bin" of binwidth h:

$$B_j = [x_0 + (j-1)h, x_0 + jh],\ j \in \mathbf{Z};$$

Histogram

Histograms are nonparametric estimates of an *unknown density function*, $f(x)$, **without assuming any well-known functional form**. In order to build an histogram, you have to:

1. select an origin x_0 and divide the real line into "bin" of binwidth h:

$$B_j = [x_0 + (j-1)h, x_0 + jh],\ j \in \mathbf{Z};$$

2. count how many observations fall into each bin (n_j for each bin j);

Histogram

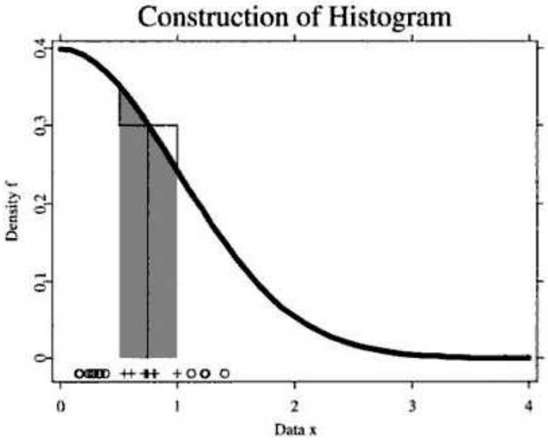
Histograms are nonparametric estimates of an *unknown density function*, $f(x)$, **without assuming any well-known functional form**. In order to build an histogram, you have to:

1. select an origin x_0 and divide the real line into "bin" of binwidth h:

$$B_j = [x_0 + (j-1)h, x_0 + jh], \; j \in \mathbf{Z};$$

2. count how many observations fall into each bin (n_j for each bin j);
3. for each bin divide the frequency by the sample size n and the binwidth h, to get the relative frequencies $f_j = \frac{n_j}{nh}$

Histogram: Cont.



Histogram: Cont.

Crucial parameter: the binwidth h

- A higher binwidth produces smoother estimates

Histogram: Cont.

Crucial parameter: the binwidth h

- A higher binwidth produces smoother estimates
- The estimate is biased and that the bias is positively related to h, while the variance of the estimate is negatively related to h

Histogram: Cont.

Crucial parameter: the binwidth h

- A higher binwidth produces smoother estimates
- The estimate is biased and that the bias is positively related to h, while the variance of the estimate is negatively related to h
- Thus, it is not possible to choose h in order to have a small bias and a small variance

Histogram: Cont.

Crucial parameter: the binwidth h

- A higher binwidth produces smoother estimates
- The estimate is biased and that the bias is positively related to h, while the variance of the estimate is negatively related to h
- Thus, it is not possible to choose h in order to have a small bias and a small variance

$\rightarrow$ we need to find an "optimal" binwidth, which represents an optimal compromise.

Histogram: Cont.

Problems with the histogram:

Histogram: Cont.

Problems with the histogram:

1. each observation x in $[m_j - \frac{h}{2}, m_j + \frac{h}{2})$ is estimated by the same value, $\hat{f}_h(m_j)$, where m_j is the center of the bin;

Histogram: Cont.

Problems with the histogram:

1. each observation x in $[m_j - \frac{h}{2}, m_j + \frac{h}{2})$ is estimated by the same value, $\hat{f}_h(m_j)$, where m_j is the center of the bin;
2. $f(x)$ is estimated using the observations that fall in the interval containing x, and that receive the same weight in the estimation. That is, for $x \in B_j$,

$$\hat{f}_h(m_j) = \frac{1}{nh} \sum_{i=1}^{n} I(X_i \in B_j),$$

where I is the indicator function.

Nonparametric density estimation

- Density estimation is a generalization of the histogram.

Nonparametric density estimation

- Density estimation is a generalization of the histogram.
- It is based on **Kernel functions**: estimate $f(x)$ using the observations that fall into an interval around x, which (typically) receive decreasing weight the further they are from x.

Kernel functions

Consider the *uniform* kernel function, which assigns *the same weight to all observations in an interval* of length $2h$ around observation x, $[x-h, x+h)$:

$$\hat{f}_h(x) = \frac{1}{2nh} \sharp\{X_i \in [x-h, x+h)\}$$

can be obtained by means of a kernel function $K(u)$ such that:

$$K(u) = \frac{1}{2} I(|u| \leq 1)$$

where I is the indicator function and $u = (x - X_i)/h$.

