A Brief Overview of Density Estimation

(MTH516A: Non-Parametric Inference)

Sagnik Dey (201397),
Soumyadip Sarkar (201431),
Saumyadip Bhowmick (201408)

Indian Institute of Technology, Kanpur

April 16, 2022

Topics Covered

1. Introduction
2. Histogram
3. Moving Histogram
4. Kernel Density Estimation
5. Bandwidth Selection Method
6. Practical Problems
7. Conclusions

Why Estimate Density

- To have an idea of the properties of a given Dataset.
- Density provides indication of skewness, multimodality in the data.

How to Estimate Density

- A common choice is Histogram.
- Suppose we have an i.i.d. sample $X_1, X_2, .., X_n$ from some distribution, and we want to estimate the density of that population.
- The histogram at a point x is defined as,

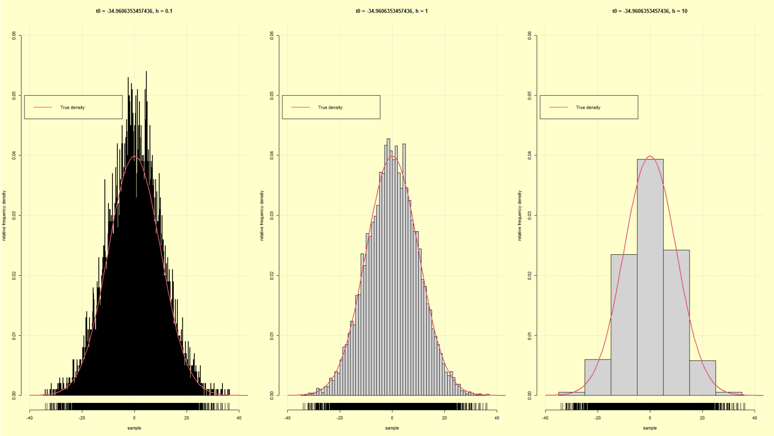
$$\hat{f}_H(x; t_0, h) := \frac{1}{nh} \sum_{i=1}^{n} 1_{\{X_i \in B_k : x \in B_k\}}$$

where, $\{B_k := [t_k, t_{k+1}) : t_k = t_0 + hk, k \in \mathbb{Z}\}$

- i.e. first choose t_0 and h, it will provide B_k, then plot $\hat{f}_H(x; t_0, h)$ in each B_k.

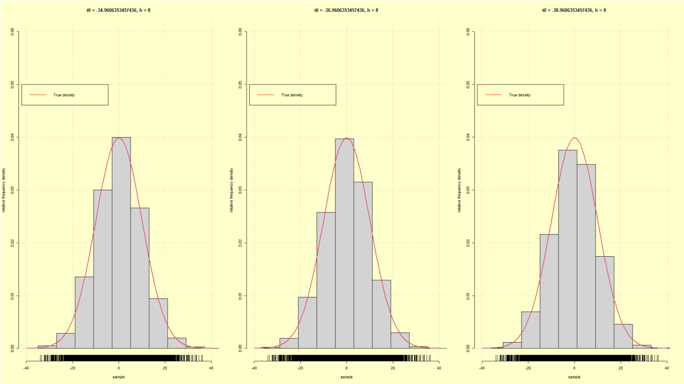
Example with varying h

- We have a sample from $N(0, 10^2)$
- Histograms are plotted using fixed t_0 = min(sample) - 1 and h = 0.1, 1, 10.



.

- Again for the previous sample from $N(0, 10^2)$ histograms are plotted using varying t_0 = min(sample) - 1 , min(sample) - 3, min(sample) - 5 and fixed h = 8.



Moving Histograms

.

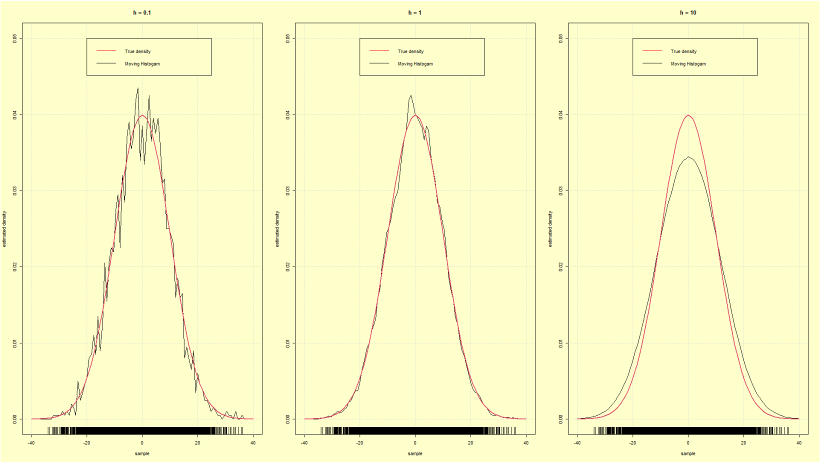
- We can see histograms are dependent on choice of t_0. An alternative method to avoid the dependence on t_0 is the moving histogram, also known as naive density estimator.
- Given a $h > 0$, the naive density estimator builds a piecewise constant function by considering the relative frequency of $X_1, X_2, ..., X_n$ inside $(x - h, x + h)$:

$$\hat{f}_N(x; h) := \frac{1}{2nh} \sum_{i=1}^{n} 1_{\{x-h<X_i<x+h\}}$$

- i.e. first choose h, then plot $\hat{f}_N(x; h)$ in the interval (x-h,x+h).

Example with varying h

- Again for the previously used sample from $N(0, 10^2)$ moving histograms are plotted for h $= 0.1$, 1, 10.



.

- The moving histogram can be equivalently written as,

$$\begin{aligned}\hat{f}_N(x;h) &= \frac{1}{2nh}\sum_{i=1}^{n} 1_{\{x-h<X_i<x+h\}} \\ &= \frac{1}{nh}\sum_{i=1}^{n}\frac{1}{2}1_{\{-1<\frac{x-X_i}{h}<1\}} \\ &= \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x-X_i}{h}\right) \qquad \left[where, K(z) = \frac{1}{2}1_{\{-1<z<1\}}\right] (*)\end{aligned}$$

by (*) we are giving equal weight to all the points in neighborhood of $X_1, X_2, ..., X_n$. So from generalisation of (*) to non-uniform weighting, we can replace K by any arbitrary density. This K is known as Kernel.

Kernels

.

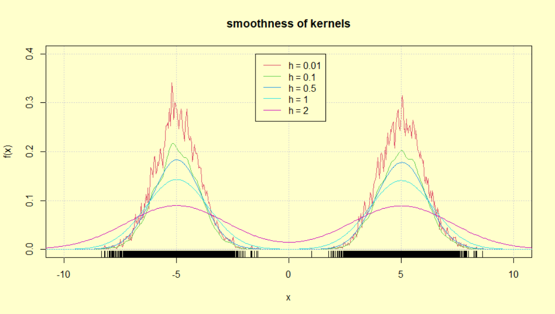
- Properties of Kernel
  1. Kernel functions are symmetric about 0.
  2. Since these are densities, $\int_{-\infty}^{\infty} K(z)dz = 1$
- Examples of Kernels The following tables shows some popularly used kernels to estimate density.

Names	Density
Rectangular/Uniform	$\frac{1}{2}1_{\{\|z\|<1\}}$
Triangular	$(1-\|z\|)1_{\{\|z\|<1\}}$
Epanechnikov	$\frac{3}{4}(1-z^2)1_{\{\|z\|<1\}}$
Gaussian	$\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}z^2\right)$

What is Bandwidth

.

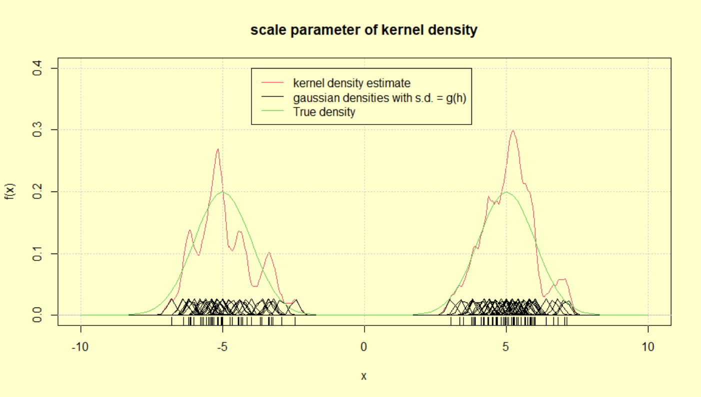
- Bandwidth h is the smoothing parameter of the Kernel function.
- Small values of h makes the estimated density curve wiggly whereas large values of h smooth out the estimated density.
- KDE for simulated data from Gaussian Mixture, with Gaussian kernel and varying h.