Kernel functions

Consider the *uniform* kernel function, which assigns *the same weight to all observations in an interval* of length $2h$ around observation x, $[x-h, x+h)$:

$$\hat{f}_h(x) = \frac{1}{2nh} \sharp \{X_i \in [x-h, x+h)\}$$

can be obtained by means of a kernel function $K(u)$ such that:

$$K(u) = \frac{1}{2} I(|u| \leq 1)$$

where I is the indicator function and $u = (x - X_i)/h$.

- It assigns weight $1/2$ to each observation X_i whose distance from x, the point where we want to estimate the density, is not bigger than h.

Kernel functions

Consider the *uniform* kernel function, which assigns *the same weight to all observations in an interval* of length $2h$ around observation x, $[x-h, x+h)$:

$$\hat{f}_h(x) = \frac{1}{2nh} \sharp\{X_i \in [x-h, x+h)\}$$

can be obtained by means of a kernel function $K(u)$ such that:

$$K(u) = \frac{1}{2} I(|u| \leq 1)$$

where I is the indicator function and $u = (x - X_i)/h$.

- It assigns weight $1/2$ to each observation X_i whose distance from x, the point where we want to estimate the density, is not bigger than h.
- For each observation that falls into the interval $[x-h, x+h)$ the indicator function takes on value 1

Kernel functions

Consider the *uniform* kernel function, which assigns *the same weight to all observations in an interval* of length $2h$ around observation x, $[x-h, x+h)$:

$$\hat{f}_h(x) = \frac{1}{2nh} \sharp\{X_i \in [x-h, x+h)\}$$

can be obtained by means of a kernel function $K(u)$ such that:

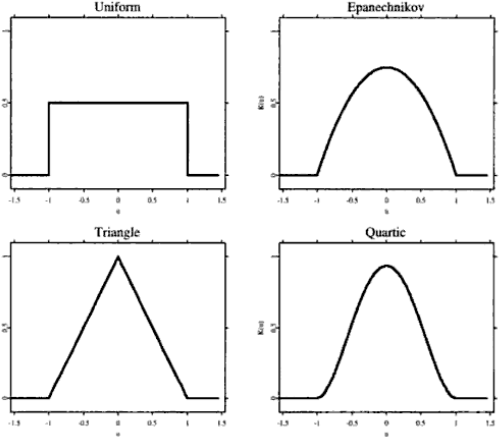
$$K(u) = \frac{1}{2} I(|u| \leq 1)$$

where I is the indicator function and $u = (x - X_i)/h$.

- It assigns weight $1/2$ to each observation X_i whose distance from x, the point where we want to estimate the density, is not bigger than h.
- For each observation that falls into the interval $[x-h, x+h)$ the indicator function takes on value 1
- Each contribution to the function is weighted equally no matter how close the observation X_i is to x

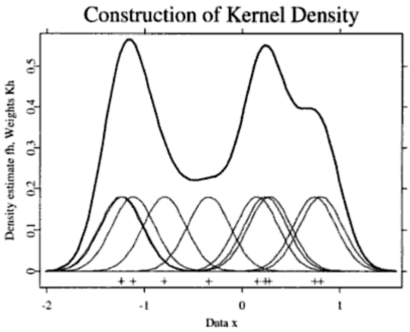
Kernel functions: Cont.

A Kernel function in general (e.g. Epanechnikov, Gaussian, etc), assigns higher weights to observations in $[x - h, x + h)$ closer to x.



Kernel density

A kernel density estimation appears as a sum of bumps: at a given x, the value of $\hat{f}_h(x)$ is found by vertically summing over the "bumps":



$$\hat{f}_h(x) = \sum_{i=1}^{n} \frac{1}{nh} K\left(\frac{x - X_i}{h}\right) = \sum_{i=1}^{n} \frac{1}{n} \underbrace{K_h(x - X_i)}_{\text{"rescaled kernel function"}}$$

Properties of Kernel density estimator

Same problems found for the histogram, that is the bias and the variance depending on h, also hold for the Kernel:

$$Bias\{\hat{f}_h(x)\} = E\{\hat{f}_h(x)\} - f(x);$$

that positively depends on h;

$$Var\{\hat{f}_h(x)\} = Var\left\{\sum_{i=1}^{n} \frac{1}{n} K_h(x - X_i)\right\};$$

that negatively depends on h.