What is Bandwidth

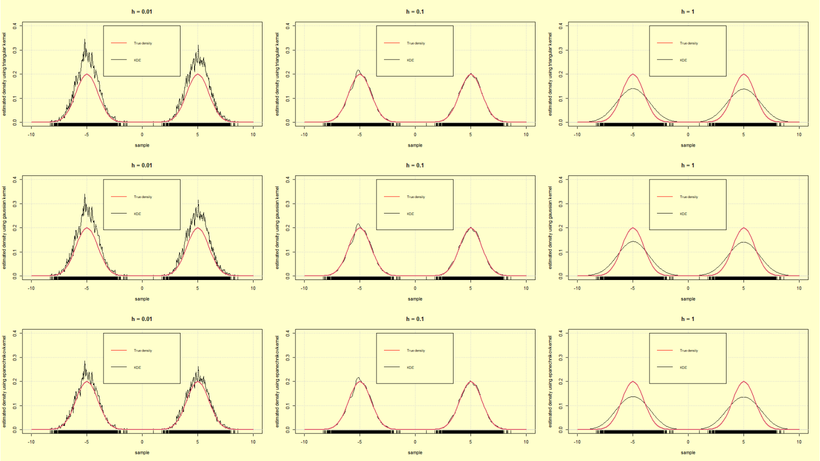
.

- For any Kernel function, the standard deviation is a function of h, hence h is a measure of variability of the variable associated with the Kernel.



What is Bandwidth

- if a certain smoothness(h) is guaranteed (continuity at least), the choice of the kernel has little importance in practice,



Bias and Variance of Kernel Density Estimates

Now,

$$bias[\hat{f}(x;h)] = E[\hat{f}(x;h)] - f(x) = \frac{1}{2}\mu_2(K)f''(x)h^2 + o(h^2)$$

and,

$$Var[\hat{f}(x;h)] = \frac{R(K)}{nh}f(x) + o((nh)^{-1})$$

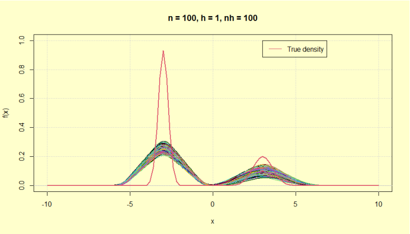
where, $\mu_j(K) = \int x^j K(x)dx$
and, $R(K) = \int K^2(z)dz$

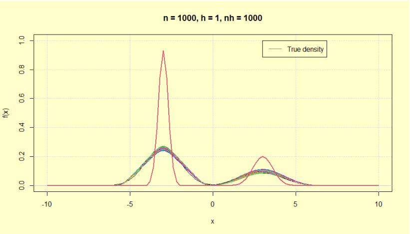
Interpretations

- The bias at x is directly proportional to $f''(x)$. This has some interesting interpretations:
  1. The bias is negative where f is concave, $f'' < 0$. These regions correspond to the peaks and modes of f, where the kde underestimates f.
  2. Conversely the bias is positive where f is convex, $f'' > 0$.These regions correspond to valleys or tails of f, where kde overestimates f.
  3. The wilder is the curvature f", the harder is to estimate f. Flat density regions are easier to estimate than wiggling regions with high curvatures.
- The variance depends directly on f(x). The higher the density the more variable is the kde. Interestingly, the variance decreases as a factor of $(nh)^{-1}$.

Example using simulation

- We have simulated data 1000 times from $0.7N(-3, 0.3) + 0.3N(3, 0.6)$ distribution.
- at 3,-3, $f" < 0$, f is concave, so bias is negative, near -4,4, $f" > 0$, f is convex, so bias is positive.
- at 3, -3 f(x) is high so is variance, at 4,-4 f(x) is low so is variance.
- As *nh* inrceases, variance of the density estimates decreases and bias remains similar.





How "good" is the estimator

.