Properties of Kernel density estimator

Same problems found for the histogram, that is the bias and the variance depending on h, also hold for the Kernel:

$$Bias\{\hat{f}_h(x)\} = E\{\hat{f}_h(x)\} - f(x);$$

that positively depends on h;

$$Var\{\hat{f}_h(x)\} = Var\left\{\sum_{i=1}^{n} \frac{1}{n} K_h(x - X_i)\right\};$$

that negatively depends on h.
So, how do we choose h given the trade-off between bias and variance?

Choosing the bandwidth h

(a) Define MSE (mean squared error)

$$MSE\{\hat{f}_h(x)\} = E[\{\hat{f}_h(x) - f(x)\}^2]$$

$$\dots$$

$$MSE\{\hat{f}_h(x)\} = Var\{\hat{f}_h(x)\} + [Bias\{\hat{f}_h(x)\}]^2$$

$\rightarrow$ minimizing MSE may solve the trade-off, but h_{opt} depends on $f(x)$ and $f''(x)$, which are unknown.

Choosing the bandwidth h

(a) Define MSE (mean squared error)

$$MSE\{\hat{f}_h(x)\} = E[\{\hat{f}_h(x) - f(x)\}^2]$$

$$\ldots$$

$$MSE\{\hat{f}_h(x)\} = Var\{\hat{f}_h(x)\} + [Bias\{\hat{f}_h(x)\}]^2$$

$\rightarrow$ minimizing MSE may solve the trade-off, but h_{opt} depends on $f(x)$ and $f''(x)$, which are unknown.

(b) Define MISE (mean integrated squared error), global measure:

$$MISE\{\hat{f}_h(x)\} = E\left[\int_{-\infty}^{\infty} \{\hat{f}_h(x) - f(x)\}^2 dx\right] = \int_{-\infty}^{\infty} MSE\{\hat{f}_h(x)\} dx$$

Choosing the bandwidth h

(a) Define MSE (mean squared error)

$$MSE\{\hat{f}_h(x)\} = E[\{\hat{f}_h(x) - f(x)\}^2]$$

$$\dots$$

$$MSE\{\hat{f}_h(x)\} = Var\{\hat{f}_h(x)\} + [Bias\{\hat{f}_h(x)\}]^2$$

$\rightarrow$ minimizing MSE may solve the trade-off, but h_{opt} depends on $f(x)$ and $f''(x)$, which are unknown.

(b) Define MISE (mean integrated squared error), global measure:

$$MISE\{\hat{f}_h(x)\} = E\left[\int_{-\infty}^{\infty} \{\hat{f}_h(x) - f(x)\}^2 dx\right] = \int_{-\infty}^{\infty} MSE\{\hat{f}_h(x)\} dx$$

(c) Define AMISE (an approximation of MISE) $\rightarrow$ still h_{opt} depends on the unknown $f(x)$, in particular on its second derivative $f''(x)$.

Choosing the bandwidth h

(a) Define MSE (mean squared error)

$$MSE\{\hat{f}_h(x)\} = E[\{\hat{f}_h(x) - f(x)\}^2]$$
$$\dots$$
$$MSE\{\hat{f}_h(x)\} = Var\{\hat{f}_h(x)\} + [Bias\{\hat{f}_h(x)\}]^2$$

$\rightarrow$ minimizing MSE may solve the trade-off, but h_{opt} depends on $f(x)$ and $f''(x)$, which are unknown.

(b) Define MISE (mean integrated squared error), global measure:

$$MISE\{\hat{f}_h(x)\} = E\left[\int_{-\infty}^{\infty} \{\hat{f}_h(x) - f(x)\}^2 dx\right] = \int_{-\infty}^{\infty} MSE\{\hat{f}_h(x)\} dx$$

(c) Define AMISE (an approximation of MISE) $\rightarrow$ still h_{opt} depends on the unknown $f(x)$, in particular on its second derivative $f''(x)$.

(d) One possibility is a plug-in method suggested by Silverman, and consists in assuming that the unknown function is a Gaussian density function (whose variance is estimated by the sample variance). In this case h_{opt} has a simple formulation, and can be defined as a rule-of-thumb bandwidth.

Adaptive Kernel

- Up to know we have seen the possibility of giving higher weights to the observations whose distance from x, the point where we want to estimate the density, is not bigger than $h \rightarrow$ assuming only one h!

Adaptive Kernel

- Up to know we have seen the possibility of giving higher weights to the observations whose distance from x, the point where we want to estimate the density, is not bigger than $h \rightarrow$ assuming only one h!
- But we can get a better estimate by allowing the window width of the kernels to vary from one point to another.