- Since Kernel density estimation critically depends on bandwidth, we use automatic bandwidth selectors that attempt to minimise the error in estimaton of the target density f.
- There are many error criteria to judge the goodness of fit like ISE, MISE, AMISE.
- There are different bandwidth selection methods depending on which criterion like ISE, MISE or AMISE is being minimised.
- In this project we restrict our study upto minimising ISE.
- Integrated Squared Error (ISE) is defined as,

$$\begin{aligned} ISE(\hat{f}(x;h)) &= \int (\hat{f}(x;h) - f(x))^2 dx \\ &= R(\hat{f}(x;h)) - 2E_{f(x)}[\hat{f}(x;h)] + R(f(x)) \end{aligned}$$

Least Squares Cross-Validation method

.

- Here the last term is independent of h and hence minimising ISE becomes equivalent to minimisation of first 2 terms.
- This unknown quantity can be estimated unbiasedly as,

$$LSCV(h) := \int \hat{f}(x;h)^2 dx - \frac{2}{n}\sum_{i=1}^{n} \hat{f}_{-i}(Xi;h)$$

- here $\hat{f}_{-i}(.,h)$ is the leave-one-out kernel density estimate and is based on the sample with the X_i removed

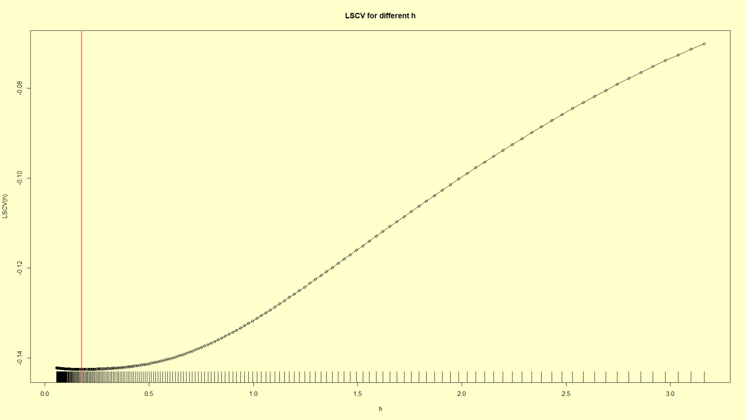
$$\hat{f}_{-i}(x;h) = \frac{1}{n-1}\sum_{j=1;j\neq i}^{n} K_h(x - X_j)$$

.

- Here the data used for computing the kde is not used for its evaluation, so this is a cross validatory way of bandwidth selection, known as **Least Squares Cross-Validation method**.

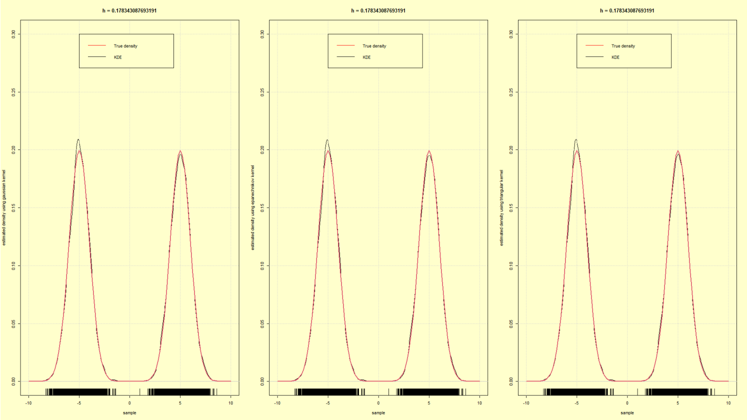
LSCV for Gaussian Mixture

- We have a sample from $0.5N(-5,1)+0.5N(5,1)$ distribution.
- We plot LSCV values for different values of h, and choose that h for which LSCV is minimum.