Adaptive Kernel

- Up to know we have seen the possibility of giving higher weights to the observations whose distance from x, the point where we want to estimate the density, is not bigger than $h \rightarrow$ assuming only one h!
- But we can get a better estimate by allowing the window width of the kernels to vary from one point to another.
- In particular, a natural way to deal with long-tailed densities is to use a broader kerne l in regions of low density.

Adaptive Kernel

- Up to know we have seen the possibility of giving higher weights to the observations whose distance from x, the point where we want to estimate the density, is not bigger than $h \rightarrow$ assuming only one h!
- But we can get a better estimate by allowing the window width of the kernels to vary from one point to another.
- In particular, a natural way to deal with long-tailed densities is to use a broader kerne l in regions of low density.
- Thus an observation in the tail would have its mass smudged out over a wider range than one in the main part of the distribution.

Adaptive Kernel: Cont.

- A practical problem is deciding in the first place whether or not an observation is in a region of low density

Adaptive Kernel: Cont.

- A practical problem is deciding in the first place whether or not an observation is in a region of low density
- The adaptive kernel approach copes with this problem by means of a two-stage procedure:

Adaptive Kernel: Cont.

- A practical problem is deciding in the first place whether or not an observation is in a region of low density
- The adaptive kernel approach copes with this problem by means of a two-stage procedure:
  1. get an initial estimate to have a rough idea of the density

Adaptive Kernel: Cont.

- A practical problem is deciding in the first place whether or not an observation is in a region of low density
- The adaptive kernel approach copes with this problem by means of a two-stage procedure:
  1. get an initial estimate to have a rough idea of the density
  2. use the former density to get a pattern of bandwidths corresponding to various observations to be used in a second estimate

Adaptive Kernel: Cont.

In particular:

Step 1 Find a *pilot estimate* $\tilde{f}(x)$ that satisfies $\tilde{f}(x_i) > 0\ \forall i$

Adaptive Kernel: Cont.

In particular:

Step 1 Find a *pilot estimate* $\tilde{f}(x)$ that satisfies $\tilde{f}(x_i) > 0 \ \forall i$

Step 2 Define *local bandwidth factor* λ_i by:

$$\lambda_i = [\tilde{f}(x_i)/g]^{-\alpha} \tag{1}$$

where g is the geometric mean of the $\tilde{f}(x_i)$, $\log g = n^{-1} \sum \log \tilde{f}(x_i)$; and α the *sensitivity parameter* $(\alpha \leq 0)$

Adaptive Kernel: Cont.

In particular:

Step 1 Find a *pilot estimate* $\tilde{f}(x)$ that satisfies $\tilde{f}(x_i) > 0 \; \forall i$

Step 2 Define *local bandwidth factor* λ_i by:

$$\lambda_i = [\tilde{f}(x_i)/g]^{-\alpha} \tag{1}$$

where g is the geometric mean of the $\tilde{f}(x_i)$, $\log g = n^{-1} \sum \log \tilde{f}(x_i)$; and α the *sensitivity parameter* $(\alpha \leq 0)$

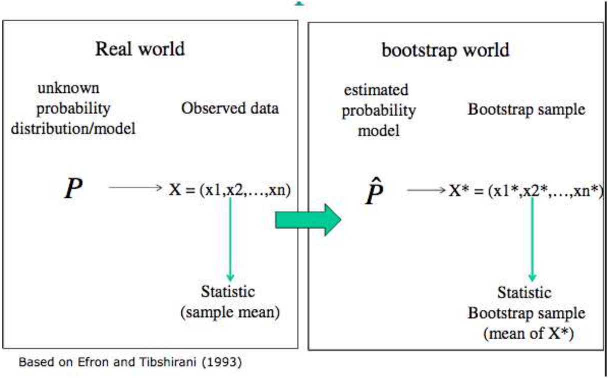
Step 3 Define the *adaptive kernel estimate* $\hat{f}(x)$ by:

$$\hat{f}(x) = nh^{-1} \sum \lambda_i^{-1} K\{h^{-1} \lambda_i^{-1} (x - X_i)\} \tag{2}$$

Bootstrap

- The bootstrap technique allows estimation of the population distribution by using the information based on a number of resamples from the sample.

- The bootstrap technique allows estimation of the population distribution by using the information based on a number of resamples from the sample.



Based on Efron and Tibshirani (1993)

Bootstrap: Cont.

- Use the information of a number of resamples from the sample to estimate the population distribution

Bootstrap: Cont.

- Use the information of a number of resamples from the sample to estimate the population distribution
- Procedure:
  - Given a sample of size n:

Bootstrap: Cont.