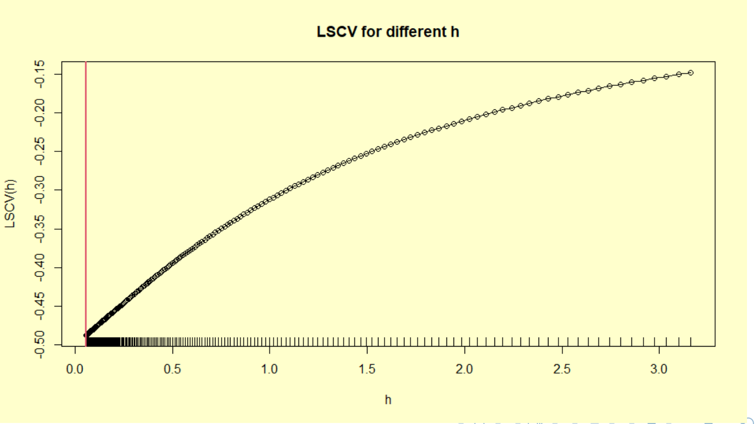
LSCV for Gaussian Mixture

- minima is at $h_{LSCV} = 0.1783431$.
- We plot KDEs using different kernels and bandwidth $h_{LSCV} = 0.1783431$.



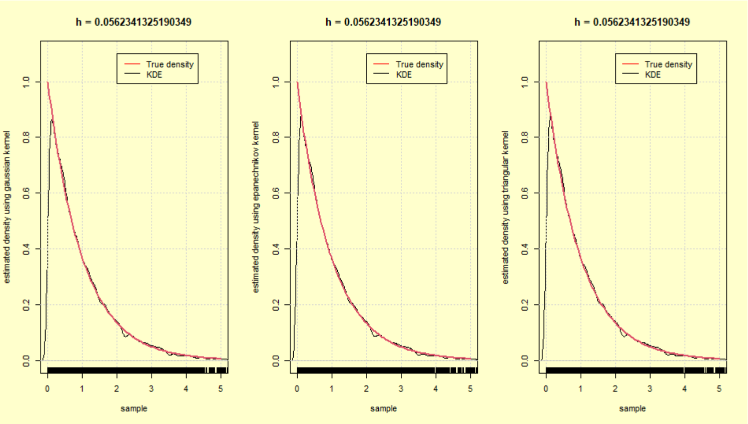
Practical issues

- Sometimes LSCV(h) may not have global minima at all, or may have several local minimas.
- Let us estiamte density of a sample from exp(1) distribution.
- the LSCV plot is as follows:



Continuation

- We see there is no global minima.
- We plot KDEs using different kernels and bandwidth $h_{LSCV} = 0.05623413$, which is near to 0.



Continuation

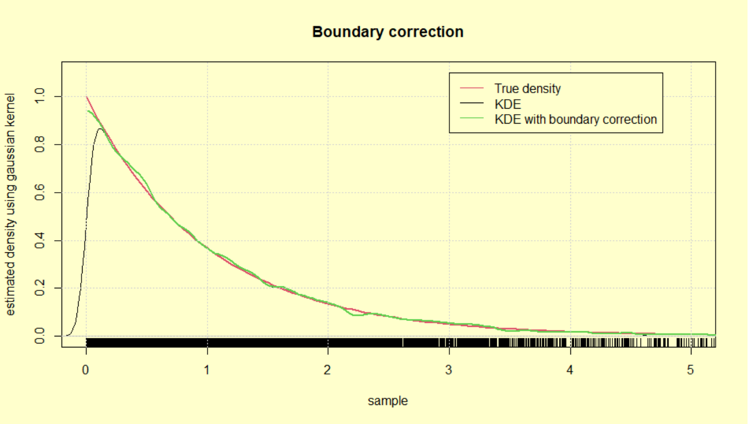
- In case of any density with a support with boundary, for example Exponential distribution in $(0, \infty)$, kde runs into trouble.
- Here what happens is that since kde is defined over entire real line, it is spreading probability mass outside the support of the distribution.
- This results in a severe negative bias about 0. As a result the kde does not integrate to 1 in the support of the data.

Reflection method

- It is one of the simplest approach for boundary correction.
- Once the sample $X_1, X_2, ..., X_n$ is obtained set the new sample of size 2n, $Y_j = \begin{cases} -X_j & \text{for j = 1,2,...,n} \\ X_{2n-j+1} & \text{for j = n+1,n+2,...,2n} \end{cases}$
- Now estimate kde on this new sample constructed, say $\hat{g}(y; h)$.
- Finalise the estimate such that:
  1. for $y \geq 0$, final density is $2\hat{g}(y; h)$
  2. for $y < 0$, final density is 0.

Reflection method

- Boundary correction for previously simulated data from exp(1), using gaussian kernel.



Thank You !!!