- Use the information of a number of resamples from the sample to estimate the population distribution
- Procedure:
  - Given a sample of size n:
    - Treat the sample as population

Bootstrap: Cont.

- Use the information of a number of resamples from the sample to estimate the population distribution
- Procedure:
  - Given a sample of size n:
    - Treat the sample as population
    - Draw B samples of size n with replacement from your sample (the bootstrap samples)

Bootstrap: Cont.

- Use the information of a number of resamples from the sample to estimate the population distribution
- Procedure:
  - Given a sample of size *n*:
    - Treat the sample as population
    - Draw B samples of size n with replacement from your sample (the bootstrap samples)
    - Compute for each bootstrap sample the statistic of interest

Bootstrap: Cont.

- Use the information of a number of resamples from the sample to estimate the population distribution
- Procedure:
  - Given a sample of size n:
    - Treat the sample as population
    - Draw B samples of size n with replacement from your sample (the bootstrap samples)
    - Compute for each bootstrap sample the statistic of interest
    - Estimate the sample distribution of the statistic by the bootstrap sample distribution

Bootstrap: Cont.

- Basic idea: If the sample is a good approximation of the population, bootstrapping will provide a good approximation of the sample distribution.

Bootstrap: Cont.

- Basic idea: If the sample is a good approximation of the population, bootstrapping will provide a good approximation of the sample distribution.
- Justification:

Bootstrap: Cont.

- Basic idea: If the sample is a good approximation of the population, bootstrapping will provide a good approximation of the sample distribution.
- Justification:
  1. If the sample is representative for the population, the sample distribution (empirical distribution) approaches the population (theoretical) distribution if n increases;

Bootstrap: Cont.

- Basic idea: If the sample is a good approximation of the population, bootstrapping will provide a good approximation of the sample distribution.
- Justification:
  1. If the sample is representative for the population, the sample distribution (empirical distribution) approaches the population (theoretical) distribution if n increases;
  2. If the number of resamples (B) from the original sample increases, the bootstrap distribution approaches the sample distribution.

Bootstrap Procedure for Confidence Bands

Given a sample of observations $X = \{X_1, ..., X_m\}$ where each X_i is a vector of dimension n the bootstrap algorithm is the following.

Bootstrap Procedure for Confidence Bands

Given a sample of observations $X = \{X_1, ..., X_m\}$ where each X_i is a vector of dimension n the bootstrap algorithm is the following.

1. Estimate from sample x the density $\hat{f}$.

Bootstrap Procedure for Confidence Bands

Given a sample of observations $X = \{X_1, ..., X_m\}$ where each X_i is a vector of dimension n the bootstrap algorithm is the following.

1. Estimate from sample x the density $\hat{f}$.
2. Select B independent bootstrap samples $\{X^{*1}, ..., X^{*B}\}$, each consisting of n data values drawn with replacement from x.

Bootstrap Procedure for Confidence Bands

Given a sample of observations $X = \{X_1, ..., X_m\}$ where each X_i is a vector of dimension n the bootstrap algorithm is the following.

1. Estimate from sample x the density $\hat{f}$.
2. Select B independent bootstrap samples $\{X^{*1}, ..., X^{*B}\}$, each consisting of n data values drawn with replacement from x.
3. Estimate the density $\hat{f}_b^*$ corresponding to each bootstrap sample $b = 1, ..., B$.

Bootstrap Procedure for Confidence Bands

Given a sample of observations $X = \{X_1, ..., X_m\}$ where each X_i is a vector of dimension n the bootstrap algorithm is the following.

1. Estimate from sample x the density $\hat{f}$.
2. Select B independent bootstrap samples $\{X^{*1}, ..., X^{*B}\}$, each consisting of n data values drawn with replacement from x.
3. Estimate the density $\hat{f}_b^*$ corresponding to each bootstrap sample $b = 1, ..., B$.

The distribution of $\hat{f}^*$ about $\hat{f}$ can therefore be used to mimic the distribution of $\hat{f}$ about f, that is it can be used to calculate the confidence intervals for estimates.

References

Histogram and Density Estimation

Bowman, A.W. and Azzalini A. (1997). Applied smoothing techniques for data analysis: the kernel approach with S-Plus illustrations: the kernel approach with S-Plus illustrations. *Oxford University Press*.

- Estimate: Chapter 1
- Inference (confidence bands): Chapter 2

Adaptive Density Estimation

Silverman, B.W. (1986). Density estimation for statistics and data analysis. *CRC press*.

- Estimate: Chapter 5.3